

Implementasi *Deep Learning* menggunakan Algoritma *EfficientDet* untuk Sistem Deteksi Kelayakan Penerima Bantuan Langsung Tunai berdasarkan Citra Rumah di Wilayah Kabupaten Kediri

Muhammad Ferian Rizky Akbari¹, Bayu Rahayudi², Lailil Muflikhah³

Program Studi Teknologi Informasi, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹muhammadferian7@student.ub.ac.id, ²ubay1@ub.ac.id, ³lailil@ub.ac.id

Abstrak

Indonesia merupakan salah satu negara berkembang yang memiliki permasalahan dalam hal kemiskinan. Salah satu upaya pemerintah untuk mengurangi kemiskinan ada membuat sebuah program bernama Bantuan Langsung Tunai. Program ini untuk membantu masyarakat yang masuk dalam kategori miskin atau sedang terkena musibah. Penentuan calon penerima bantuan masih dilakukan secara konvensional melalui sensus penduduk. Salah satu aspek yang menjadi pertimbangan adalah dari kondisi rumah, salah satunya adalah dari kondisi lantai serta kondisi dinding rumah. Saat ini belum ada sebuah sistem yang mampu membantu melakukan klasifikasi calon penerima dari citra rumahnya. Oleh karena itu dibutuhkan sebuah sistem khusus pemrosesan citra rumah untuk membantu pengambilan keputusan berhak atau tidak menerima bantuan dengan pendekatan *deep learning*. *Deep learning* merupakan salah satu metode yang cocok untuk melakukan klasifikasi citra dengan ukuran data yang relatif besar. Algoritma yang digunakan adalah *EfficientDet*, karena algoritma ini mampu untuk mendeteksi objek pada gambar atau video dengan akurasi yang tinggi. Data yang digunakan adalah citra rumah yang diambil dari hasil survei wilayah Kabupaten Kediri, yaitu sebanyak 1500 data. Data tersebut diberi label berhak atau tidak berhak menerima bantuan dan dibagi menjadi dua, yaitu 1200 data *training* dan 300 data *test*. Penelitian ini mendapatkan akurasi yang cukup baik dengan rincian sebagai berikut, nilai *accuracy* 93%, nilai *recall* 92%, nilai *precision* 96%, serta nilai *f1-score* 94%. Selain itu, pada penelitian juga dilakukan pengujian performa sistem dengan mengubah nilai dari beberapa parameter serta jumlah dataset yang digunakan pada saat proses *training* untuk melihat efeknya terhadap hasil akurasi.

Kata kunci: Bantuan Langsung Tunai, citra rumah, *deep learning*, *EfficientDet*.

Abstract

Indonesia is one of the developing countries that has problems in terms of poverty. One of the government's efforts to reduce poverty is to create a program called Direct Cash Transfer. This program is to help people who fall into the poor category or are affected by a disaster. The determination of potential beneficiaries is still done conventionally through a population census. One aspect that is taken into consideration is the condition of the house, one of which is the condition of the floor and the condition of the walls of the house. Currently, there is no system that can help classify prospective recipients from the image of their house. Therefore, a special system for processing house images is needed to help make decisions on whether or not to receive assistance with a deep learning approach. Deep learning is one method that is suitable for image classification with relatively large data sizes. The algorithm used is *EfficientDet*, because this algorithm is able to detect objects in images or videos with high accuracy. The data used is an image of a house taken from the survey results of the Kediri Regency area, which is a total of 1500 data. The data is labeled as entitled or not entitled to receive assistance and is divided into two, which are 1200 training data and 300 test data. This research obtained a fairly good accuracy with the following details, accuracy value 93%, recall value 92%, precision value 96%, and f1-score value 94%. In addition, the research also tested the performance of the system by changing the values of several parameters and the number of datasets used during the training process to see the effect on accuracy results.

Keywords: Direct Cash Assistance, house image, deep learning, *EfficientDet*.

1. PENDAHULUAN

Indonesia adalah negara berkembang yang termasuk dalam salah satu negara dengan penduduk terbanyak di dunia. Maka tidak heran kemiskinan menjadi salah satu permasalahan utama. Jika dilihat pada Data Terpadu Kesejahteraan Sosial (DTKS) menunjukkan bahwa pada tahun 2015 penduduk miskin di Indonesia berada pada angka 11.13%. Namun dengan seiringnya waktu angka tersebut semakin menurun, pada tahun 2021 persentase penduduk miskin di Indonesia sebesar 9,71%. Hal ini menunjukkan bahwa pemerintah serta masyarakat Indonesia saling bekerja sama untuk menjadi lebih baik.

