

MODEL DETEKSI JALAN UNTUK SMART GLASSES MENGUNAKAN ALGORITMA YOLO

Moehammad Firly Djulyansyah, Gibtha Fitri Laxmi, Sahid Agustian H

Teknik Informatika, Universitas Ibn Khaldun Bogor
Jl. Sholeh Iskandar Kedungbadak, Kota Bogor, Indonesia
Firlijuliansyah71@gmail.com

ABSTRAK

Kemajuan pengetahuan dan teknologi, khususnya di bidang kecerdasan buatan dan pembelajaran mesin, telah mempermudah penyelesaian masalah dan meningkatkan efisiensi kerja manusia. Dalam konteks ini, model *Smart Glasses* adalah model visi komputer berbasis kecerdasan buatan yang menggunakan algoritma YOLO (*You Only Look Once*) untuk mendeteksi dan mengidentifikasi kondisi jalan berlubang, jalan menanjak, jalan menurun, gundukan, dan genangan air secara *real time*. Penelitian ini memberikan solusi inovatif untuk mendukung dan membantu penyandang tunanetra dalam melakukan aktivitas sehari-hari, khususnya di daerah yang memiliki kondisi jalan buruk sehingga dapat membahayakan keselamatan mereka. Teknologi *deep learning* dan pengolahan citra digital digunakan dalam penelitian ini untuk membuat model deteksi jalan untuk *Smart Glasses* berhasil mencapai tingkat akurasi sebesar 82,8%, membuktikan bahwa model dapat mendeteksi jalan dengan cukup baik. Model dapat dijadikan sebagai alat bantu yang potensial dalam meningkatkan identifikasi kondisi jalan dan keamanan bagi penyandang tunanetra.

Kata kunci: Teknologi, model *Smart Glasses*, algoritma YOLO (*You Only Look Once*)

1. PENDAHULUAN

Kemajuan ilmu pengetahuan dan teknologi telah memicu usaha manusia dalam menyelesaikan masalah dan mengurangi pekerjaan [1]. Selain itu kecerdasan Buatan (*Artificial Intelligence*) juga sedang berkembang pesat, dengan memiliki 3 tipe *machine learning* yaitu *supervised learning*, *unsupervised learning*, dan *reinforcement learning*. *Artificial Intelligence* adalah kemampuan yang dirancang untuk berinteraksi dengan dunia melalui komputer [2]. Keahlian *Artificial Intelligence* khususnya dalam bentuk teknologi *image classification* dan *object detection* dimanfaatkan dalam berbagai aspek kehidupan manusia [3].

Meningkatnya kemajuan teknologi tersebut banyak yang sudah dirasakan oleh masyarakat umum dan juga masyarakat yang berkebutuhan khusus. Dengan melihat bertambahnya masyarakat yang berkebutuhan khusus salah satunya para penderita tunanetra, menurut estimasi data kementerian Kesehatan Indonesia saat ini terdapat 4 juta jiwa tunanetra yaitu sekitar 1,5% dari 270 juta jiwa penduduk [4]. Jalan merupakan infrastruktur darat dan merupakan akses para penyandang tunanetra yang penting sehingga pemantauan dan pemeliharaan kondisi jalan diperlukan untuk mengurangi terjadinya kecelakaan yang disebabkan oleh keadaan jalan yang buruk [5]. Di Indonesia tercatat 52% jalan yang berlubang yang dilansir dari data Bidang Pemberdayaan dan Penguatan Kewilayahan pada tanggal (6/5/2023). Kondisi jalan yang rusak menjadi hambatan bagi para tunanetra yang dapat membuat mereka harus berhati-hati agar tidak terjadi celaka. Maka dari itu dibutuhkan alat bantu agar mereka dapat mengetahui kondisi jalan disebut *smart glasses*. *Smart glasses* adalah model kaca mata pintar yang

berteknologi AI dimana model ini dapat mendeteksi *object*. *Smart glasses* memiliki komponen berupa, kamera, yang dimana dengan komponen ini dapat membantu kelancaran operasi model tersebut.

Deep Learning adalah cabang dari *machine learning* yang menggunakan *Neural Network* (NN) dengan beberapa lapisan dan teknik tertentu, seperti *Restricted Boltzmann Machine* (RBM), untuk mempercepat proses pembelajaran [6]. Dalam *Deep Learning* terdapat metode *Region-Convolutional Neural Network* (R-CCN) dan algoritma YOLO (*You Only Look Once*) merupakan metode dalam mendeteksi objek secara *classifer*, namun pada penelitian sebelumnya metode algoritma YOLO memiliki keunggulan dibandingkan dengan sistem yang berorientasi pada *classifer*, dapat dilihat dari kecepatan dalam memprediksi objek citra saat melakukan *test* dengan prediksi yang diinformasikan secara menyeluruh dibandingkan dengan sistem *Region-Convolutional Neural Network* (R-CCN) dan algoritma YOLO lebih populer [7].

