

Klasifikasi Citra Sistem Isyarat Bahasa Indonesia (SIBI) dengan Metode *Convolutional Neural Network* pada Perangkat Lunak berbasis *Android*

Sherryl Sugiono Sindarto¹, Dian Eka Ratnawati², Issa Arwani³

Program Studi Teknologi Informasi, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹sherrylsin20@student.ub.ac.id, ²dian_ilkom@ub.ac.id, ³issa.arwani@ub.ac.id

Abstrak

Sistem Isyarat Bahasa Indonesia (SIBI) yang dibakukan merupakan salah satu media yang membantu komunikasi sesama kaum tunarungu dan tunawicara di dalam masyarakat yang lebih luas. Terhitung 211.889 penduduk Indonesia merupakan penyandang disabilitas yang terdiri dari 6.5% (13.802) adalah tunarungu dan 2.6% (5.580) adalah tunawicara. Banyak orang awam belum mengerti bahasa isyarat yang menjadi batasan untuk berkomunikasi dengan kaum tunarungu dan tunawicara. Penelitian ini mengembangkan aplikasi IBIS dengan sistem penerjemah bahasa isyarat secara langsung. Sistem penerjemah bahasa isyarat secara langsung dikembangkan menggunakan metode *convolutional neural network* dengan *Tensorflow Lite Model Maker* sebagai media pengembangannya. Peneliti menggunakan metode *convolutional neural network* karena hasil akurasi yang relatif tinggi untuk model deteksi objek. Model diintegrasikan ke aplikasi berbasis *Android* yang dikembangkan dengan *framework Flutter*. Pengembangan aplikasi IBIS dimulai dari perancangan sistem dan antarmuka menggunakan metode perancangan *waterfall*. Selanjutnya dilakukan implementasi sistem sesuai dengan kebutuhan yang didefinisikan. Integrasi model menggunakan *plugin tflite_flutter* dan *tflite_flutter_helper*. Setelah itu dilakukan pengujian terhadap aplikasi IBIS dan model deteksi objek. Pengujian aplikasi terdiri dari pengujian validasi dan pengujian *usability*. Pengujian validasi menggunakan metode *blackbox* dengan hasil pengujian menunjukkan fungsionalitas sistem sesuai dengan kebutuhan yang didefinisikan. Pengujian *usability* dengan metode *System Usability Scale* (SUS) mencapai nilai 86 dan masuk dalam kategori *acceptable*. Pengujian model deteksi objek dilakukan dengan membandingkan kelas sebenarnya dengan kelas yang dideteksi. Pengujian akurasi mencapai 88% untuk 15 kelas.

Kata kunci: bahasa isyarat, deteksi objek, *convolutional neural network*, *Tensorflow Lite*

Abstract

The standardized Indonesian Sign System (SIBI) is one of the media that helps communication among the deaf and mute in a wider community. It is known that 211.889 Indonesians are persons with disabilities consisting of 6.5% (13.802) are deaf and 2.6% (5.580) are speech impaired. Many ordinary citizens do not understand sign language which becomes a limitation for communicating with the deaf or mute. This research develops an application named IBIS with real-time sign language translator system. The real-time sign language translator system is developed using the convolutional neural network method with *Tensorflow Lite Model Maker* as the development medium. Researcher used the convolutional neural network method as the accuracy is relatively high. The model is integrated into *Android* based application developed with *Flutter framework*. IBIS application development starts from system and interface design using the *waterfall method*. Furthermore, the system is implemented in accordance to the defined requirements. The model is integrated into the *Android* based application using *tflite_flutter* and *tflite_flutter_helper* plugin. After that, testing is carried out for IBIS application and object detection model. The test for application testing includes validation testing and usability testing. The validation test is carried out using the *blackbox* method with the results show that all functionalities is in accordance with the defined requirements. Usability test with *System Usability Scale* (SUS) method reached a value of 86 and fall into the *acceptable* category. Testing for object detection model is done by comparing the original class with the detected class. The accuracy test reached 88% for 15 classes.

Keywords: sign language, object detection, *convolutional neural network*, *Tensorflow Lite*

1. PENDAHULUAN

Sistem Isyarat Bahasa Indonesia (SIBI) yang dibakukan merupakan salah satu media yang membantu komunikasi sesama kaum tunarungu dan tunawicara pada masyarakat yang lebih luas (Kementerian Pendidikan dan Budaya Republik Indonesia, 2021). Terhitung 211.889 penduduk Indonesia merupakan penyandang disabilitas yang terdiri dari 6.5% (13.802) adalah tunarungu dan 2.6% (5.580) adalah tunawicara (Kementerian Sosial Republik Indonesia, 2021). Minat masyarakat Indonesia dalam mempelajari bahasa isyarat tergolong tinggi, namun kurangnya media pembelajaran Bahasa Isyarat Indonesia menjadi salah satu faktor yang mempersulit masyarakat dalam mengerti bahasa isyarat. Kurikulum pendidikan di Indonesia juga belum sepenuhnya dapat mengakomodasi kebutuhan penyandang disabilitas rungu maupun wicara (Kementerian Sosial Republik Indonesia, 2021).