Pada Data Terpadu Kesejahteraan Sosial (DTKS) juga terdapat data sekaligus persentase dari setiap Provinsi maupun Kota/Kabupaten di seluruh Indonesia. Provinsi Jawa Timur masih memiliki persentase penduduk miskin yang cukup tinggi yaitu sebesar 11,4%. Untuk wilayah Kabupaten Kediri masih terbilang tinggi yaitu di angka 11,64%. Berdasarkan data tersebut bisa dikatakan masih banyak penduduk Kabupaten Kediri yang tergolong miskin serta membutuhkan perhatian lebih dari pemerintah agar tetap hidup berkecukupan.

Pemberian BLT biasanya didasarkan pada kriteria tertentu, seperti jumlah penghasilan, status sosial, usia, dan lokasi. Penerima BLT harus memenuhi syarat dan ketentuan yang telah ditetapkan oleh pemerintah, seperti tidak memiliki penghasilan tetap atau terdaftar sebagai penerima bantuan sosial lainnya (Sofi, 2021).

Salah satu bentuk pertimbangan untuk mendapatkan BLT adalah melihat kondisi rumah. Menurut Permendesa PDFT No 6 Tahun 2020, ada 14 kategori untuk sebuah rumah dikatakan layak menerima bantuan. Pada penelitian ini berfokus pada dua kategori saja, yaitu dilihat dari kondisi lantai serta kondisi dinding. Pertama adalah jenis lantai bangunan, apakah masih berupa tanah/kayu dengan kualitas buruk/semen murahan. Kedua adalah jenis dinding yang masih menggunakan anyaman bambu/kayu dengan kualitas buruk.

Saat ini belum ada penelitian atau sistem yang dapat membantu deteksi calon penerima BLT dari citra rumah. Namun terdapat beberapa penelitian yang berfokus pada penentuan calon penerima BLT, salah satunya adalah penelitian yang dilakukan oleh Anita Sindar Sinaga, Murni Marbun, dan Arjon Samuel Sitio (2021)

menerapkan teknologi informasi untuk menentukan tingkat prioritas calon penerima BLT di Desa Pagar Jati, Kabupaten Deli, Provinsi Sumatera Utara. Aplikasi yang dikembangkan pada penelitian tersebut mampu melakukan perhitungan serta prediksi untuk calon penerima bantuan berdasarkan beberapa aspek. Mulai dari pendidikan, pekerjaan, kesehatan, usia, serta status kepala keluarga. Kelemahan dari aplikasi ini adalah tidak dijelaskan algoritma yang digunakan untuk melakukan prediksi.

Penelitian lain yang dilakukan oleh Umu Habibah dan Miftahurrahma Rosyda (2022) dengan topik yang sama yaitu mengenai penentuan calon penerima bantuan. Algoritma yang digunakan pada penelitian ini adalah AHP-TOPSIS. Algoritma ini dianggap mampu mengoptimalkan proses penentuan keputusan dan menjadikan proses prediksi menjadi lebih praktis. Kriteria yang digunakan dalam proses prediksi adalah status pekerjaan, status kesehatan, serta sedang menerima bantuan sosial atau tidak. Kelebihan dari penelitian ini adalah memiliki akurasi yang tinggi yaitu sebesar 91%. Namun hasil tersebut didapatkan hanya dari 70 data penduduk, artinya tidak ada percobaan dengan data penduduk yang lebih banyak untuk melihat pengaruhnya terhadap akurasi.

Dari uraian diatas penulis memiliki ide untuk membuat sebuah sistem deteksi kelayakan penerima Bantuan Langsung Tunai berdasarkan citra rumah dengan menggunakan deep learning serta algoritma *EfficientDet* untuk wilayah Kabupaten Kediri. Penelitian dengan menggunakan algoritma yang sama dilakukan oleh Shaojun Song, Junfeng Jing, Yanqing Huang, dan Mingyang Shi (2021). Penelitian tersebut mengembangkan sebuah sistem deteksi untuk melihat cacat hasil produksi pakaian di sebuah pabrik.

Algoritma ini mampu memaksimalkan proses deteksi objek pada gambar dengan skala yang beragam dari objek kecil hingga objek besar. Serta dapat dilatih dengan dataset yang relatif kecil dan kemudian diterapkan pada dataset yang lebih besar. Dengan beragamnya ukuran gambar serta jumlah yang cukup banyak, maka *EfficientDet* merupakan salah satu algoritma yang mampu untuk melakukan deteksi objek pada gambar rumah untuk calon penerima BLT.