Pada penelitian dalam mendeteksi api pada video menggunakan metode algoritma YOLO dengan nilai *confidence* rata-rata 0,66 dengan nilai terendah 0,55 dan nilai tertinggi 0,71 menghasilkan skor *mAp* sebesar 72,63% [8]. Dan pada penelitian yang dilakukan dalam mendeteksi kerusakan jalan menggunakan pengolahan citra *deep learning* penelitian ini menggunakan perhitungan metode algoritma YOLOv4 tiny dengan menguji objek kerusakan jalan menghasilkan nilai *mAp* sebesar 41% dengan *average loss* 0,6428 dan peneliti menguji menggunakan perhitungan *confussion matrix* dengan *overall accuracy* 88% dan *kappa accuracy* 86% hasil *accuracy* sudah memenuhi standar dari *Congalton and Green* [5]. Sedangkan pada penelitian yang dilakukan

dalam mengklasifikasikan Bahasa isyarat secara *realtime* melalui kamera, video, dan gambar diam menggunakan metode algoritma *You Only Look Once Version 8*. Data yang diperoleh dari 25 relawan dengan mengambil data citra huruf abjad A-Z sebanyak 26 kali, dan menghasilkan tingkat *accuracy* 99,8%, tingkat *precision* 99,4% dan tingkat *recall* 99,8% dengan pengambilan citra pada jarak 100cm menghasilkan *output* berupa *text* dan suara *google Text-To-Speech* [9].

Algoritma YOLO merupakan metode deteksi *object* secara *realtime* dan menunjukkan probabilitas kategori dalam menampilkan kotak pembatas untuk menunjukkan lokasi objek yang terdeteksi. Maka dari itu penggunaan YOLO memiliki kecepatan deteksi objek yang sangat cepat dan akurat [10]. Dalam penelitian ini *object* yang dideteksi yaitu jalan berlubang, jalan yang tergenang air dan jalan menanjak. Sistem kerja pada model *smart glasses* adalah model deteksi kondisi kelayakan jalan menggunakan metode algoritma YOLO.

Penelitian ini bertujuan untuk membuat model identifikasi jalan berlubang, jalan menanjak, jalan menurun, genangan air dan gundukan dengan tujuan dapat menjadi model dasar untuk pengembangan alat *smart glasses*. Sumber informasi dari pengolahan citra digital dapat meningkatkan keberhasilan dalam identifikasi kualitas jalan dan keamananidak an bagi para penyandang tunanetra.

2. TINJAUAN PUSTAKA

Dalam bab kajian teoritis tentang teori-teori yang mendukung tema penelitian. Berikut adalah penjelasan tentang teori-teori yang mendukung penelitian.

2.1. Artificial Intelligence

Kecerdasan buatan atau *Artificial Intelligence* adalah teknik program *computer* yang berfungsi untuk mempelajari data, dan menirukan proses kecerdasan yang dimiliki oleh makhluk hidup dalam menyelesaikan masalah atau persoalan. Dalam perkembangan *Artificial Intelligence* saat ini mengalami perkembangan yang sangat tinggi dan pesat. Tingkat perkembangan *Artificial Intelligence* dilihat dari aspek fungsi kecerdasan manusia seperti persepsi dan kreativitas pada pengetahuan [11].

2.2. Tunanetra

Tunanetra adalah sebuah istilah yang digunakan dalam keadaan suatu individu yang mengalami kelainan atau gangguan fungsi pada indra penglihatan, dan ketidakmampuan dalam menerima informasi secara visual. Tunanetra digolongkan menjadi dua kelompok yaitu kelompok individu yang mengalami kebutaan total (*Blind*) dan individu yang masih mempunyai sisa penglihatan (*Low Vision*). Ketunanetraan ini dapat disebabkan adanya kerusakan pada saraf mata seseorang akan kehilangan fungsi visual untuk merekam objek pada lingkungan sekitar [12].

2.3. Jalan (Infrastruktur)

Jalan merupakan sebuah aspek kebutuhan prasarana yang sangat penting untuk aktivitas sehari-hari karena memudahkan untuk berpindah dari satu lokasi ke lokasi lainnya. Jalan juga sangat berperan dalam menempuh perjalanan sampai tujuan sebagai media infrastruktur transportasi darat dan pejalan kaki [5].

2.4. Citra Digital

Citra digital adalah representasi bentuk pendekatan dari citra yang diambil oleh mesin berdasarkan *sampling* dan kuantisasi (jumlah pada citra yang akan diolah). *Sampling* menggambarkan matriks yang tersusun dari baris dan kolom, dengan kata lain besar kecilnya ukuran pada titik (*pixel*) dapat mempengaruhi besar nilai tingkat kecerahan dalam nilai keabuan (*grayscale*) sesuai dengan digit *biner* dan kuantisasi mewakili warna dalam jumlah gambar [7].

2.5. Deep Learning

Deep Learning adalah teknik dalam *Neural Network* (NN) yang menggunakan teknik tertentu seperti *Restricted Boltzmann Machine* (RBM) untuk mempercepat proses pembelajaran dalam NN yang menggunakan lapis yang banyak atau lebih dari 7 lapis. Dengan adanya DL, waktu yang dibutuhkan saat melakukan *training* akan semakin berkurang karena adanya masalah hilangnya gradien pada propogasi balik akan semakin rendah. Ada beberapa jenis DL antara lain *Deep Auto Encoder*, *Deep Belief Nets*, *Convolutional Neural Network* dan lain-lain. *Deep Learning* dalam proses deteksi objek secara *real time* terdapat dua metode yaitu metode algoritma YOLO dan metode *RCCN* yang memiliki perbedaan pada setiap metode sebagai berikut [13].

