

Pengembangan Sistem Monitoring Perawatan Lanjut Usia Menggunakan Webcam Dan Algoritma YOLOv7

1st Muhammad Syarif
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

syarifmuhammad@student.telkomuniversity.ac.id

2nd Casi Setianingsih
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

setiacasie@telkomuniversity.ac.id

3rd Astri Novianty.
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

astrinov@telkomuniversity.ac.id

Abstrak — Inovasi dalam mengembangkan sistem pemantauan perawatan lanjut usia melalui integrasi webcam dan algoritma YOLOv7 memiliki dampak yang signifikan dalam mengatasi berbagai tantangan yang dihadapi oleh populasi lanjut usia. Ancaman terhadap kesejahteraan dan kesehatan para lansia dapat meningkat karena minimnya pengawasan, terutama dalam menghadapi keterbatasan fisik dan mental yang dapat menghambat kemandirian mereka dalam menjalankan aktivitas sehari-hari. Oleh karena itu, diperlukan solusi inovatif yang mampu memberikan pemantauan yang lebih efisien terhadap kondisi mereka. Riset ini fokus pada pengembangan sistem yang dapat mengidentifikasi kejadian jatuh pada lansia serta individu non-lansia, dengan memanfaatkan algoritma YOLOv7. Hasil uji coba sistem ini mengungkap tingkat akurasi yang tinggi, mencapai tingkat keberhasilan sebesar 90,55% dalam mendeteksi kejadian jatuh pada kelompok lansia dan non-lansia. Oleh karena itu, perkembangan sistem pemantauan ini memiliki potensi untuk meningkatkan perawatan dan kualitas hidup lansia, sambil memberikan dukungan berharga bagi semua pihak yang terlibat dalam menghadapi tantangan dalam merawat populasi lanjut usia.

Kata kunci— Deteksi Jatuh, Pemantauan lansia, YOLOv7

I. PENDAHULUAN

Mereka yang telah mencapai usia 60 (enam puluh) tahun ke atas diidentifikasi sebagai populasi lanjut usia atau lansia. Seiring dengan kemajuan fasilitas dan layanan kesehatan, pengendalian tingkat kelahiran, perpanjangan harapan hidup, dan penurunan angka kematian, jumlah dan bagian dari populasi lansia terus bertambah. Dalam kurun lima puluh tahun terakhir, persentase populasi lansia di Indonesia meningkat secara signifikan, mulai dari 4,5 persen pada tahun 1971 menjadi sekitar 10,7 persen pada tahun 2020. Prediksi menunjukkan bahwa angka ini diharapkan terus meningkat hingga mencapai sekitar 19,9 persen pada tahun 2045 [1].

Dampak dari proses penuaan dapat mengakibatkan tantangan dalam berbagai aspek, termasuk fisik, biologis, mental, dan sosial-ekonomi. Seiring bertambahnya usia, umumnya terjadi penurunan terutama dalam hal kemampuan fisik, yang kemudian dapat mengakibatkan hambatan dalam memenuhi kebutuhan sehari-hari. Hal ini bisa mengarah pada

peningkatan tingkat ketergantungan terhadap bantuan orang lain (Fera & Husna, 2018) [2]. Dalam situasi tertentu, kelupaan pada populasi lanjut usia bisa berkembang menjadi gangguan kognitif yang lebih serius. Kelainan ini dapat mengakibatkan penurunan yang signifikan dalam kapasitas kognitif, termasuk kemampuan mengingat informasi, membuat keputusan, dan menjalani aktivitas sehari-hari. Kelupaan yang berat bisa menyebabkan kesulitan bagi orang lanjut usia untuk mengenali kerabat, berorientasi pada lingkungan sekitar, dan bahkan mengingat tindakan rutin seperti makan atau minum obat [3].

Masalah yang kerap timbul adalah kesulitan signifikan yang dihadapi oleh anggota keluarga dalam melakukan pengawasan efisien terhadap populasi lanjut usia. Hal ini disebabkan oleh kesibukan masing-masing anggota keluarga yang mengakibatkan kurangnya upaya pengawasan yang intensif terhadap lansia [4]. Mengawasi lansia yang tinggal sendirian di rumah tanpa pengawasan dari orang dewasa yang sehat dan aktif dalam aktivitas harian memiliki peranan penting dalam mencegah risiko yang dapat meningkat secara signifikan. Di samping itu, terdapat sistem pemantauan khusus yang bertujuan untuk mengenali situasi-situasi yang terjadi secara real-time pada lansia [5] dengan menggunakan WebCam.