2. LANDASAN KEPUSTAKAAN

2.1 Deep Learning

Deep learning adalah bagian dari *machine learning* namun memiliki jaringan tersendiri, disebut *deep* (dalam) karena di dalam struktur algoritmanya memiliki ratusan lapisan jaringan saraf (Wulandari, et al., 2020). Salah satu kelebihan *deep learning* dapat membantu menyelesaikan masalah yang cukup kompleks, seperti kemampuan untuk mengenali sebuah citra, mengenali suara, hingga dapat meniru cara kerja otak manusia melalui saraf tiruan yang ada pada algoritmanya (Purnama, 2021).

Deep learning memiliki fitur utama untuk melakukan ekstraksi pola yang berguna dari data untuk memudahkan model untuk membedakan kelas, fitur ini disebut dengan *Feature Engineering* (Purnama, 2021). Fitur ini mampu menghasilkan prediksi yang akurat, namun fitur ini sulit untuk dipelajari karena banyaknya jenis data yang berbeda sehingga memerlukan metode pendekatan yang berbeda.

2.2 EfficientDet

Memproses data citra dengan jumlah yang relatif besar dibutuhkan sebuah algoritma yang mampu melakukan training dengan efisien serta memiliki akurasi yang baik. Salah satu algoritma yang cocok adalah *EfficientDet*, algoritma ini memiliki kemampuan training yang cukup cepat dan hasil akurasi yang cukup baik algoritma ini mencapai akurasi tertinggi dengan periode pelatihan paling sedikit dalam tugas deteksi objek (Song, et al., 2021). Arsitektur ini mengalahkan YOLO, *AmoebaNet* dengan daya komputasi minimum.

EfficientDet adalah sebuah algoritma yang digunakan untuk melakukan deteksi objek pada gambar atau video. Algoritma ini dikembangkan oleh tim *Google Research* dan merupakan salah satu algoritma yang paling akurat dan efisien saat ini dalam melakukan deteksi objek.

Algoritma *EfficientDet* sendiri dibagi menjadi tiga bagian yaitu bagian *backbone*, *neck*, dan *prediction*. Bagian *backbone* menggunakan algoritma *EfficientNet* untuk melakukan perubahan skala pada input gambar. Bagian *neck* merupakan arsitektur *Bi-directional Feature Pyramid Network* (BiFPN) yang melakukan *sampling* objek untuk setiap tingkat, dimulai dari tingkat P3 sampai P7 dan akan melakukan prediksi objek. Setelah itu, akan

didapatkan sebuah hasil prediksi berupa *box prediction*.

EfficientDet menggunakan teknik *transfer learning* untuk mempercepat proses pembelajaran. *Transfer learning* adalah teknik dimana model yang sudah di-*train* pada satu dataset, kemudian diaplikasikan pada dataset yang lain dengan harapan performa model tersebut akan lebih baik daripada model yang ditrain dari awal pada dataset yang baru. Selain itu, *EfficientDet* juga menggunakan arsitektur yang disebut *Weight Agnostic Neural Networks* (WANN), dimana model tersebut tidak terikat pada bobot yang sudah ditetapkan sebelumnya, sehingga model tersebut dapat menyesuaikan diri dengan dataset yang baru dengan lebih cepat.

2.3 Klasifikasi Citra

Klasifikasi citra merupakan tahap pengelompokan piksel citra dan ditetapkan menjadi sebuah kategori. Tujuan dari tahap ini adalah melakukan duplikasi kemampuan manusia dalam mempelajari sebuah informasi pada citra digital, sehingga citra atau objek dapat diklasifikasikan oleh mesin atau komputer seperti manusia pada umumnya (Wulandari, et al., 2020). Dan *deep learning* adalah salah satu cara untuk menjalankan klasifikasi citra.

2.4 Mean Average Precision

Mean Average Precision (mAP) adalah salah satu metrik evaluasi yang umum digunakan untuk mengukur kinerja model deteksi objek pada dataset COCO (*Common Objects in Context*) (Lin, et al., 2014). *Matrix* ini merupakan perhitungan rata-rata presisi pada setiap level *recall* pada sebuah dataset.