2.6. Algoritma YOLO (You Only Look Once)

You Only Look Once adalah sebuah algoritma yang dikembangkan untuk mendeteksi objek secara *real time* menggunakan *repurpose classifier* atau *localizer* dalam melakukan proses pendeteksian pada objek. Sebuah model diterapkan pada sebuah citra di beberapa lokasi dan skala. Daerah dengan citra yang diberi *score* tinggi akan dianggap sebagai sebuah pendeteksian. Algoritma YOLO merupakan metode *regresi* yang dapat memindai seluruh bagian citra dengan melakukan prediksi, mengenali, dan menggolongkan objek pada citra dalam bentuk jaringan saraf tiruan. Jaringan saraf membagi menjadi beberapa letak petak dan melakukan prediksi [14].

2.7. Precision

Precision merupakan kemampuan sebuah model untuk mengidentifikasi *object* yang relevan. Nilai dari *precision* persentase dari prediksi positif yang benar. Jika model mempunyai nilai *precision* yang tinggi, maka model dapat membuat banyak klasifikasi positif dengan nilai yang benar, dan membuat nilai klasifikasi

positif yang salah lebih sedikit. Persamaan untuk mencari nilai *Precision* dapat dilihat pada Persamaan 1 [15].

$$Precision = \frac{True\ positive}{True\ positive + False\ positive} \quad (1)$$

2.8. Recall

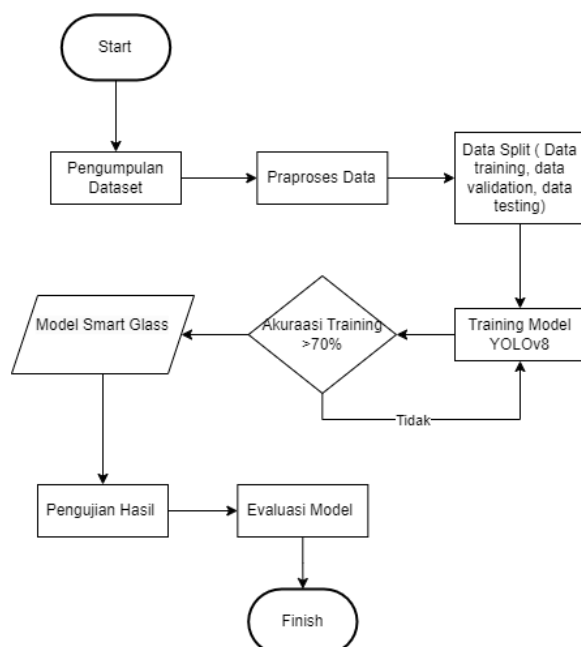
Recall adalah perhitungan rasio prediksi positif benar dibandingkan dengan semua data positif benar. Dalam membandingkannya dengan seluruh objek yang sebenarnya. Persamaan untuk mencari nilai *recall* dapat dilihat pada persamaan 2 [16].

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (2)$$

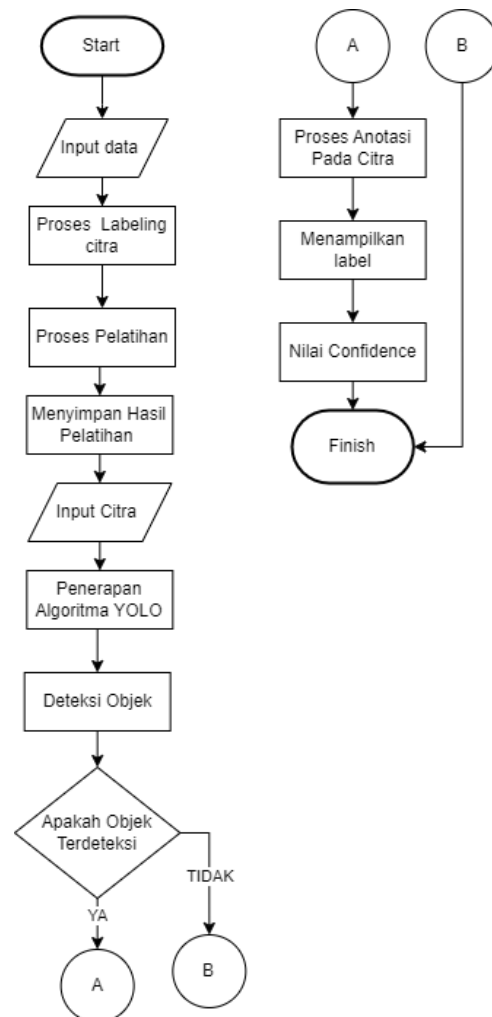
2.9. Roboflow

Roboflow merupakan suatu *preprocessing* data dengan tujuan meningkatkan *dataset* untuk melatih model komputer *Vision*. Dengan *Roboflow*, berbagai transformasi seperti perubahan ukuran, pemotongan, rotasi, dan aplikasi *filter* gambar dapat diterapkan, semua ini sangat penting untuk mengumpulkan informasi yang efektif untuk pelatihan. Selain itu, teknik seperti pembalikan acak, rotasi, dan penyesuaian kecerahan memiliki dampak yang besar dalam meningkatkan kualitas dan jumlah data, yang kritis untuk meningkatkan performa model. Perangkat-perangkat *Roboflow* memberikan kesempatan kepada pengguna untuk meningkatkan kualitas data [17].