Suatu strategi yang menarik adalah memanfaatkan teknologi kecerdasan buatan, terutama model YOLOv7 (You Only Look Once), untuk menghadapi hambatan ini. YOLOv7 merupakan salah satu metode pendeteksian objek yang sangat maju dan efektif dalam konteks *Computer Vision* [6]. Dengan penerapan struktur YOLOv7, sistem mampu secara instan mengidentifikasi berbagai objek dan tindakan, seperti kejadian jatuh atau perubahan perilaku yang mencurigakan pada populasi lanjut usia.

Sebagai hasilnya, penelitian ini memusatkan perhatian pada pengembangan sistem pemantauan serta peringatan dini untuk populasi lanjut usia, menggunakan model pendeteksian objek YOLOv7. Dengan menggabungkan teknologi ini ke dalam rutinitas harian lansia, tujuannya adalah untuk meningkatkan mutu pengasuhan dan memberikan bantuan yang lebih optimal kepada keluarga dan pengasuh. Dengan penerapan YOLOv7, diharapkan dapat diciptakan suatu solusi yang mampu mengurangi potensi risiko kecelakaan

serta meningkatkan taraf hidup para lansia dalam masyarakat modern yang terus berkembang.

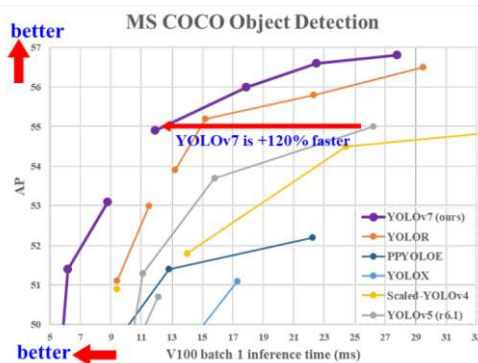
II. KAJIAN TEORI

A. Kondisi Lansia

Secara fisik kondisi tubuh lansia menjadi lebih lemah dibanding masa muda. Sering kali fungsi panca indera lansia mengalami kemunduran. Gerakan motorik kasar maupun halus juga sering kali terganggu. Kondisi ini mengakibatkan para lansia terbatas mobilitasnya. Secara sosial lansia tidak bisa bersosialisasi seperti di masa muda, dan pada kondisi tertentu lansia menjadi tergantung dengan orang lain. Kondisi fisiknya mengharuskan mereka selalu didampingi jika melakukan aktivitas di luar rumah [7].

B. YOLOv7

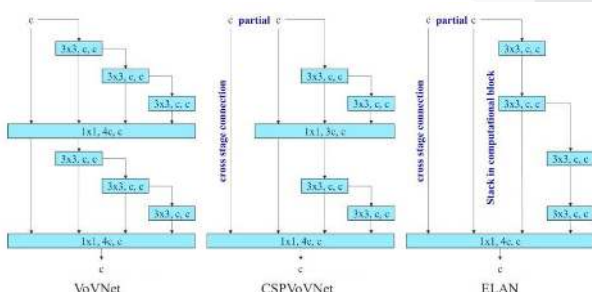
YOLOv7 merupakan sebuah versi model algoritma baru pada 2022 dari seri YOLO. YOLOv7 lebih unggul daripada algoritma pendeteksi objek lain dalam hal kecepatan dan juga akurasi. Penulis dari YOLOv7 mengoptimalkan arsitektur dan mengusulkan untuk pelatihan praktis dan inferensi [6].



GAMBAR 1.

Grafik Perbandingan Antara YOLOv7 dengan YOLO lainnya

Berdasarkan gambar grafik diatas, grafik dari algoritma YOLOv7 lebih cepat serta lebih akurat dibanding dengan algoritma yang lain. Dapat kita lihat algoritma YOLOv7 dibandingkan dengan YOLOv5 (r6.1), bahwa pada sekitar 12 (ms) YOLOv7 memiliki sekitar 55 AP sedangkan YOLOv5 (r6.1) memiliki AP yang sama pada sekitar 26 (ms). Hal tersebut membuat YOLOv7 120% lebih cepat daripada YOLOv5 (r6.1) pada uji coba tersebut.