Banyak orang awam belum mengerti bahasa isyarat yang menjadi batasan untuk berkomunikasi dengan kaum tunarungu dan tunawicara. Penulis ingin memanfaatkan *machine learning* untuk membuat jembatan antara orang awam dan tunarungu dalam berkomunikasi. Pengguna aplikasi dapat menerjemahkan bahasa isyarat secara langsung menggunakan *smartphone Android* yang terintegrasi dengan *machine learning*. *Machine learning* adalah bagian dari *Artificial Intelligence* (AI) yang menjadi alat untuk mengoptimalkan kinerja sistem.

Klasifikasi citra atau *image classification* merupakan analisa dan kategorisasi citra ke dalam satu atau lebih kelas. Beberapa metode klasifikasi citra adalah *Support Vector Machine* (SVM), *Artificial Neural Network* (ANN), dan *Decision Tree* (Dhaware and Wanjale, 2016). Penelitian tentang klasifikasi *American Sign Language* (ASL) pernah dilakukan oleh Fatmi, Rashad, dan Integlia yang membandingkan metode ANN, SVM, dan *Hidden Markov Model* (HMM). Metode ANN mencapai hasil akurasi paling tinggi yaitu 93.79% diikuti dengan SVM sebesar 85.56% dan HMM 85.90%. Dari penelitian ini, dibuktikan bahwa ANN memberikan hasil akurasi yang paling baik untuk mengenali ASL (Fatmi, Rashad and Integlia, 2019). Penulis memilih metode ANN untuk klasifikasi citra karena hasil akurasi dari penelitian sebelumnya tergolong tinggi.

Penelitian tentang deteksi bahasa isyarat pernah dilakukan oleh Hoque dan penulis lainnya yang mendeteksi bahasa isyarat Bangladeshi menggunakan metode *Faster Region Based Convolutional Neural Networks* (R-CNN) dan menghasilkan model yang dapat melakukan klasifikasi secara langsung (Hoque et al., 2019). Penelitian tentang klasifikasi video gestur bahasa isyarat Indonesia pernah dilakukan oleh Setyono dan Rakun menggunakan metode *Deep Convolutional Neural Network* (DeepCNN) dan *Bidirectional Long Short-Term Memory* (BiLSTM) dengan hasil akurasi dan waktu komputasi yang cukup tinggi (Setyono and Rakun, 2019). Selain itu penelitian oleh Suharjito, Thiracitta, dan Gunawan yang membuat model klasifikasi SIBI menggunakan metode CNN digabungkan dengan *transfer learning* dan parameter *non-trainable* memiliki akurasi yang cukup tinggi walaupun dengan jumlah data yang kecil untuk *training* (Suharjito, Thiracitta and Gunawan, 2021). Penelitian yang dilakukan oleh Rachardi menghasilkan *prototype* yang dapat dijalankan di *smartphone Android* untuk mengklasifikasi BISINDO (Rachardi, 2020).

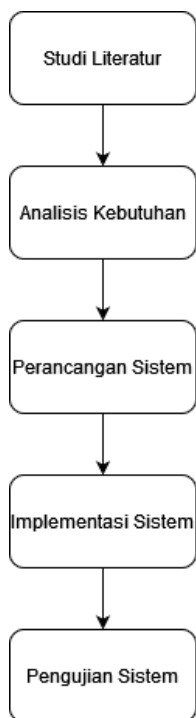
Penelitian yang dilakukan oleh penulis menggunakan dataset SIBI yang terdiri dari 15 kelas dengan jumlah data 300 citra. Setiap kelas memiliki 20 citra yang terdiri dari alfabet, numerik, kata ganti orang, dan kata dasar untuk percakapan sehari-hari yang ditentukan oleh penulis. Penulis memutuskan untuk mengklasifikasikan 15 kata SIBI untuk meningkatkan akurasi sistem klasifikasi yang sudah dikembangkan sebelumnya. Gestur yang dipilih untuk klasifikasi memiliki perbedaan yang signifikan satu sama lain, sehingga kesalahan klasifikasi karena kemiripan gestur dapat dihindari. Seluruh citra dari dataset SIBI memiliki variasi latar belakang dan pencahayaan untuk mengurangi bias pada saat tahap *training*. Model klasifikasi citra dikembangkan dengan *transfer learning* untuk meningkatkan akurasi deteksi dan klasifikasi. *Transfer learning* menggunakan model *EfficientDet-Lite4* yang memiliki dasar arsitektur *convolutional neural network*. Model *EfficientDet-Lite4* di optimasi untuk pengembangan model deteksi objek dengan hasil akurasi yang cukup tinggi.