2.5 Confusion Matrix

Confusion matrix merupakan salah satu teknik yang digunakan dalam evaluasi model pembelajaran mesin (Visa, et al., 2011). *Matrix* ini menyajikan hasil prediksi dari model tersebut dibandingkan dengan nilai sebenarnya dari data uji. Setiap baris pada matriks terdiri dari nilai sebenarnya, sedangkan setiap kolom terdiri dari nilai yang diprediksi oleh model.

Confusion matrix terdiri dari empat bagian yaitu *true positive* (TP), *false positive* (FP), *false negative* (FN), dan *true negative* (TN). TP merupakan jumlah data yang benar terprediksi sebagai positif, FP merupakan jumlah data yang salah terprediksi sebagai positif, FN merupakan

jumlah data yang salah terprediksi sebagai negatif, dan TN merupakan jumlah data yang benar terprediksi sebagai negatif.

3. METODOLOGI

Langkah-langkah dalam penelitian ini memiliki 4 tahapan yang digambarkan dengan *flowchart* pada Gambar 1.



Gambar 1. Metode Penelitian

3.1 Penyiapan Dataset

Jenis data yang digunakan adalah data sekunder, yaitu data yang diperoleh secara tidak langsung atau mengambil dari sumber yang sudah tersedia. Dataset yang disiapkan adalah berupa citra rumah yang berjumlah 1500 data. Pertama melakukan *labelling* untuk setiap citra rumah apakah termasuk kategori “berhak menerima” atau “tidak berhak menerima”. Selanjutnya menyamakan ukuran dataset menjadi 720 x 720 *pixels*, serta melakukan *augmentations* agar data citra menjadi lebih tajam dan mudah dideteksi. Proses penyiapan dataset menggunakan aplikasi web *Roboflow* yang menyediakan fitur untuk membuat dataset dengan jenis COCO (*Common Objects in Context*).

3.2 Perancangan Model

Membuat perancangan model yang digunakan untuk mendeteksi objek di dalam suatu citra. Mulai dari instalasi *package*, *training* model, serta *testing* model.

3.3 Implementasi Sistem

Implementasi / *training* algoritma *EfficientDet* dengan menggunakan bahasa pemrograman *Python*. Proses ini menghasilkan sebuah algoritma yang mampu mengenali objek di dalam sebuah citra sekaligus melakukan klasifikasi berdasarkan kelas yang sudah ditentukan, yaitu berhak menerima dan tidak berhak menerima.

3.4 Evaluasi Sistem

Melakukan evaluasi sistem, evaluasi yang dimaksud adalah menghitung nilai akurasi algoritma yang sudah dilatih. Untuk mendapatkan nilai akurasi digunakan dua metode, yang pertama adalah menggunakan evaluasi dari *library EfficientDet* yang digunakan untuk menghitung nilai *mean Average Precision* (mAP). Kedua adalah melakukan perhitungan akurasi dengan *confusion matrix* berdasarkan data tes yang diuji menggunakan algoritma *EfficientDet* untuk mendapatkan nilai *accuracy*, *precision*, *recall*, dan *F1-score*.

4. PERANCANGAN SISTEM

Pada bab ini membahas mengenai rancangan sistem yang akan diimplementasikan pada penelitian. Perancangan yang dilakukan dimulai dari pembuatan alur/*flowchart* sistem yang akan dibuat, serta rancangan pengujian akurasi.

4.1 Algoritma Sistem



Gambar 2. Flowchart Sistem

Sistem deteksi kelayakan penerima Bantuan Langsung Tunai berdasarkan citra rumah yang akan dikembangkan oleh penulis

merupakan salah satu bentuk bantuan dalam mengambil keputusan apakah sebuah rumah berhak atau tidak untuk menerima bantuan. Sistem akan mengenali beberapa objek pada citra, seperti kondisi dinding, serta lantai. Untuk *flowchart* dari alur sistem dapat dilihat pada Gambar 2 yang dimulai dari penyiapan dataset, kemudian melakukan instalasi *package* yang dibutuhkan. Selanjutnya melakukan *training* algoritma *EfficientDet* dengan dataset yang sudah disiapkan. Tahap selanjutnya, melakukan deteksi objek pada citra dengan menggunakan *EfficientDet* yang sudah dilatih. Terakhir melakukan perhitungan akurasi berdasarkan hasil deteksi menggunakan *confusion matrix* serta *mean average precision* (mAP).

4.2 Implementasi Sistem

Pada sub bab ini membahas mengenai implementasi sistem berdasarkan algoritma sistem yang sudah dijelaskan pada Sub Bab 4.1.