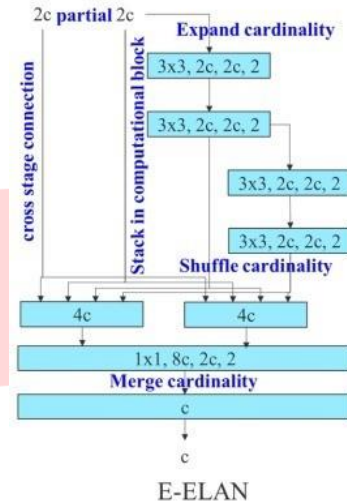


GAMBAR 2.

Model Arsitektur YOLOv7

Diatas merupakan beberapa model dari arsitektur yang diambil dari paper YOLOv7. Terdapat tiga buah model arsitektur yaitu VoVNet, CSPVoVNet, dan ELAN. Dalam

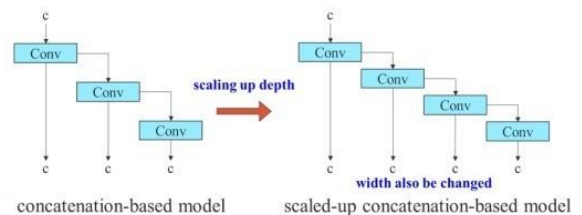
berbagai literatur tentang perancangan arsitektur yang efisien, pertimbangan utamanya tidak lebih dari jumlah parameter, jumlah komputasi, dan kerapatan komputasinya. Desain arsitektur YOLOv7 didasarkan pada model ELAN (*Efficient Layer Aggregation Network*), karena model ELAN mempertimbangkan untuk merancang jaringan yang efisien dengan cara mengontrol jalur gradien terpendek dan terpanjangnya, sehingga jaringan yang lebih dalam dapat belajar dan menyatu secara efektif [6].



GAMBAR 3.

Arsitektur E-ELAN YOLOv7

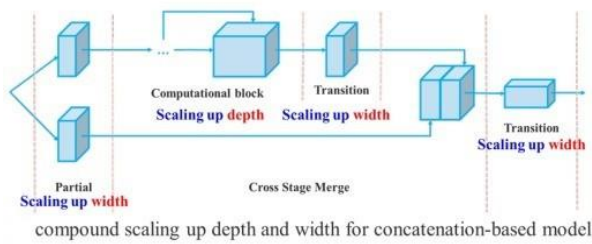
Penulis YOLOv7 memodifikasi arsitektur ELAN, yang disebut dengan E-ELAN (*Extended Efficient Layer Aggregation Networks*). E-ELAN yang diusulkan sama sekali tidak mengubah jalur transmisi gradien dari arsitektur ELAN yang aslinya. Tetapi E-ELAN menggunakan konvolusi grup untuk meningkatkan kardinalitas dari fitur tambahannya dan menggabungkan fitur dari grup yang berbeda secara acak kemudian menggabungkan kardinalitasnya. Cara pengoperasian tersebut dapat meningkatkan fitur yang dipelajari oleh fitur peta yang berbeda dan meningkatkan penggunaan parameter serta perhitungan [6].



GAMBAR 4.

Penskalaan Model YOLOv7

Penskalaan model merupakan sebuah konsep yang penting. Dengan menggunakan penskalaan model, kita dapat meningkatkan kedalaman, resolusi gambar, dan lebar model. Penulis YOLOv7 mengamati bahwa ketika menggunakan penskalaan lebih dalam dari model *concatenation-based*, maka lebar dari keluaran blok komputasi akan meningkat. Hal tersebut menyebabkan lebar dari masukan lapisan transmisi berikutnya juga akan meningkat [6].



GAMBAR 5.

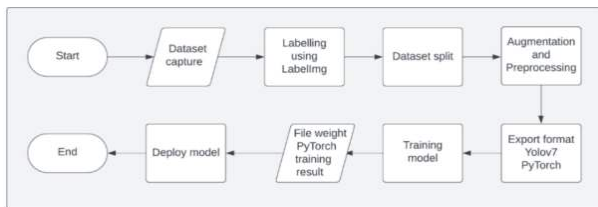
Compound Scaling Up Depth and Width for Concatenation-based Model

Oleh karena itu penulis YOLOv7 mengusulkan model *compound scaling up depth and width for concatenation-based*, model tersebut hanya melakukan pendalaman pada blok komputasi yang diperlukan untuk penskalaan dan sisa dari lapisan transmisinya dilakukan dengan skala lebar yang sesuai [6].