Model dikonversi ke format TFLite dan diimplementasikan ke perangkat lunak berbasis *Android* untuk klasifikasi citra dengan deteksi objek secara langsung. Fitur deteksi objek secara langsung dilakukan dengan menggunakan

kamera *smartphone* untuk mengambil gambar *real-time* dan mengembalikan label kelas beserta *bounding box* sebagai hasil deteksi objek.

2. METODOLOGI

Metode penelitian diawali dengan pengumpulan data, lalu dilanjutkan dengan tahap studi literatur, tahap analisis kebutuhan, tahap perancangan, tahap implementasi, dan tahap pengujian sistem. Metode penelitian digunakan untuk mengembangkan model klasifikasi citra dengan deteksi objek dan mengembangkan perangkat lunak berbasis *Android*. Metode yang digunakan untuk klasifikasi citra adalah *Convolutional Neural Network (CNN)* dengan *transfer learning*. Salah satu kelebihan CNN adalah tidak memerlukan langkah *preprocessing* dan bisa secara otomatis mendeteksi fitur unik tanpa pengawasan. Metode yang digunakan untuk pengembangan perangkat lunak berbasis *Android* adalah metode *waterfall*. Diagram alur untuk penelitian ini dapat dilihat pada Gambar 1.



Gambar 1. Alur Diagram Penelitian

2.1. Pengumpulan Data

Pengumpulan data primer diambil melalui relawan yang bersedia serta situs resmi Kamus Sistem Isyarat Bahasa Indonesia yang didirikan oleh Kementerian Pendidikan dan Budaya Republik Indonesia. Data terdiri dari 15 kelas

dengan masing-masing kelas memiliki 20 citra. Setiap gerakan berdasarkan gestur yang terdaftar di situs Kamus SIBI yang didirikan oleh Kementerian Pendidikan dan Budaya Republik Indonesia. Gestur SIBI yang digunakan sebagai kelas terdiri dari alfabet, kata ekspresi, kata kerja, kata ganti orang, dan kata obyek. Daftar gestur SIBI beserta indeks yang digunakan dapat dilihat pada Tabel 1.

Tabel 1. Daftar Gestur SIBI

Indeks	Gestur SIBI
0	Aku
1	B
2	Bayar
3	Cinta
4	Doa
5	G
6	K
7	L
8	O
9	Pikir
10	Rumah
11	T
12	Takut
13	Telepon
14	Uang

2.2. Klasifikasi Citra

Klasifikasi citra merupakan proses kategorisasi dan pemberian label dari kumpulan *pixel* atau *vector* yang ada di dalam sebuah citra dengan aturan spesifik. Terdapat beberapa teknik klasifikasi citra seperti *neural network*, *support vector machine*, *fuzzy logic*, dan *genetic algorithm* (Gavali and Banu, 2019). Pada penelitian ini, peneliti memilih teknik *neural network* karena memiliki waktu komputasi yang cepat dan hasil akhir yang relatif tinggi.

2.3. Convolutional Neural Network (CNN)

Convolutional neural network terdiri dari *neuron* yang melakukan optimisasi otomatis melalui tahap *learning*. Perbedaan CNN dengan ANN untuk klasifikasi citra adalah ANN memerlukan data spesifik untuk menentukan fitur pada citra secara manual. Selain itu ANN juga harus mengubah citra 2 dimensi menjadi vektor 1 dimensi yang menambah parameter pelatihan, sekaligus menambah kapasitas memori dan komputasi lebih banyak. CNN bisa secara otomatis mendeteksi fitur unik yang diekstraksi dari *input* citra.