4.2.1 Penyiapan Dataset

Jenis data yang digunakan adalah data sekunder, yaitu data hasil survei dari pihak terkait di daerah Kabupaten Kediri. 1500 citra diambil untuk membuat dataset yang digunakan dalam penelitian ini. Penyiapan dataset dilakukan dalam beberapa tahapan, yaitu sebagai berikut.

1. Memilih data citra dan melakukan *upload* data ke website *roboflow* untuk melakukan *labelling*.
2. Melakukan *labelling* dengan label “berhak menerima” dan “tidak berhak menerima”.
3. Melakukan transformasi data yang bertujuan untuk meningkatkan performa model dan mengurangi waktu *training*. Transformasi yang dilakukan adalah sebagai berikut.
 - a. *Auto orient* untuk menghilangkan rotasi citra dan menyamakan urutan piksel.
 - b. *Isolate object* untuk memotong citra sesuai *labelling* dan mengekstraknya menjadi data tersendiri. Hal ini untuk memperjelas detail dari objek yang diberi *labelling*.
 - c. *Resize* untuk menyamakan ukuran gambar dalam satuan piksel. Ukuran yang digunakan adalah 720 x 720 piksel.

d. *Flip* untuk menambahkan variasi citra dengan kondisi diputar secara vertikal dan horizontal

4. Melakukan *export* dataset dalam bentuk file COCO dataset.

Dari tahapan tersebut dibuat 5 variasi dataset yang memiliki perbedaan pada jumlah data latih, yaitu sebagai berikut.

1. Dataset 1 memiliki 1200 data latih dan 300 data tes.
2. Dataset 2 memiliki 1000 data latih dan 300 data tes.
3. Dataset 3 memiliki 800 data latih dan 300 data tes.
4. Dataset 4 memiliki 600 data latih dan 300 data tes.
5. Dataset 5 memiliki 400 data latih dan 300 data tes.

Karena dataset berbentuk folder dengan ekstensi ZIP, sehingga dataset yang digunakan dalam sistem akan dilakukan *splitting* data untuk memisah antara folder data latih dan data tes.

4.2.2 Instalasi Package

Melakukan penyiapan instalasi *package* yang digunakan pada sistem deteksi. Pertama adalah *pycocotools* untuk mengolah dataset. Kedua *opencv-python* untuk melakukan pemrosesan citra. *Tdqm* untuk menampilkan progress bar serta perulangan yang sederhana. *Tensorboard* untuk visualisasi data, serta *matplotlib* untuk *plotting* 2 dimensi.

4.2.3 Training Model

Proses *training* algoritma *EfficientDet* dilakukan dengan menggunakan dataset yang sudah disiapkan. Dataset yang digunakan sudah dilakukan *splitting* data, jadi untuk proses *training* hanya menggunakan data pada folder data latih. Proses ini bertujuan untuk mendapatkan sebuah algoritma *EfficientDet* yang mampu mengenali objek pada citra rumah, serta dapat melakukan klasifikasi berdasarkan objek yang dideteksi.

4.2.4 Deteksi Objek pada Citra

Deteksi objek pada citra dilakukan dengan menggunakan algoritma *EfficientDet* yang sudah dilatih. Keluaran dari proses ini ada sebuah citra rumah dengan hasil prediksi serta klasifikasinya. Hasil ini yang digunakan sebagai pembantu pengambilan keputusan apakah berhak atau tidak untuk menerima bantuan.

4.3 Evaluasi Sistem

4.3.1 Mean Average Precision (mAP)

Pengujian performa model dapat dilakukan dengan menggunakan salah satu file dari *library* yang digunakan. Hasil perhitungan mAP pada dataset menunjukkan seberapa baik model deteksi objek dapat mengenali dan menempatkan kotak pembatas (*bounding box*) yang tepat pada setiap objek yang ada di dalam gambar.

4.3.2 Confusion Matrix

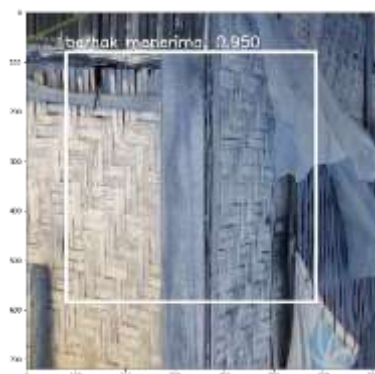
Pengujian performa model terhadap dataset dilakukan dengan menggunakan *confusion matrix*. *Confusion matrix* dapat membandingkan hasil prediksi (klasifikasi) yang dihasilkan dari model dengan hasil klasifikasi yang sebenarnya. Terdapat tiga parameter performa model yaitu *accuracy*, *precision*, dan *recall*.