III. METODE

A. Cara Kerja Sistem

Pada penelitian ini, sistem dapat mendeteksi lansia dan non-lansia saat terjadi jatuh dan tidak jatuh.

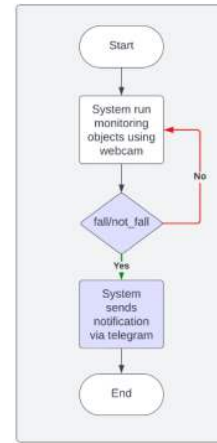


GAMBAR 6.

Skematik Sistem Deteksi Jatuh

Pada gambar 6 merupakan proses dari sistem deteksi jatuh pada lansia. Proses dalam sistem deteksi jatuh pada lansia dimulai dari pembuatan dataset hingga implementasi menggunakan webcam. Tahap pertama adalah mengumpulkan dataset berupa gambar yang selanjutnya akan dilakukan pelabelan menggunakan LabelImg. Selanjutnya, gambar akan diberi label/anotasi menjadi dua class, yaitu *fall* dan *not_fall*. Pada tahap pemisahan dataset, dataset akan dibagi menjadi beberapa bagian, seperti *data training*, dan *data validation*. Setelah itu, dataset akan melalui tahap augmentasi dan *preprocessing*. Dataset akan diekspor dalam format YOLOv7 *Pytorch*. Selanjutnya dataset akan dilatih menggunakan Google Colab, menghasilkan *weight PyTorch* yang nantinya akan diimplementasikan ke dalam sistem yang terhubung dengan webcam.

Berdasarkan skematik yang sudah dijelaskan diatas maka dihasilkan flowchart di bawah ini.



GAMBAR 7.

Flowchart Sistem Deteksi Jatuh

1. Data Collection

Penelitian ini membutuhkan dataset berupa kumpulan gambar dari orang dari posisi berdiri ke posisi jatuh. Dataset yang digunakan berjumlah 1200 gambar. Gambar-gambar yang diambil adalah gambar lansia dan orang biasa, baik secara manual dan mengunduh gambar pada internet.

Data jatuh dan tidak jatuh pada penelitian ini kami kumpulkan dari pengambilan gambar pribadi dan mengunduh gambar pada internet. Gambar yang kami kumpulkan terdiri dari 72 beberapa gerakan serta beberapa posisi seseorang tersebut ketika diambil gambar dan gambar dengan kompleksitas yang tinggi. Berikut contoh gambar *dataset* ditunjukkan pada gambar 9.

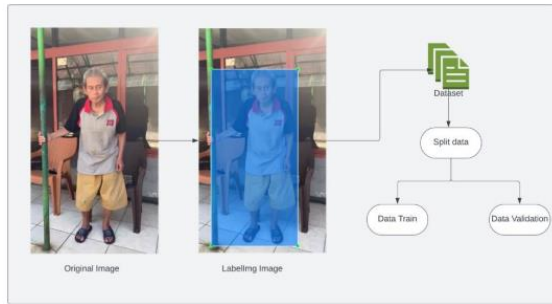


GAMBAR 9.

Contoh Dataset Sistem Deteksi jatuh

2. Data Pre-processing

Pada penelitian ini proses pertama yang dilakukan adalah *pre-processing*. Tahap *pre-processing* dilakukan dengan pengolahan gambar *dataset*. Pada proses dilakukan pelabelan dari kumpulan gambar tersebut dengan menggunakan LabelImg. Pada penelitian ini format anotasi yang digunakan adalah anotasi YOLO. Kemudian hasil dari pelabelan ini disimpan dalam format *.txt*. Pada file *.txt* tersebut terdapat perhitungan indeks diawali dengan ke-0. Dari hasil pelabelan bahwa objek seseorang terdeteksi jatuh (*fall*) berada pada indeks ke-0. Sedangkan objek seseorang terdeteksi tidak jatuh (*not_fall*) berada pada indeks ke-1. Pada angka selanjutnya merupakan *Object coordinate x* dan *y*, *Height*, dan *Width* yang merupakan batas *rectbox*. Skematik tahapan data *pre-processing* dapat dilihat pada gambar 10.