3. METODE PENELITIAN



Gambar 1. Flowchart Penelitian



Gambar 2. Flowchart Deteksi Objek

Penelitian ini memerlukan jenis data primer. Berikut data – data yang diperlukan dalam penelitian ini. Data primer meliputi: Survey lapangan di sekitar Kota Bogor. Data primer yang diambil sebanyak 1700 citra dari 5 kelas data yang diteliti. Pada Tabel 1.

Tabel 1. Data Penelitian

No	Kelas	Jumlah Data
1	Jalan Berlubang	340
2	Jalan Menanjak	340
3	Jalan Menurun	340
4	Gundukan	340
5	Genangan air	340

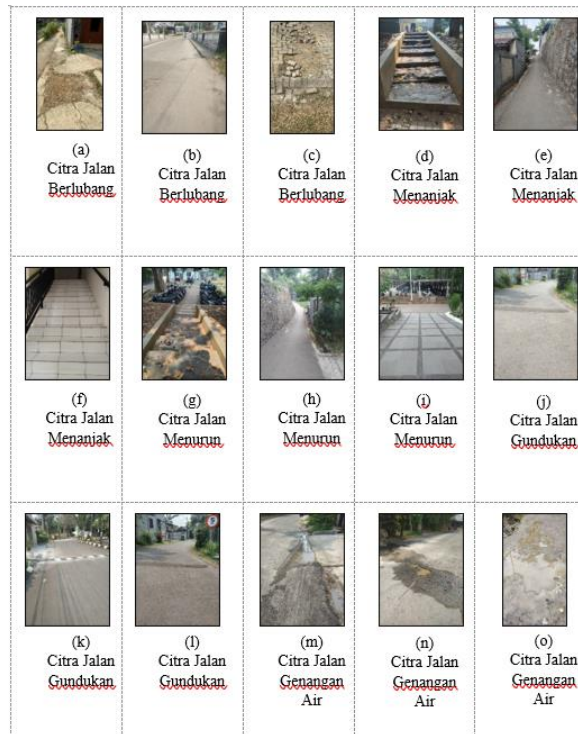
Pada tahap pengumpulan *dataset* menggunakan *device* kamera *iphone 11* dan kamera *handphone Vivo Y12* dengan jarak pengambilan yaitu 1 meter.

Tabel 2. Jumlah Data *Indoor* dan *Outdoor*

Tipe Kelas	Jumlah Data
Data In Door	563
Data Out Door	1137

Pada praproses data penelitian dimaksudkan untuk mempersiapkan data yang digunakan pada penelitian ini agar data yang diperoleh, dapat diproses

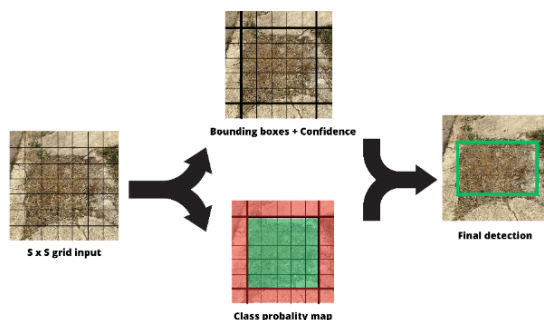
dengan baik oleh model yang akan dibuat menggunakan metode YOLOv8. Dataset citra memiliki resolusi 2268x4032 *pixels* (94 citra menggunakan Iphone 11) dan 3120x4160 *pixels* (1.606 citra menggunakan *handphone* VIVO Y12). Pada Gambar 3.



Gambar 3. Contoh Dataset Kondisi Jalan

Pada proses data *splitting* adalah membagi dataset citra kondisi jalan berjumlah 1700 citra dengan perbandingan 70% *training* sebanyak (1189 citra), 20% data *validation* (337 citra), dan 10% data *testing* (174 citra). Setelah melakukan data *preprocessing* adalah melakukan *training* model yang telah dirancang pada tahap praproses data. Tahap ini merupakan proses data citra kondisi jalan menggunakan YOLOv8, maka didapatkan sebuah model prediksi.

Dilihat pada Gambar 4.



Gambar 4. Architecture YOLO

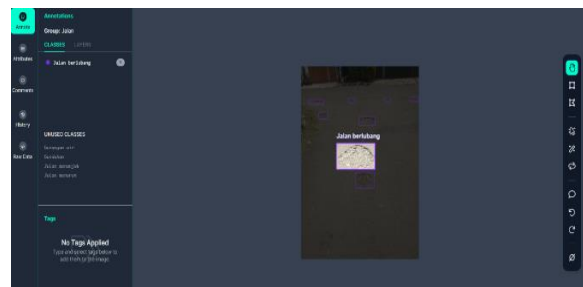
4. HASIL DAN PEMBAHASAN

4.1. Praproses Data

Menentukan dan mengklasifikasi data citra kondisi kelayakan jalan yang diambil oleh dua tipe kamera *handphone* yang akan diklasifikasi menurut kelas data citra yaitu kelas jalan berlubang, jalan menanjak, jalan menurun, gundukan, dan genangan air.