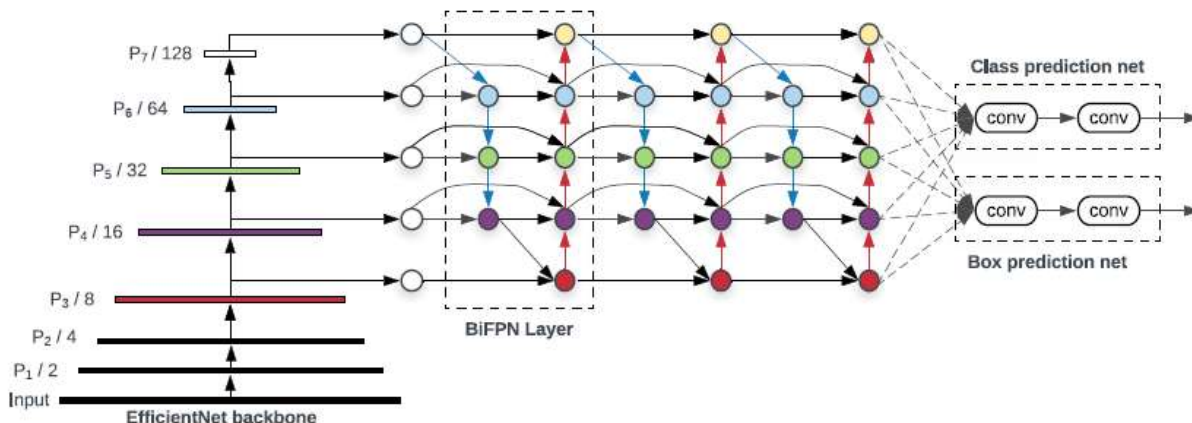
CNN memiliki 3 tipe *layer* yaitu *convolutional*, *pooling*, dan *fully-connected layers*. *Layer input* sebagai nilai *pixel* dari citra. *Convolutional layer* yang menentukan *output*

dari *neuron* yang terkoneksi dengan input. *Pooling layer* bertugas untuk melakukan *downsampling* pada dimensi spasial dari *input*. Dan yang terakhir adalah *fully-connected layers* yang memiliki tugas untuk menghasilkan nilai prediksi (O’Shea and Nash, 2015). Arsitektur sederhana dari CNN dapat dilihat pada Gambar 2.

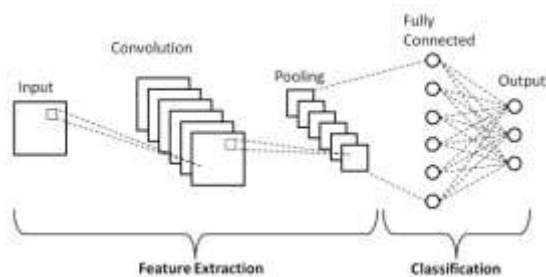
2019). Arsitektur model *EfficientDet-Lite4* dapat dilihat pada Gambar 3.

2.5. Flutter

Perangkat lunak *open source* untuk membuat aplikasi berbasis *iOS* dan *Android* yang memiliki performa dan fidelitas tinggi. (A. Madhuram, M, 2019). Beberapa fitur penting dari *Flutter* adalah cara eksekusi kode yang



Gambar 2. Arsitektur Model *EfficientDet-Lite 4* (Tan, Pang and Le, 2019)



Gambar 3. Arsitektur CNN (Phung and Rhee, 2019)

2.4. EfficientDet-Lite 4

Model yang digunakan untuk *transfer learning* adalah *EfficientDet-Lite4*. Arsitektur dari model ini terdiri dari 3 jaringan, yaitu jaringan *backbone*, jaringan *Bi-directional Feature Pyramid Network* (BiFPN), serta jaringan prediksi kelas dan kotak. Jaringan *backbone* menggunakan pengukuran tinggi dan kedalaman dari model *EfficientNet*. Jaringan BiFPN digunakan untuk ekstraksi fitur yang memungkinkan informasi untuk mengalir dari atas-bawah dan sebaliknya. BiFPN juga menambah bobot tambahan untuk setiap fitur *input* dan memungkinkan jaringan untuk mempelajari setiap fitur. Jaringan prediksi kelas dan kotak digunakan sebagai hasil dari model yang terdiri dari label hasil klasifikasi dan *bounding box* hasil deteksi (Tan, Pang and Le,

melakukan kompilasi pada eksekusi program secara bersamaan.

2.6. TensorFlow Lite

TensorFlow Lite adalah kerangka kerja *machine learning* yang ringan, cepat, dan *cross-platform* yang didesain khusus untuk *mobile* dan *IoT*. *TensorFlow Lite* adalah bagian dari *TensorFlow* dan mendukung banyak *platform* seperti *Android*, *iOS*, dan *embedded Linux*. *TensorFlow Lite* membantu mengurangi kesulitan pengembang dan mempercepat pengembangan *On-Device Machine Learning* (Li, 2020).

2.7. Deteksi Objek

Object Detection menentukan keberadaan sebuah objek dan ruang lingkup, serta lokasi objek pada gambar. *Object detection* dapat dibagi menjadi *soft detection* yang hanya mendeteksi keberadaan dari objek dan *hard detection* yang mendeteksi keberadaan serta lokasi dari objek (Jalied and Voronkov, 2016). *TensorFlow Lite Task Library* menyediakan *API* yang bisa diintegrasikan dengan model deteksi objek. Model deteksi objek dilatih untuk mendeteksi kehadiran dan lokasi objek dari berbagai kelas. Dalam penelitian ini, model deteksi objek dilatih untuk mendeteksi bahasa

isyarat.