GAMBAR 10.
Pembagian Dataset Deteksi Jatuh

3. Data Processing

Dalam langkah *data processing*, terlebih dahulu dataset yang telah terpilih dipersiapkan untuk melanjutkan proses pelatihan. Google Colaboratory dipilih sebagai platform pelatihan karena kapabilitas komputasi yang diperlukan oleh proses pelatihan YOLOv7 ini sangatlah intensif. Umumnya, YOLOv7 tetap mengadopsi struktur layer konvolusional dalam latihan modelnya. Langkah pelatihan dataset ini melibatkan pembuatan model menggunakan metode transfer learning, dengan memanfaatkan parameter bobot yang telah disediakan oleh pembuat YOLOv7.



GAMBAR 11.
Komponen pelatihan YOLO

Dalam tahap ini, dataset akan ditransformasikan menjadi suatu model yang memiliki kemampuan untuk mengidentifikasi perbedaan antara kategori jatuh dan tidak jatuh. Model akan mengasah pemahaman terhadap pola atau atribut dalam dataset ini melalui serangkaian komponen, termasuk bagian masukan, inti (*backbone*), bagian pengambilan keputusan (*head*), dan akhirnya menghasilkan prediksi melalui bagian *output model*. Rincian dari masing-masing komponen ini diuraikan sebagai berikut:

1. **Input:** Tahap pertama training dataset adalah menginput gambar sebagai objek untuk dinormalisasikan dan diubah ukurannya menjadi 640x640 sebelum masuk kedalam jaringan.
2. **Backbone:** tahap ini melibatkan konversi gambar input menjadi atribut yang akan diolah di bagian "head". Langkah awal melibatkan teknik konvolusi dengan variasi ukuran kernel serta penggabungan atribut dan koneksi pintasan. Langkah ini bertujuan untuk memungkinkan inti (*backbone*) memahami representasi atribut yang semakin mendalam dan abstrak dari gambar.
3. **Head:** Di bagian ini, gambar-gambar dalam dataset yang telah melalui tahap inti sebelumnya akan diperlakukan lebih lanjut untuk menghasilkan prediksi mengenai deteksi objek dalam berbagai rentang spasial.
4. **Output:** Di bagian ini, dataset gambar akan menghasilkan tampilan prediksi label yang mengidentifikasi apakah individu tersebut adalah anggota keluarga atau tamu. Dalam struktur YOLOv7, hasil yang dihasilkan juga akan memberikan informasi tentang jenis objek yang terdeteksi, koordinat batas kotak (*bounding box*), serta skor keyakinan untuk setiap prediksi objek.

4. Validation

Setelah proses *data processing* atau biasa disebut sebagai pembuatan model dilakukan, langkah berikutnya adalah menginisiasi proses validasi model. Proses evaluasi ini bertujuan untuk mengukur tingkat kemampuan model dalam mengenali objek secara akurat dan konsisten. Pada proses ini model akan dievaluasi berdasarkan *Confusion matrix*, *Precision*, *Recall*, *Precision-Recall*, *F1 Score*, dan *mAP*.

Proses evaluasi model YOLOv7 melibatkan tindakan menilai dan mengukur seberapa baik model deteksi yang telah disusun berfungsi. Dalam evaluasi ini, *Confusion Matrix* digunakan sebagai alat utama; sebuah tabel yang membandingkan output prediksi model dengan data asli atau *ground truth*. Tabel ini memuat informasi mengenai *True Positive (TP)*, *True Negative (TN)*, *False Positive (FP)*, dan *False Negative (FN)*. Alat ini memberikan gambaran mendalam tentang seberapa akurat model dalam mengidentifikasi data dan menunjukkan kesalahan yang mungkin dilakukan oleh model. Akurasi, yang didefinisikan sebagai proporsi prediksi benar dari seluruh data, termasuk dalam kelas positif dan negatif, dihitung untuk memberikan gambaran umum tentang kinerja model.