4.2. Anotasi Data Citra Kondisi Kelayakan Jalan

Setelah data citra kondisi kelayakan jalan telah mencukupi, proses yang harus dilakukan yaitu membuat anotasi pada data citra kondisi kelayakan jalan yang sebelumnya telah dikumpulkan, proses ini bertujuan untuk membantu sistem komputer mengenali dan mengklasifikasikannya dalam pemrosesan, analisis, atau pencarian informasi terkait data kondisi kelayakan jalan, dalam proses anotasi ini menggunakan aplikasi *roboflow*.



Gambar 5. Proses Anotasi Dataset

4.3. Data Splitting

Pada tahap ini dilakukan proses data *splitting* bertujuan untuk membagi data hasil anotasi citra agar dapat melihat persentase model yang akan di rancang. Dalam penelitian ini, jumlah citra kondisi kelayakan jalan yang telah disiapkan dengan rasio 70% data *training*, 20% data *validation*, dan 10% data *testing*. Hasil data disajikan dalam bentuk Tabel 3. Pembagian data dilakukan untuk mendapatkan pemahaman dari komputer (model) pada data latih, kemudian untuk mengetahui seberapa baik model terlatih dengan mencoba memvalidasi data model untuk memprediksi kelas data latih.

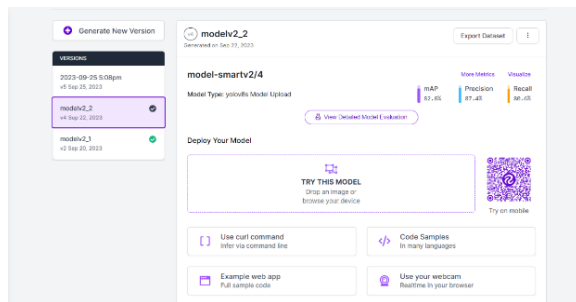
Tabel 3. Hasil Data *Splitting*

Kelas	Training	Validation	Testing
Jalan Berlubang	235	70	35
Jalan Menanjak	239	67	34
Jalan Menurun	238	67	35
Gundukan	238	67	35
Genangan Air	239	66	35

4.4. Generate Dataset

Pengolahan Proses *generate dataset* menggunakan *roboflow* adalah proses untuk membuat versi *generate* model yang sudah dihasilkan dalam proses data anotasi, data *splitting* dan proses *resize* citra data dengan ukuran 640x640 *pixels*. Hasil

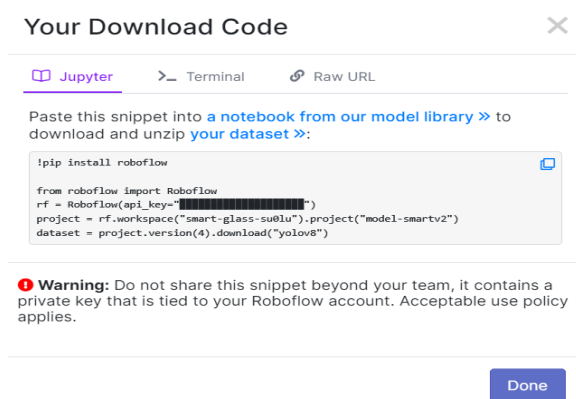
generate model YOLOv8 akan diekspor Dataset Pada Gambar 6.



Gambar 6. Proses Generate Dataset

4.5. Export Dataset

Pada tahap ini hasil dari proses *generate* data dari data anotasi dan data *splitting* akan di *export dataset* ke dalam bentuk berupa *code*, dari *code* tersebut akan digunakan dalam proses deteksi data citra kondisi kelayakan jalan di *Google Collaboratory*. Dilihat pada Gambar 7.



Gambar 7. Export Dataset

4.6. Proses Pelatihan Model YOLOv8

Pada tahap ini penelitian model, penelitian ini menggunakan mesin *virtual google colabs*. *Google Collaboratory* memiliki *GPU* gratis yang dapat digunakan selama 12 jam. Jenis *GPU* dalam mesin ini adalah *Tesla T4*, *RAM* yang disediakan sebesar 12,7 GB dengan disk penyimpanan *virtual* sebesar 24 GB, serta alat penyimpanan dapat ditambah menggunakan *google drive*, fokus utama untuk evaluasi model deteksi dilihat pada hasil nilai *mean average precision (mAp)*. Untuk dimensi *input* model yang digunakan *640x640 pixels*, dimensi ini disajikan dari hasil data citra yang melalui proses *generate* dan *resize*.