3. PERANCANGAN

3.1. Gambaran Umum Aplikasi

Penelitian ini mengembangkan aplikasi penerjemah Sistem Isyarat Bahasa Indonesia berbasis Android. Tujuan dari pengembangan aplikasi penerjemah bahasa isyarat adalah untuk memudahkan orang awam dalam berkomunikasi dengan penyandang disabilitas rungu atau wicara. Aplikasi yang dibuat mengintegrasikan *Application Programming Interface* (API) deteksi objek dan klasifikasi citra menggunakan *TensorFlow Lite* untuk menerjemahkan Sistem Isyarat Bahasa Indonesia (SIBI) secara langsung menggunakan kamera gawai.

3.2. Kebutuhan Fungsional

Kebutuhan fungsional merupakan kebutuhan yang terdiri dari proses yang harus dilakukan dan informasi yang harus disediakan oleh sistem. Kebutuhan fungsional dapat dilihat pada Tabel 2.

Tabel 2. Daftar Kebutuhan Fungsional

No.	Kode	Kebutuhan Fungsional	Use case
1.	IBIS-F-001	Sistem dapat menampilkan halaman pengenalan aplikasi	Melihat pengenalan aplikasi
2.	IBIS-F-002	Sistem dapat menampilkan daftar kursus	Melihat daftar kursus
3.	IBIS-F-003	Sistem dapat menampilkan daftar materi	Melihat daftar materi
4.	IBIS-F-004	Sistem dapat menampilkan rincian materi	Melihat rincian materi
5.	IBIS-F-005	Sistem dapat memperbaharui status penyelesaian materi	Menyelesaikan materi
6.	IBIS-F-006	Sistem dapat menerjemahkan Sistem Isyarat Bahasa Indonesia	Menggunakan kamera klasifikasi citra
7.	IBIS-F-007	Sistem dapat menampilkan informasi aplikasi	Melihat informasi aplikasi

3.3. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional adalah kebutuhan yang menitikberatkan pada properti perilaku sistem dan mendukung sistem untuk berjalan dengan baik. Kebutuhan non-fungsional dapat dilihat pada Tabel 3.

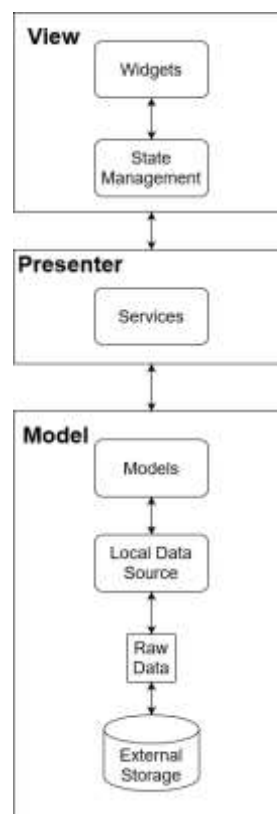
Tabel 3. Daftar Kebutuhan Non-Fungsional

No.	Kode	Kebutuhan Non-Fungsional	Parameter
1.	IBIS-NF-001	Sistem dapat digunakan dengan mudah oleh pengguna	Usability

3.4. Perancangan Arsitektur Sistem IBIS

Arsitektur sistem aplikasi IBIS menggunakan pola arsitektur *Model-View-Presenter* (MVP). Lapisan *view* digunakan untuk menampilkan *interface* ke pengguna. Lapisan *view* terdiri dari *widgets* yang menyusun antarmuka aplikasi *Flutter* dan *state management* untuk manajemen view apabila terjadi interaksi oleh pengguna pada *widgets*. Lapisan *presenter* memiliki kewajiban untuk menjembatani komunikasi antara lapisan *view* dan lapisan *model*. Lapisan *presenter* memanggil data dari model sebelum dikembalikan ke lapisan *view*.

Lapisan *presenter* terdiri dari *services* yang berisi layanan untuk memanggil data dan melakukan pembaharuan pada data. Lapisan *model* digunakan untuk menyimpan data yang diterima dari basis data. Lapisan model terdiri dari model yang digunakan untuk memetakan data yang berasal dari sumber data lokal yaitu penyimpanan eksternal gawai. Arsitektur sistem dapat dilihat pada Gambar 4.