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

Precision mengukur seberapa akurat model dalam mengidentifikasi kelas positif dari semua prediksi yang diklasifikasikan sebagai kelas positif. *Precision* dihitung dengan rumus:

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

Recall mengukur seberapa baik model dapat mengidentifikasi data kelas positif dari keseluruhan data yang sebenarnya adalah kelas positif. *Recall* dihitung dengan rumus:

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

F1 Score adalah rata-rata harmonik antara *Precision* dan *Recall*. *F1-Score* menyediakan keseimbangan antara *Precision* dan *Recall* dan berguna ketika ada ketidakseimbangan kelas atau jumlah *False Negative* yang penting [6].

$$F1\ Score = \frac{2 \times (precision \times recall)}{(precision + recall)}$$

5. Deployment

Setelah model sudah jadi, maka langkah selanjutnya adalah proses *deployment model*. Pada tahap *deployment model*, akan dilakukan pengujian dengan mendeteksi objek jatuh atau tidak jatuh secara *realtime* menggunakan webcam. Hasil dari deteksi tersebut akan dikirimkan ke aplikasi telegram. Berikut adalah contoh hasil *deployment* di bawah ini:



GAMBAR 12.

Realtime Deployment Model Sistem Deteksi Jatuh

B. Hyperparameter Tuning

Setelah proses pelatihan model, akan dilakukan pengujian dengan memodifikasi hyperparameter bawaan YOLOv7. Berikut ini adalah hyperparameter yang akan divariasikan dalam penelitian ini:

1. Learning Rate

Pengujian learning rate bertujuan untuk menemukan nilai learning rate yang optimal untuk melatih model secara efisien dan mencapai akurasi yang tinggi. Learning rate adalah hyperparameter yang menentukan seberapa besar langkah yang diambil oleh optimizer dalam mencari nilai bobot yang optimal selama pelatihan.

2. Optimizer

Pengujian optimizer digunakan untuk mengoptimalkan model pada proses training. Sehingga model dapat dilatih dan menghasilkan model dengan kinerja yang baik. Fungsi lain dari optimizer ialah sebagai meminimalkan nilai fungsi kerugian (loss function). Ada dua jenis optimizer yang digunakan pada pengujian ini, yaitu

a. SGD (Stochastic Gradient Descent):

Ini merupakan teknik optimasi tradisional yang sering diaplikasikan. SGD menentukan gradien dari fungsi kerugian berdasarkan setiap dataset dan memodifikasi bobot dengan mengurangi hasil perkalian antara gradien dengan tingkat belajar.

b. ADAM (Adaptive Moment Estimation):

metode optimasi fleksibel yang menyatukan ide dari RMSprop dan Momentum. Adam menyesuaikan learning rate untuk tiap parameter berdasarkan perkiraan sebelumnya dari momentum gradien serta perhitungan momen kuadrat.

3. Epoch

Pengujian jumlah epoch bertujuan untuk menemukan jumlah epoch yang optimal untuk melatih model dengan mencapai akurasi yang maksimal tanpa terjadi overfitting. Epoch adalah satu iterasi melalui seluruh data latih yang digunakan dalam proses pelatihan.

4. Batch Size

Pengujian batch size bertujuan untuk menemukan ukuran batch yang optimal untuk melatih model dengan mencapai akurasi yang maksimal dan efisiensi pelatihan yang tinggi. Batch size adalah jumlah data pelatihan yang diproses dalam satu iterasi dalam proses pelatihan.

5. Confidence Threshold

Pengujian confidence threshold bertujuan untuk mengevaluasi kinerja model deteksi objek dengan memvariasikan nilai confidence threshold yang digunakan dalam deteksi. Tujuan dari pengujian ini adalah untuk

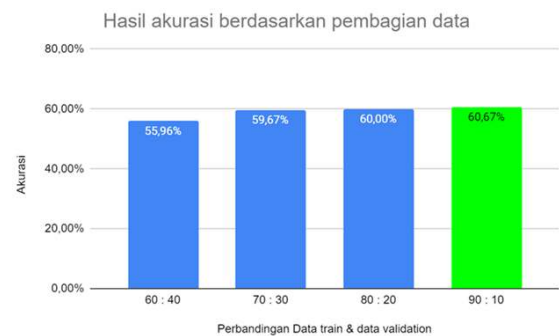
memahami bagaimana performa model berubah ketika threshold diubah, serta menemukan nilai threshold optimal yang memberikan kinerja terbaik.