Tabel 4. Konfigurasi Parameter Model

Parameter	Konfigurasi
Resize roboflow	640x640
epoch	100
Imgsz	200

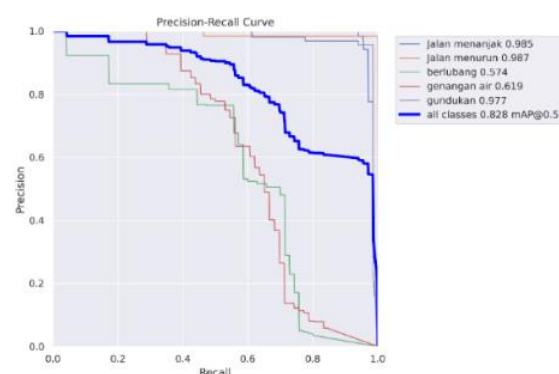
Tabel 4. merupakan parameter dan konfigurasi nilai yang digunakan. Input citra yang digunakan merupakan dimensi 640 x 640 *pixels*. Pada proses

training model *epoch* yang dilakukan sebanyak 100 kali dan *image size* yang digunakan 200 *pixels* agar dapat melakukan proses cepat dan dapat menghasilkan hasil akurasi yang baik maka model secara otomatis menyimpan *checkpoint* dalam file*.pt.

4.7. Evaluasi Model

Dalam mendeteksi objek kelas maka diberikan *IoU* sebagai pembanding antara nilai dari kotak pembatas prediksi dengan kotak pembatas kebenaran dasar. Dari teknik ini memperoleh hasil *confidence score*, jika nilai *confidence score* lebih besar dari *threshold IoU* maka terdapat objek kelas yang di target, namun lebih kecil, maka tidak terdapat objek. Akan tetapi, dalam mendeteksi objek memerlukan akurasi yang mampu menghitung setiap kelas atau seluruh kelas dalam *dataset* citra kondisi kelayakan jalan, untuk itu metode yang dapat menyelesaikan tugas tersebut adalah *mean average precision (mAP)*. Tujuan dari evaluasi ini adalah untuk mengetahui seberapa baik metode yang digunakan dalam menentukan kelas objek berdasarkan kotak pembatas prediksi terhadap kotak pembatas kebenaran. Pada proses mengukur seberapa baik metode dalam mendeteksi objek dengan menghitung nilai *mAP@0,5* sebagai dasar bahwa objek memiliki akurasi yang baik.

Proses mengukur nilai *IoU* dengan nilai *threshold 0,5* menjelaskan terjadi perlakuan "*true positive*" bila nilai prediksi kotak pembatas dan kotak pembatas kebenaran dasar memiliki nilai *confidence* yang lebih besar dari *IoU 0,5* dan menampilkan kelas sesuai dengan target, namun apabila kondisi kotak pembatas prediksi dan *confidence* melewati batas *IoU* tetapi kelas yang terdeteksi tidak sesuai maka kejadian ini disebut sebagai "*false positive*" dan kondisi terakhir apabila tidak menampilkan kotak pembatas prediksi dan *confidence* lebih rendah dari *IoU* maka kondisi ini disebut sebagai "*false negative*".

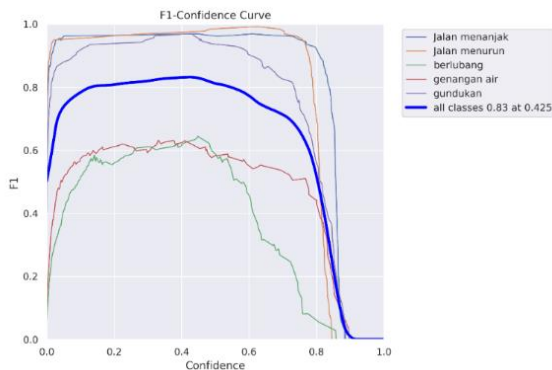


Gambar 8. Hasil Evaluasi *mAp50*

4.8. F1 Confidence Curve

Pada tahap penelitian ini dalam mendeteksi objek melakukan evaluasi *F1 confidence curve* bertujuan untuk melakukan perhitungan pada pengukuran kinerja model klasifikasi, terutama dalam konteks masalah *trade-off* antara keakuratan positif dan negatif dalam kasus klasifikasi biner. Dalam *F1 confidence*

curve juga menggambarkan hubungan antara presisi (*precision*) dan *recall* pada berbagai nilai ambang batas yang berbeda.



Gambar 9. Hasil Evaluasi *F1 Confidence Curve*

Pada Gambar 9. berupa *F1 confidence curve* terdapat 5 kelas data citra, dimana 3 kelas data citra berada pada nilai akurasi yang tinggi yaitu jalan menanjak 0,97, jalan menurun 0,975, dan gundukan 0,969. Sedangkan 2 kelas data citra memiliki akurasi rendah yaitu genangan air 0,6285 dan jalan berlubang 0,6223. Akurasi citra genangan air dan jalan berlubang memiliki nilai rendah dikarenakan kedua citra tersebut memiliki struktur sebab akibat yaitu genangan air timbul disaat adanya jalan berlubang yang tergenang oleh air. Kurva ini digambar dengan mengambil sejumlah ambang batas yang berbeda dan mencatat nilai *precision* dan *recall* untuk setiap ambang batas pada semua kelas data citra.