Gambar 4. Arsitektur Sistem IBIS

3.5. Diagram Class

Diagram *class* menggambarkan model spesifikasi kebutuhan yang membentuk perangkat lunak berdasarkan analisis kebutuhan yang sudah didefinisikan sebelumnya. Relasi antar kelas dapat dilihat pada Gambar 5.

Tabel 4. Kode Implementasi *Splash Screen*

No.	Kode
1	...
2	void initState() {
3	super.initState();
4	startTime();
5	print('init splash');
6	}
7	...
8	startTime() async {
9	var duration = new Duration(
10	seconds: 5);
11	return Timer(duration,
12	navigationPage);
13	}
14	void navigationPage() {
15	GetStorage box = GetStorage();
16	bool _seenSplash = box.read(
17	'seen') ?? false;
18	if (!_seenSplash) {
19	box.write('seen', true);
20	Get.offAllNamed('onboard');
21	} else {
22	Get.offAllNamed('/home');
23	}
24	}

4.3. Implementasi Kode Model Klasifikasi

Model klasifikasi citra menggunakan pustaka *TensorFlow Lite* dan API deteksi objek. *Transfer learning* menggunakan *pre-trained* model *EfficientDet Lite-4* untuk mempercepat proses pelatihan dari dataset SIBI. Implementasi model klasifikasi citra menggunakan Google Colab Pro dan bahasa pemrograman Python.

Dataset SIBI dibagi menjadi 85% untuk *training* dan 15% untuk *testing* sebelum diubah menjadi format *TFRecord* menggunakan layanan *Roboflow*. Tahap selanjutnya adalah *training* model klasifikasi dengan iterasi sebanyak 100 kali. Implementasi kode model dapat dilihat pada Tabel 5.

Tabel 5. Kode *Training Model*

No.	Kode
1	spec = model_spec.get(
2	'efficientdet_lite4')
3	
4	model = object_detector.create(
5	train_data, model_spec=spec,
6	epochs=100, validation_data=
7	test_data, batch size=8)

Tahap training membutuhkan rata-rata waktu 80 detik setiap iterasi dengan jumlah data setiap iterasi sebanyak 51 citra. Hasil dari *training* adalah *loss* sebesar 43% (0.43) dan *val_loss* sebesar 60% (0.60). Berdasarkan hasil *training*, akurasi klasifikasi citra dengan deteksi objek untuk gestur bahasa isyarat memiliki performa yang baik. Validasi pada dataset *testing* juga memiliki performa yang sedang.

4.4. Implementasi Struktur Data JSON

Kursus IBIS disimpan dalam bentuk JSON yang terdiri dari beberapa objek yaitu objek *courses*, *title*, *description*, *length*, *progress*, *lessons*, *name*, *explanation*, *video*, dan *status*. Berdasarkan perancangan struktur data JSON pada tahap sebelumnya, implementasi file JSON dapat dilihat pada Gambar 9.



Gambar 9. Implementasi Struktur Data JSON

5. PENGUJIAN

Pengujian aplikasi terdiri dari pengujian validasi, dan pengujian *usability*. Pengujian model terdiri dari pengujian akurasi dengan membandingkan hasil prediksi dari model klasifikasi citra dengan kelas yang sebenarnya.

5.1. Pengujian Validasi

Pengujian validasi dilakukan dengan metode *blackbox* untuk menguji apakah sistem sudah memenuhi kebutuhan fungsional yang didefinisikan pada tahap perancangan. Salah satu contoh pengujian validasi dapat dilihat pada Tabel 6.

Tabel 6. Hasil Pengujian Validasi

Kebutuhan	Sistem dapat menampilkan halaman pengenalan aplikasi
Test Case	1. Pengguna membuka aplikasi 2. Sistem menampilkan halaman pembuka selama 5 detik 3. Sistem melakukan navigasi ke halaman pengenalan setelah halaman pembuka selesai
Hasil yang diharapkan	Sistem dapat menampilkan halaman pengenalan saat pengguna membuka aplikasi untuk pertama kalinya
Hasil Akhir	Sistem menampilkan halaman pengenalan

5.2. Pengujian Usability

Pengujian *usability* dilakukan dengan metode *System Usability Scale* (SUS)

mengevaluasi pengalaman pengguna dalam menggunakan sistem. Kuisisioner terdiri dari 10 pertanyaan dan disebarikan kepada 5 pengguna. Berdasarkan nilai responden rata-rata nilai pengalaman pengguna saat melakukan interaksi dengan sistem aplikasi IBIS adalah 90,5. Rata-rata nilai tersebut masuk dalam kategori *Excellent* dengan *grade A*. Dari nilai rata-rata yang didapatkan, dapat disimpulkan bahwa kebutuhan non-fungsional *usability* aplikasi IBIS yang cukup baik dan dapat diterima oleh pengguna.