IV. PENGUJIAN

Pengujian *training model* dimulai dari membagi partisi data, selanjutnya model dievaluasi dengan beberapa *optimizer* yang berbeda, setelah itu dilakukan eksperimen dengan mengubah variabel pada *learning rate*, selanjutnya melakukan pengujian terhadap nilai *epoch*, setelah itu melakukan pengujian terhadap nilai *batch size* dan yang terakhir melakukan pengujian terhadap nilai *confidence threshold*. Berikut adalah rincian dari pengujian dari penelitian ini:

A. Partisi Data

Berikut adalah hasil pengujian akurasi berdasarkan partisi data dengan memanfaatkan *optimizer SGD*, *learning rate* sebesar 0.1, jumlah *epoch* sebanyak 100, *batch size* sebesar 16, dan *confidence threshold* sebesar 0.25.

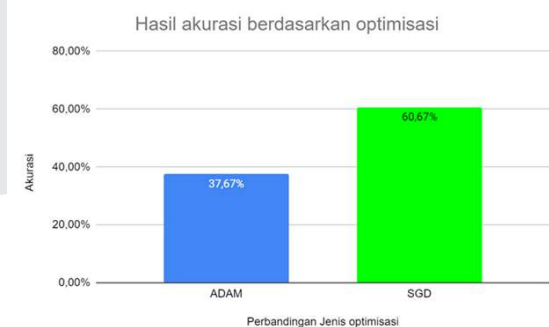


GAMBAR 13.

Grafik Perbandingan Akurasi Antar Partisi Data pada Sistem Deteksi Jatuh

B. Optimizer

Berikut adalah hasil pengujian akurasi berdasarkan *optimizer* dengan memanfaatkan partisi data pelatihan 90% dan data evaluasi 10%, *learning rate* sebesar 0.1, jumlah *epoch* sebanyak 100, *batch size* sebesar 16, dan *confidence threshold* sebesar 0.25.



GAMBAR 14.

Grafik Perbandingan Akurasi Antar Optimizer pada Sistem Deteksi Jatuh

C. Learning Rate

Berikut adalah hasil pengujian akurasi berdasarkan nilai *learning rate* dengan memanfaatkan partisi data pelatihan 90% dan data evaluasi 10%, *optimizer SGD*, jumlah *epoch* sebanyak 100, *batch size* sebesar 16, dan *confidence threshold* sebesar 0.25.

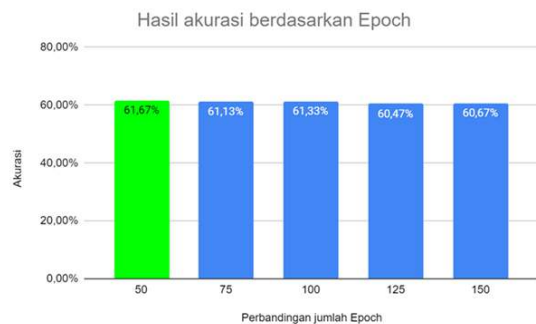


GAMBAR 15.

Grafik Perbandingan Akurasi Antar Learning Rate pada Sistem Deteksi Jatuh

D. Epoch

Berikut adalah hasil pengujian akurasi berdasarkan jumlah *epoch* dengan memanfaatkan partisi data pelatihan 90% dan data evaluasi 10%, *optimizer* SGD, *learning rate* sebesar 0.0001, *batch size* sebesar 16, dan *confidence threshold* sebesar 0.25.

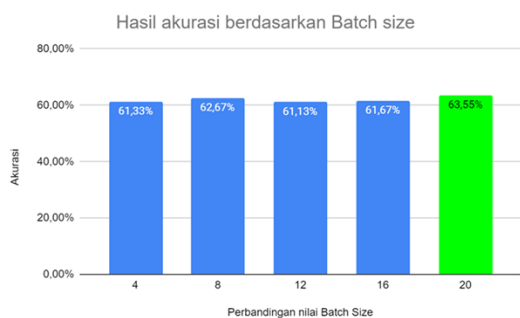


GAMBAR 16.