Tabel 5. Akurasi *F1 Confidence, Precision, Recall*

Kelas Data Citra	<i>Precision</i>	<i>Recall</i>	<i>F1 Confidences</i>
Jalan Menanjak	0,983	0,955	0,97
Jalan Menurun	0,952	1	0,975
Berlubang	0,705	0,557	0,6223
Genangan Air	0,772	0,53	0,6285
Gundukan	0,954	0,985	0,969
Jumlah <i>F1 Confidences</i>	4,1648		
Rata-Rata	0,83		

4.9. Uji Deteksi Citra pada Intensitas Cahaya

Pada Pengujian ini menguji nilai akurasi dari aspek intensitas cahaya, dimana intensitas cahaya merupakan *parameter* dalam menguji nilai akurasi *confidence* pada citra kondisi jalan. Uji citra pada intensitas cahaya terbagi menjadi 2 kategori yaitu uji citra pada kondisi cahaya terang dan uji citra pada kondisi gelap atau kurang cahaya



Gambar 10. Citra Kondisi terang

Pada Gambar 10. Citra dalam kondisi terang dapat mendeteksi objek dengan baik dan menampilkan kotak akurasi pada nilai confidence 0,75.



Gambar 11. Citra Kondisi Gelap

Pada Gambar 11. Citra dalam kondisi gelap atau kurang pencahayaan tidak dapat mendeteksi objek dengan baik dan tidak menampilkan kotak akurasi pada citra jalan tersebut.

4.10. Uji Deteksi Objek Pada Citra

Tahap evaluasi yang telah dilakukan sebelumnya memperoleh data validasi akurasi yang baik. Untuk melihat seberapa baik model maka dilakukan uji deteksi secara langsung objek citra kondisi kelayakan jalan dan pada tahap ini juga membandingkan deteksi objek yang benar dan salah dalam gambar maka dilakukan pengujian pada tahap ini. Pengujian dilakukan menggunakan data uji yang didapat dari data *splitting*. Percobaan deteksi objek secara *realtime* dilakukan langsung di *Google Collaboratory*. Hasil luaran model pada *google collaboratory* ketika mendeteksi citra *realtime* yang digunakan.



Gambar 12. Hasil Deteksi Objek Pada Citra

4.11. Uji Deteksi Objek Pada Video

Pengujian deteksi objek pada video hasil rekaman dilakukan untuk melihat seberapa baik model terlatih dalam menampilkan hasil deteksi secara *realtime*. Kemampuan ini sebagai simulasi visual komputer dalam mendeteksi kondisi nyata di

Lapangan. Video yang digunakan untuk menguji model terlatih berdurasi 1 menit 10 detik, jarak video berkisar 1 sampai 2 meter dengan berjalan pada objek jalan. Percobaan dilakukan pada perangkat laptop dan *handphone* dengan menyambungkan koneksi LAN dengan *Droid Cam*. Dilihat pada Gambar 13.



Gambar 13. Hasil Deteksi Objek Pada Video

5. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian ini didapatkan model pembelajaran yang baik dalam mendeteksi kondisi jalan, dimana hasil pembelajaran terdapat satu model pembelajaran yaitu dengan nilai $mAP@0,5$ sebesar 82,8%, dari hasil evaluasi tersebut kemudian dilihat dari performa deteksi berdasarkan setiap kelas dengan menghitung nilai *confidence score*. Adapun hasil yang diperoleh adalah kelas jalan menanjak, jalan menurun, gundukan dengan nilai yang di dapat lebih dari 96% dan didapat juga hasil yang cukup baik lebih dari 62% pada kelas jalan berlubang dan genangan air. Hasil dari penelitian ini sangat baik karena model sudah dapat membedakan kelas kondisi jalan. Model ini tidak akan menampilkan hasil deteksi jika kondisi jalan tidak masuk dalam lima kelas tersebut. Untuk penelitian selanjutnya diharapkan bisa mencoba deteksi objek yang lain selain objek kondisi kelayakan

jalan untuk menguji metode algoritma YOLOv8, bisa menambahkan lebih banyak data dan bervariasi yang akan dibutuhkan dalam menguji metode algoritma YOLOv8.