5.3. Pengujian Akurasi

Pengujian akurasi deteksi model dilakukan secara manual dengan menggunakan aplikasi IBIS untuk deteksi objek secara langsung. Pengujian dilakukan dengan mengarahkan kamera dari gawai melalui aplikasi IBIS ke citra yang tidak masuk dalam dataset SIBI sebanyak 10 kali untuk setiap kelas. Hasil dari pengujian akurasi dapat dilihat pada Tabel 7.

Tabel 7. Hasil Pengujian Akurasi

Indeks	Label	1	2	3	4	5	6	7	8	9	10	Akurasi
0	Aku	X	0	0	0	0	0	0	0	0	0	0,9
1	B	0	0	0	0	0	0	X	0	0	X	0,8
2	Bayar	0	X	0	0	X	X	0	0	0	X	0,6
3	Cinta	0	0	0	X	0	0	0	0	0	0	0,9
4	Doa	0	0	0	0	0	0	0	0	0	0	1,0
5	G	0	0	0	0	0	X	0	0	0	0	0,9
6	K	0	0	0	0	0	0	0	0	0	0	1,0
7	L	0	0	0	0	0	0	0	X	0	0	0,9
8	O	0	0	0	0	0	0	0	0	0	0	1,0
9	Pikir	0	0	0	0	0	0	0	0	0	X	0,9
10	Rumah	0	0	0	0	0	0	0	0	X	0	0,9
11	T	0	0	X	0	0	X	0	0	0	X	0,7
12	Takut	0	0	0	0	0	0	0	0	0	0	1,0
13	Telepon	0	X	0	0	0	0	0	0	0	X	0,8
14	Uang	0	0	0	0	0	0	0	0	0	0	1,0
Nilai Rata-rata Akurasi												0,88

Kelas ‘Bayar’ memiliki akurasi yang rendah sebesar 60% (0,6) karena kesalahan klasifikasi dengan kelas ‘Aku’. Gestur kelas ‘Bayar’ memiliki gestur yang mirip dengan gestur kelas ‘Aku’ yaitu telapak tangan tertutup dan menghadap ke badan. Akurasi rendah kelas ‘T’ disebabkan dengan kesalahan klasifikasi dengan kelas ‘Uang’ karena gestur kedua kelas tidak jauh berbeda. Gestur ‘T’ adalah mengepalkan tangan dengan jari telunjuk membentuk 90 derajat dan ditopang dengan jari jempol. Gestur kelas ‘Uang’ adalah mengepalkan tangan dengan jari telunjuk membentuk 90 derajat dan jari jempol diletakkan di atas jari jempol.

Berdasarkan tabel pengujian untuk model klasifikasi citra, dapat dilihat bahwa rata-rata akurasi klasifikasi untuk 15 kelas adalah 88% (0,88). Hasil akurasi tersebut dapat digolongkan cukup baik.

6. KESIMPULAN DAN SARAN

1. Arsitektur model yang digunakan untuk klasifikasi Sistem Isyarat Bahasa Indonesia adalah *EfficientDet-Lite4* yang memiliki dasar arsitektur *Convolutional Neural Network* dengan ekstraksi fitur *Bi-directional Feature Pyramid Network* (BiFPN). Lapisan arsitektur terdiri dari arsitektur *EfficientNet* sebagai jaringan *backbone*, BiFPN sebagai jaringan fitur, dan jaringan tergabung untuk prediksi kelas dan *bounding box*. *Training* model dilakukan dengan *transfer learning* menggunakan model *EfficientDet-Lite4* dengan iterasi 100 *epoch* dan total data 300 citra.
2. Pengujian dilakukan dengan membandingkan kelas sebenarnya dengan kelas yang dideteksi model. Model klasifikasi citra Sistem Isyarat Bahasa Indonesia dengan metode *convolutional neural network* mencapai akurasi 88% untuk 15 kelas.
3. Implementasi aplikasi *Android* dilakukan dengan dasar kebutuhan yang didefinisikan sebelumnya. Implementasi dimulai dengan membuat *high-fidelity mockup* sampai implementasi kode dengan *Dart*.
4. Seluruh kebutuhan fungsionalitas aplikasi *Android* dapat diimplementasikan dan berjalan dengan baik. Hasil dari pengujian dengan uji kasus menunjukkan seluruh fungsionalitas bekerja sesuai dengan kebutuhan yang sudah didefinisikan. Nilai *usability* dari aplikasi IBIS mencapai 90,5 dengan kategori *excellent*.