Grafik Perbandingan Akurasi Antar Epoch pada Sistem Deteksi Jatuh

E. Batch Size

Berikut adalah hasil pengujian akurasi berdasarkan *batch size* dengan memanfaatkan partisi data pelatihan 90% dan data evaluasi 10%, *optimizer* SGD, *learning rate* sebesar 0.0001, jumlah *epoch* sebanyak 50, dan *confidence threshold* sebesar 0.25.

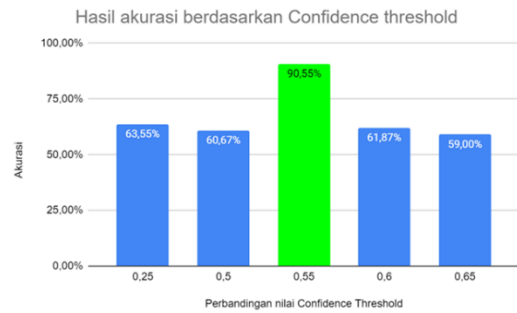


GAMBAR 17.

Grafik Perbandingan Akurasi Antar Batch Size pada Sistem Deteksi Jatuh

F. Confidence Threshold

Berikut adalah hasil pengujian akurasi berdasarkan *confidence threshold* dengan memanfaatkan partisi data pelatihan 90% dan data evaluasi 10%, *optimizer* SGD, *learning rate* sebesar 0.0001, jumlah *epoch* sebanyak 50, dan *batch size* sebesar 20.



GAMBAR 18.

Grafik Perbandingan Akurasi Antar Confidence Threshold pada Sistem Deteksi Jatuh

Dengan demikian pengujian *training model* yang paling optimal adalah dengan menggunakan partisi data pelatihan 90% dan data evaluasi 10%, *optimizer* SGD, nilai *learning rate* sebesar 0.0001, jumlah *epoch* sebanyak 50, nilai *batch size* sebesar 20 dan nilai *confidence threshold* sebesar 0.55.

V. KESIMPULAN

Pembuatan sistem Deteksi Jatuh pada Lansia pada penelitian saat ini mendapatkan hasil akurasi yang paling optimal sebesar 90,55% dengan menggunakan partisi data 90:10, Optimisasi SGD, Learning rate 0,0001, Epoch 50, Batch size 20, Confidence Threshold 0,55. Dan Pengujian kondisi saat deployment realtime mendapatkan akurasi paling optimal dengan ketinggian webcam 210 cm, jarak 3 meter, intensitas Cahaya 264 lux, dan sudut webcam 0 derajat.

REFERENSI

- [1] "Statistik Penduduk Lanjut Usia 2021".
- [2] Fera D, Husna A, "Hubungan Dukungan Keluarga Dengan Kemandirian Lansia Dalam Pemenuhan Aktivitas Sehari-hari Di Desa Alue Tho Kecamatan Seunagan Kabupaten Nagan Raya", [Online]. Available: www.utu.ac.id
- [3] B. Isna Nabila, W. E. Kurniawan, M. Maryoto, F. Kesehatan, and U. Harapan Bangsa, "Gambaran Tingkat Demensia pada Lansia di Rojinhom Ikedaen Okinawa Jepang," Jurnal Ilmiah Indonesia, vol. 2022, no. 8, pp. 671–681, doi: 10.36418/cerdika.v2i8.425.
- [4] A. permana Sanusi, A. Hariyadi, M. Nanak Zakaria, P. Studi Jaringan Telekomunikasi Digital, J. Teknik Elektro, and P. Negeri Malang, "E-ISSN: 2654-6531," 2020.
- [5] H. G. Kim and G. Y. Kim, "Deep Neural Network-Based Indoor Emergency Awareness Using Contextual Information from Sound, Human Activity, and Indoor Position on Mobile Device," IEEE Transactions on Consumer Electronics, vol. 66, no. 4, pp. 271–278, Nov. 2020, doi: 10.1109/TCE.2020.3015197.

- [6] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," Jul. 2022, [Online]. Available: <http://arxiv.org/abs/2207.02696>
- [7] Lukman Nul Hakim and Pusat Penelitian Badan Keahlian DPR RI, "Urgensi Revisi Undang-Undang tentang Kesejahteraan Lanjut Usia," Jurnal Masalah-Masalah Sosial, vol. Volume 11, Jun. 2020, doi: 10.22212/aspirasi.v11i1.1589.