DAFTAR PUSTAKA

- [1] R. Muzawi, S. Imardi, Y. Efendi Jurusan Teknologi Informasi, and S. Amik Riau, "SATIN-Sains dan Teknologi Informasi Prototype Kacamata Pemandu bagi Tunanetra dengan Keterbatasan Penglihatan," vol. 6, no. 1, 2020, [Online]. Available: <http://jurnal.sar.ac.id/index.php/satin>
- [2] I. Salamah, M. R. A. Said, and S. Soim, "Perancangan Alat Identifikasi Wajah Dengan Algoritma You Only Look Once (YOLO) Untuk Presensi Mahasiswa," *J. Media Inform. Budidarma*, vol. 6, no. 3, p. 1492, 2022, doi: 10.30865/mib.v6i3.4399.
- [3] A. W. Sutjiadi, K. Gunadi, and L. W. Santoso, "Pengenalan Jenis Masakan Melalui Gambar Menggunakan YOLO," *J. Infra*, 2021, [Online]. Available: <http://publication.petra.ac.id/index.php/teknik-informatika/article/view/11436%0Ahttp://publication.petra.ac.id/index.php/teknik-informatika/article/download/11436/10046>
- [4] Yosua Alvin, "Tim PKM Undip mulai mengembangkan Teknologi Binaural 3D Audio untuk Tunanetra," *Universitas Diponegoro, Departemen Teknik Elektro*, 2022. [https://elektro.ft.undip.ac.id/v3/tim-pkm-undip-mulai-mengembangkan-teknologi-binaural-3d-audio-untuk-tunanetra/#:~:text=Merujuk pada estimasi data dari,pada kisaran 4 juta jiwa.](https://elektro.ft.undip.ac.id/v3/tim-pkm-undip-mulai-mengembangkan-teknologi-binaural-3d-audio-untuk-tunanetra/#:~:text=Merujuk%20pada%20estimasi%20data%20dari,pada%20kisaran%204%20juta%20jiwa.) (accessed Jan. 31, 2024).
- [5] B. Sasmito, B. H. Setiadji, and R. Isnanto, "Deteksi Kerusakan Jalan Menggunakan Pengolahan Citra Deep Learning di Kota Semarang," *Teknik*, vol. 44, no. 1, pp. 7–14, 2023, doi: 10.14710/teknik.v44i1.51908.
- [6] B. School University, "Mengenal Deep Learning Beserta Contoh Penerapannya." <https://sis.binus.ac.id/2023/07/18/mengenal-deep-learning-beserta-contoh-penerapannya/>
- [7] L. Rahma, H. Syaputra, A. H. Mirza, and S. D. Purnamasari, "Objek Deteksi Makanan Khas Palembang Menggunakan Algoritma YOLO (You Only Look Once)," *J. Nas. Ilmu Komput.*, vol. 2, no. 3, pp. 213–232, 2021, doi: 10.47747/jurnalnik.v2i3.534.
- [8] M. L. Nazilly, B. Rahmat, and E. Y. Puspaningrum, "Implementasi Algoritma Yolo (You Only Look Once) Untuk Deteksi Api," *J. Inform. dan Sist. Inf.*, vol. 1, no. 1, pp. 81–91, 2020.
- [9] A. Ma'ruf and M. Hardjianto, "Penerapan Algoritme You Only Look Once Version 8 Untuk Identifikasi Abjad Bahasa Isyarat Indonesia," *Pros. Semin. Nas. Mhs. Fak. Teknol. Inf.*, vol. 2,

- no. 2, pp. 567–576, 2023.
- [10] F. Agustina, “Deteksi Kematangan Buah Pepaya Menggunakan Algoritma YOLO Berbasis Android,” *J. Ilm. Infokam*, vol. 18, no. 2, pp. 70–78, 2022, doi: 10.53845/infokam.v18i2.320.
- [11] A. Arly, N. Dwi, and R. Andini, “Implementasi Penggunaan Artificial Intelligence Dalam Proses Pembelajaran Mahasiswa Ilmu Komunikasi di Kelas A,” *Pros. Semin. Nas.*, pp. 362–374, 2023.
- [12] D. Furwasyih *et al.*, “Audiobook Kesehatan Ibu dan Anak sebagai Media Konseling Kesehatan Ibu dan Anak pada Disabilitas Netra,” *J. Kreat. Pengabd. Kpd. Masy.*, vol. 6, no. 4, pp. 1447–1461, 2023, doi: 10.33024/jkpm.v6i4.8998.
- [13] A. Ammar, A. Koubaa, M. Ahmed, A. Saad, and B. Benjdira, “Vehicle detection from aerial images using deep learning: A comparative study,” *Electron.*, vol. 10, no. 7, pp. 1–31, 2021, doi: 10.3390/electronics10070820.
- [14] Y. E. K. Ignasius Widira Kristianto, “Pendeteksian Sepeda Motor di Jalur Khusus Sepeda Menggunakan Algoritma Pendeteksi Objek YOLO,” *KALBISIANA J. Mhs. Inst. Teknol. dan Bisnis Kalbis*, vol. 8, no. 1, pp. 275–281, 2022.
- [15] E. Tirtana, K. Gunadi, and I. Sugiarto, “Penerapan Metode YOLO dan Tesseract-OCR untuk Pendataan Plat Nomor Kendaraan Bermotor Umum di Indonesia Menggunakan Raspberry Pi,” *J. Infra*, vol. 9, no. 2, pp. 241–247, 2021, [Online]. Available: <https://publication.petra.ac.id/index.php/teknik-informatika/article/view/11454>
- [16] Sandy Eka Putra, Casi Setianingsih, and Ratna Astuti Nugrahaeni, “Deteksi Pelanggaran Parkir Pada Bahu Jalan Tol Dengan Intelligent Transportation System Menggunakan Algoritma Ssd Parking Violation Detection On The Roadside Of Toll Roads With The Intelligent Transportation System Using Ssd Algorithm,” *e-Proceeding Eng.*, vol. 9, no. 3, pp. 1064–1069, 2022.
- [17] P. Kakani and S. Vyas, “AUTOMATED CATALOG GENERATION USING DEEP LEARNING,” no. 08, pp. 937–944, 2023.