Penelitian ini masih banyak kekurangan, maka dari itu peneliti memiliki beberapa saran dan rekomendasi untuk peneliti yang ingin mengembangkan sistem IBIS di masa mendatang, antara lain:

1. Menambah jumlah *dataset* untuk untuk pelatihan dan validasi karena pelatihan deteksi objek membutuhkan data yang banyak untuk dapat melakukan deteksi secara akurat dan cepat.
2. Menambah kelas yang dideteksi agar dapat digunakan untuk komunikasi dengan pemilihan kata yang lebih beragam.
3. Mengembangkan fitur yang sudah ada dan menambahkan fitur baru pada aplikasi IBIS untuk mendorong pengguna belajar bahasa isyarat.

7. DAFTAR PUSTAKA

- A. MADHURAM, M, K. AND M.P., 2019. Cross Platform Development using Flutter. *International Journal of Engineering Science and Computing*, 9(4).
- DHAWARE, C. AND WANJALE, K.H., 2016. Survey On Image Classification Methods In Image Processing. *International Journal of Computer Science Trends and Technology*, 4(3).
- ECMA INTERNATIONAL, 2013. JSON. [online] Available at: <<https://www.ecma-international.org/publications-and-standards/standards/ecma-404/>> [Accessed 17 Oct. 2021].
- FATMI, R., RASHAD, S. AND INTEGLIA, R., 2019. Comparing ANN, SVM, and HMM based Machine Learning Methods for American Sign Language Recognition using Wearable Motion Sensors. In: 2019 IEEE 9th Annual Computing and Communication Workshop and Conference, CCWC 2019.
- GAVALI, P. AND BANU, J.S., 2019. Deep Convolutional Neural Network for Image Classification on CUDA Platform. In: *Deep Learning and Parallel Computing Environment for Bioengineering Systems*.
- HOQUE, O.B., JUBAIR, M.I., ISLAM, M.S., AKASH, A.F. AND PAULSON, A.S., 2019. Real Time Bangladeshi Sign Language Detection using Faster R-CNN. In: *2018 International Conference on Innovation in Engineering and Technology, ICIET 2018*.
- JALLED, F. AND VORONKOV, I., 2016. Object Detection using Image Processing. [online] Available at: <<http://arxiv.org/abs/1611.07791>> [Accessed 17 Aug. 2021].
- KEMENTERIAN PENDIDIKAN DAN BUDAYA REPUBLIK INDONESIA, 2021. Kamus SIBI. [online] Available at: <<https://pmpk.kemdikbud.go.id/sibi/>>.
- KEMENTERIAN SOSIAL REPUBLIK INDONESIA, 2021. Sistem Informasi Management Penyandang Disabilitas. [online] Kementerian Pendidikan dan Budaya Republik Indonesia. Available at: <<https://simpd.kemensos.go.id/>>.
- LI, S., 2020. TensorFlow Lite: On-Device Machine Learning Framework. *Jisuanji Yanjiu yu Fazhan/Computer Research and Development*, 57(9).
- O'SHEA, K. AND NASH, R., 2015. An Introduction to Convolutional Neural Networks. [online] Available at: <<https://arxiv.org/abs/1511.08458v2>> [Accessed 17 Aug. 2021].
- PHUNG, V.H. AND RHEE, E.J., 2019. A High-Accuracy Model Average Ensemble of Convolutional Neural Networks for Classification of Cloud Image Patches on Small Datasets. *Applied Sciences* 2019, Vol. 9, Page 4500, [online] 9(21), p.4500. Available at: <<https://www.mdpi.com/2076-3417/9/21/4500/htm>> [Accessed 22 Dec. 2021].
- RACHARDI, F., 2020. Deteksi Gambar Gestur Kosakata Bahasa Isyarat Indonesia dengan Convolutional Neural Network. [online] Universitas Islam Negeri Syarif Hidayatullah Jakarta. Available at: <<https://www.repository.uinjkt.ac.id/dspace/handle/123456789/56075>>.
- SETYONO, N.F.P. AND RAKUN, E., 2019. Recognizing word gesture in sign system for Indonesian language (SIBI) Sentences using DeepCNN and BiLSTM. In: *2019 International Conference on Advanced Computer Science and Information Systems, ICACISIS 2019*.
- SUHARJITO, THIRACITTA, N. AND GUNAWAN, H., 2021. SIBI Sign Language Recognition Using

Convolutional Neural Network Combined with Transfer Learning and non-trainable Parameters. In: Procedia Computer Science.

TAN, M., PANG, R. AND LE, Q. V., 2019. EfficientDet: Scalable and Efficient Object Detection. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, [online] pp.10778–10787. Available at: <<https://arxiv.org/abs/1911.09070v7>> [Accessed 15 Dec. 2021].