5. HASIL DAN PEMBAHASAN

Pada sub bab ini akan membahas mengenai hasil pengujian yang sudah dilakukan serta melakukan analisa hasil perhitungan akurasi dari algoritma *EfficientDet* yang sudah dilatih dengan *confusion matrix*. Pada penelitian juga melakukan pengujian perubahan nilai parameter untuk melihat efeknya terhadap hasil akurasi. Parameter yang diujikan adalah jumlah data *training* (latih), pengaruh jumlah *epoch*, pengaruh jenis algoritma *EfficientDet*, pengaruh ukuran *batch*, pengaruh nilai *learning rate*, serta melakukan evaluasi dengan menggunakan *k-fold cross validation*.

5.1 Citra Hasil Deteksi

Algoritma *EfficientDet* yang sudah dilatih digunakan untuk mendeteksi objek pada citra rumah, sekaligus memberikan label apakah citra rumah tersebut berhak atau tidak untuk menerima bantuan. Berikut contoh dari hasil deteksi dapat dilihat pada Gambar 3 dan 4.



Gambar 3. Hasil Deteksi



Gambar 4. Hasil Deteksi

Gambar 3 merupakan salah satu contoh hasil deteksi pada citra yang terklasifikasi berhak menerima bantuan. Karena pada Gambar 3 memiliki dinding rumah yang terbuat dari anyaman bambu dengan kondisi yang buruk. Dan Gambar 4 adalah contoh hasil deteksi citra rumah yang terklasifikasi tidak berhak menerima bantuan. Citra rumah pada Gambar 4 memiliki dinding yang bagus serta sudah di plester

5.2 Perhitungan Akurasi

Dari hasil deteksi pada citra rumah maka didapatkan tabel *confusion matrix* seperti pada Tabel 1.

Tabel 1. *Confusion Matrix* Hasil Pengujian

		<i>Actual Values</i>	
		Berhak Menerima	Tidak Berhak Menerima
<i>Predicted Values</i>	Berhak Menerima	177	7
	Tidak Berhak Menerima	14	102
Total		191	109

Tabel 2 merupakan hasil perhitungan dari Tabel 1, yaitu nilai *accuracy*, *precision*, *recall*,

serta *f1-score*. Sekaligus melakukan evaluasi dengan *library EfficientDet* yang digunakan untuk mendapatkan nilai mAP.

Tabel 2. Hasil Akurasi

Accuracy	Precision	Recall	F1-Score	mAP
93%	96%	92%	94%	91.40%

5.3 Pengaruh Jumlah Dataset

Tabel 3 merupakan hasil perhitungan akurasi dari percobaan menggunakan dataset dengan jumlah data latih yang berbeda. Dimana pada percobaan ini dilakukan *training* sebanyak 50 *epoch*. Dapat disimpulkan dari Tabel 3 bahwa semakin banyaknya data latih maka akan mendapatkan hasil akurasi yang lebih baik.

Tabel 3. Hasil Akurasi Pengaruh Penggunaan Dataset yang Berbeda

Data Latih	Accuracy	Precision	Recall	F1-Score	mAP
1200	91%	96%	89%	92%	91%
1000	92%	94%	93%	93%	92%
800	91%	94%	92%	93%	86%
600	91%	94%	91%	93%	88%
400	89%	92%	90%	91%	89%

5.4 Pengaruh Jumlah Epoch

Epoch merupakan satu kali iterasi dari seluruh data latih yang digunakan pada saat proses *training*. Di setiap *epoch*, data latih digunakan untuk memperbaharui bobot model *EfficientDet*. Berikut adalah hasil perbandingan jumlah *epoch* dari hasil *training*.

Tabel 4. Hasil Akurasi Percobaan Pengaruh Epoch

Epochs	Classification Loss	Regression Loss	Waktu (jam, menit)	mAP
50	0.1170	0.0002	2.3	82.90%
100	0.0752	0.0005	4.24	91.40%
200	0.1082	0.0003	7.50	89.70%
300	0.1371	0.0014	11.24	89.30%

Dari Tabel 4 menggunakan dataset dengan jumlah data latih 1200, menggunakan algoritma *EfficientDet-D0*, *learning rate* 1e-3, serta *batch size* 16. Nilai akurasi mAP tertinggi didapatkan pada percobaan dengan *epoch* sebanyak 100, yaitu 91.40%.

5.5 Pengaruh Jenis EfficientDet

Algoritma *EfficientDet* merupakan salah satu algoritma yang cocok digunakan untuk

melakukan deteksi objek pada sebuah citra maupun video. Pada penelitian dilakukan beberapa percobaan dengan menggunakan bobot *EfficientDet* yang berbeda, yaitu D0, D1, dan D2

Tabel 5. Hasil Akurasi Pengaruh Jenis EfficientDet

EfficientDet	Classification Loss	Regression Loss	Waktu (jam, menit)	mAP
D0	0.1170	0.0002	2.3	82.90%
D1	0.0722	0.0097	3.41	91%
D2	0.0864	0.0054	4.7	85.30%

Dari Tabel 5 menggunakan dataset dengan jumlah data latih 1200, *epoch* 50, *learning rate* 1e-3, serta *batch size* 16. Didapatkan nilai mAP tertinggi pada *EfficientDet-D1*, yaitu sebesar 91%.

5.6 Pengaruh Batch Size

Batch size merupakan jumlah data latih yang diberikan pada saat *training* pada setiap iterasinya. Pengaturan *batch size* dapat mempengaruhi akurasi dari model yang digunakan.

Tabel 6. Hasil Akurasi Pengaruh Batch Size

Batch Size	Classification Loss	Regression Loss	Waktu (jam, menit)	mAP
8	0.1170	0.0002	1.43	82.90%
16	0.0208	0.0005	2.9	88.40%
32	0.1242	0.0048	2.34	92%

Tabel 6 merupakan hasil percobaan dengan menggunakan data latih sebanyak 1200, menggunakan algoritma *EfficientDet-D0*, *epoch* 50, dan *learning rate* 1e-3. Berdasarkan tabel diatas, percobaan dengan menggunakan *batch size* 32 memiliki akurasi mAP tertinggi yaitu 92%. Bisa disimpulkan, semakin besar nilai *batch* maka akurasi mAP akan naik.

5.7 Pengaruh Learning Rate

Parameter ini berfungsi untuk mengoptimalkan proses *training* dengan menentukan ukuran langkah di setiap iterasinya, serta untuk menurunkan *loss function*.

Tabel 7. Hasil Akurasi Pengaruh Learning Rate

Learning Rate	Classification Loss	Regression Loss	Waktu (jam, menit)	mAP
1e-3	0.0208	0.0005	2.9	88.40%
5e-3	0.1229	0.0002	2.10	90.60%

Tabel 7 merupakan hasil percobaan *training* dengan menggunakan data latih 1200, menggunakan algoritma *EfficientDet-D0*, *batch size* 16, dengan *epoch* sebanyak 50. Berdasarkan tabel diatas, *learning rate* $5e-3$ memiliki *classification loss* yang lebih tinggi yaitu 0.12299, namun *regression loss*-nya lebih kecil yaitu 0.00023 serta *mAP* yang lebih bagus yaitu 90.60%.

5.8 K-Fold Cross Validation

Setelah dilakukan pengujian menggunakan beberapa parameter untuk melihat pengaruh terhadap akurasi model, dilakukan pengujian selanjutnya dengan menggunakan *k-fold cross validation*. Metode ini digunakan untuk memvalidasi tingkat akurasi model dengan menggunakan data latih sebanyak 1200, yang akan dibagi menjadi sejumlah k. Pada percobaan ini dataset akan dibagi menjadi sebanyak 10.

Tabel 8. Hasil Pengujian dengan K-Fold Cross Validation

K-Fold	Classification Loss	Regression Loss	Waktu (menit, detik)	mAP
1	0.1180	0.0676	44.50	0.816
2	0.1052	0.1016	46.33	0.860
3	0.0928	0.1079	43.09	0.856
4	0.1036	0.1163	49.01	0.857
5	0.1005	0.1095	46.12	0.860
6	0.1051	0.1093	47.12	0.849
7	0.1067	0.1038	44.16	0.848
8	0.1238	0.1219	46.37	0.891
9	0.1065	0.1182	45.40	0.817
10	0.0922	0.1132	46.49	0.837

Dari Tabel 8, *fold* ke-1 sampai dengan ke-10 memiliki nilai *mAP* yang relatif tidak terlalu jauh. Sehingga dapat disimpulkan bahwa dataset yang digunakan pada saat proses *training* model menghasilkan akurasi yang baik dan konsisten.

6. KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan penelitian yang sudah dilakukan dapat diambil kesimpulan bahwa algoritma *EfficientDet* yang sudah dilatih digunakan untuk mendeteksi objek pada citra rumah dengan jumlah data 300 memiliki akurasi yang tinggi. Dari hasil deteksi tersebut dikelompokkan menjadi 2 kelas, yaitu berhak menerima dan tidak berhak menerima. Selanjutnya melakukan perhitungan *accuracy*, *precision*, *recall*, dan *f1-score* dengan

menggunakan *confusion matrix*. Hasil percobaan dengan jumlah data latih 1200 dan jumlah *epoch* 100 mendapatkan hasil *accuracy* sebesar 93%, *precision* 96%, *recall* 92%, *f1-score* 94%.

6.2 Saran

Adapun saran dalam penelitian ini untuk upaya perbaikan pada penelitian selanjutnya adalah sebagai berikut:

1. Pada penelitian ini adalah hanya menggunakan 1 ukuran citra yaitu 720 x 720 piksel, sehingga ketika menggunakan ukuran citra yang berbeda ada kemungkinan tidak mampu mendeteksi objek pada citra.
2. Menambahkan penggunaan *optimizer* pada saat proses *cross validation* agar model menghasilkan akurasi yang optimal.
3. Penelitian ini dapat dikembangkan lagi menjadi sebuah aplikasi *mobile* maupun web.

7. DAFTAR PUSTAKA

Arumdani, N., Rahmania, S.N., Nafi'ah, Z. & Tukiman, T., 2021. Efektivitas Bantuan Langsung Tunai Dana Desa (BLTDD) di Desa Mojoruntut Kecamatan Krembung Kabupatensidoarjo. *Jurnal Indonesia Sosial Teknologi*, 2(05), pp.874-885.

Dao, S. D. & Marian, R. 2011. Optimisation of precedence-constrained production sequencing and scheduling using genetic algorithms. *Proceedings of the International Multi Conference of Engineers and Computer Scientists*, 16-18 March, Hong Kong.

Danukusumo, K.P., 2017. *Implementasi deep learning menggunakan convolutional neural network untuk klasifikasi citra candi berbasis GPU* (Doctoral dissertation, UAJY).

Habibah, U. & Rosyda, M., 2022. Sistem pendukung keputusan penerima bantuan langsung tunai dana desa di pekandangan menggunakan metode AHP-TOPSIS. *Jurnal Media Informatika Budidarma*, 6(1), pp.404-413.

Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. & Zitnick, C.L., 2014. Microsoft coco:

- Common objects in context. In Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13 (pp. 740–755). Springer International Publishing.
- Pramanik, N.D., 2020. Dampak bantuan paket sembako dan bantuan langsung tunai terhadap kelangsungan hidup masyarakat padalarang pada masa pandemi covid 19. *Jurnal Ekonomi, Sosial & Humaniora*, 1(12), pp.113-120.
- Purmana, A., 2021. Implementasi Metode Deep Learning Dengan Menggunakan Algoritma Convolution Neural Network (CNN) Pada Citra Tulisan Tangan Aksara Sunda.
- Sinaga, A.S.R., Marbun, M. & Sitio, A.S., 2021. Penerapan Teknologi Informasi Penentuan Prioritas Penerima Bantuan Langsung Tunai (BLT) desa Pagar Jati. *Jurdimas (Jurnal Pengabdian Masyarakat) R*, 4(1), pp.65-70.
- Sofi, I., 2021. Efektivitas bantuan langsung tunai dana desa dalam pemulihan ekonomi di desa. *Indonesian Treasury Review: Jurnal Perbendaharaan, Keuangan Negara dan Kebijakan Publik*, 6(3), pp.247-262.
- Song, S., Jing, J., Huang, Y. & Shi, M., 2021. EfficientDet for fabric defect detection based on edge computing. *Journal of Engineered Fibers and Fabrics*, 16, p.15589250211008346.
- Tan, M., Pang, R. and Le, Q.V., 2020. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10781-10790).
- Wulandari, I., Yasin, H. & Widiharih, T., 2020. Klasifikasi citra digital bumbu dan rempah dengan algoritma convolutional neural network (cnn). *Jurnal Gaussian*, 9(3), pp.273-282.