

Rancang Bangun Sistem Deteksi Label Kardus Berbasis Model Kecerdasan Buatan YOLO dan EasyOCR serta ESP32-CAM

Stefanus Adhie Nugroho

S1 Teknik Elektro, Fakultas Teknik, Universitas Negeri Surabaya.
E-Mail: stefanus.18027@mhs.unesa.ac.id

Nur Kholis, Endryansyah, Farid Baskoro

S1 Teknik Elektro, Fakultas Teknik, Universitas Negeri Surabaya.
E-Mail: nurkholis@unesa.ac.id, endryansyah@unesa.ac.id, faridbaskoro@unesa.ac.id

Abstrak

Sistem Deteksi Label Kardus berbasis model YOLO dan EasyOCR merupakan sistem yang dirancang untuk mendeteksi keberadaan kardus serta tulisan pada label kardus menggunakan model kecerdasan buatan. Masukkan gambar untuk sistem deteksi label berasal dari ESP32-CAM yang mengirimkan gambar ke skrip *Python* via protokol WiFi. Penelitian ini bertujuan untuk menciptakan sistem pendeteksi label kardus yang cepat serta memiliki akurasi yang tinggi. Sabuk konveyor dibangun untuk menciptakan suasana simulasi asli. Adapun spesifikasi utama sabuk konveyor yaitu panjang total sabuk konveyor = 30 cm, dan kecepatan sabuk konveyor = 1,626 cm/s. Berdasarkan hasil penelitian dapat diketahui bahwa waktu rata-rata respon pemrosesan dari ESP32-CAM adalah 0,762 detik. Dalam 40 iterasi pelatihan model YOLO dengan set data kustom, model terbaik yang dihasilkan memiliki loss deteksi objek sebesar 0,00997 dan *mean Average Precision* (mAP) sebesar 0,983. Hasil pendeteksian objek kardus dengan model YOLO yang sudah dilatih menghasilkan rata-rata nilai *confidence* sebesar 58,002% pada gambar kardus berlabel. Model YOLO menggunakan threshold sebesar 0,5 yang memiliki rasio akurasi: error sebesar 19:1 dan hasil deteksi objek non kardus sebanyak 2/20 atau 10%. Setelah model YOLO berhasil mendeteksi kardus berlabel, maka digunakan model EasyOCR untuk membaca kata pada label kardus. Rata-rata nilai *confidence* tertinggi dicatatkan oleh label “jeruk” dengan nilai *confidence* sebesar 0,973364, sedangkan waktu pemrosesan gambar tercepat dicatatkan oleh label “ayam” dengan waktu rata-rata pemrosesan sebesar 11,47 detik. Hasil pembacaan label, nilai *confidence* pendeteksian kardus dan pembacaan label kemudian ditampilkan melalui laman website sederhana untuk mempermudah pengawasan secara *real-time*.

Kata Kunci: Sistem Deteksi Label Kardus, Kecerdasan Buatan, ESP32-CAM.

Abstract

Cardboard Box's Label Detection System based on YOLO and EasyOCR models is a system that is built to detect a cardboard presence and its label using Artificial Intelligence. Image input for the label detection system were fed through an ESP32-CAM camera that are continuously sending images to a Python script using the WiFi protocol. The purpose of this research is to come up with a cardboard's label detection system that are both fast, and highly accurate. Conveyor belt was also built to help the simulation. The conveyor belt's main specifications are its full length of 30 cm and its velocity of 1,626 cm/s. Based on the research conducted it is known that the average image processing time for an ESP32-CAM is 0,762 seconds. The characteristics of the best YOLO model has a detection loss value of 0,00997 and mean Average Precision (mAP) value of 0,983 after 40 epochs of training the YOLO model with custom dataset. The trained custom YOLO model were then being used to detect the cardboard boxes, which resulted in 58,002% of average confidence value across all images of cardboard box with label on top of it. YOLO model used a 0,5 threshold that has a ratio of accuracy: error of 19:1 and a non-cardboard box detection of 2/20 or 10%. After the success of detecting the cardboard boxes with the YOLO model, EasyOCR model was used to read the label. The highest average confidence value was achieved by the “jeruk” label with confidence value of 0,973364, while the fastest average image processing time was achieved by the “sawi” label with average processing time of 7,249 seconds. The results of label reading, confidence value of cardboard box detection and label reading are then displayed in a simple website for easier real time monitoring.

Keywords: Cardboard Box's Label Detection System, Artificial Intelligence, ESP32-CAM

PENDAHULUAN

Indera penglihatan manusia merupakan salah satu harta paling berharga manusia. Pada satu dekade terakhir, banyak topik penelitian yang berupaya untuk merekayasa kemampuan komputer sehingga dapat memproses suatu gambar seperti mata manusia. Subjek ini merupakan topik yang hangat di mata peneliti kecerdasan buatan karena manfaatnya yang sangat banyak pada kehidupan manusia, termasuk manfaat yang berlimpah di sektor industri.

Industri 4.0 menuntut para pelaku industri untuk berinovasi dalam menyelesaikan masalah menggunakan *Internet of Things* (IoT) dan kecerdasan buatan. Salah satu contoh permasalahan yang ada pada industri adalah sistem manajemen gudang. Maraknya situs belanja melalui internet yang menjanjikan pengiriman barang yang cepat menuntut produsen dan distributor untuk memiliki sistem yang cepat serta akurat dalam manajemen Gudang. Berbagai sistem sudah dikemukakan seperti penggunaan algoritma apriori yang dapat menemukan item berbasis pada frekuensi kemunculannya (Chen dkk., 2017), pembuatan aplikasi manajemen gudang yang diintegrasikan dengan *Internet of Things* (Hamdy dkk., 2020), penggunaan sensor RFID dan teknologi Wireless Sensor Network (WSN) (Chang, 2016), dan berbagai inovasi menarik lainnya. Namun, penelitian yang memanfaatkan teknologi kecerdasan buatan, terutama *computer vision* masih belum ada.

Oleh karena itu, diajukanlah sebuah ide untuk membuat sistem pendeteksi label yang ada pada kardus menggunakan *Machine Learning*. Sistem deteksi label kardus dirancang menggunakan 2 buah model kecerdasan buatan yang mengutamakan kecepatan. Model yang dimaksud adalah model pendeteksi objek *You Only Look Once* (YOLO) untuk mendeteksi kardus dan model *Optical Character Recognition* (OCR) bernama EasyOCR untuk mendeteksi label pada kardus.

Model pendeteksi objek YOLO merupakan model kecerdasan buatan bersifat *open-source* yang mengutamakan kecepatan dan efisiensi dalam proses pendeteksian objek (Redmon dkk, 2016). Model YOLO mulai dikembangkan sejak 2016, dan kini model YOLO terbaru yaitu YOLO versi 5 memiliki beberapa keunggulan dibanding generasi YOLO sebelumnya. Kelebihan tersebut adalah kerangka kerja PyTorch yang lebih ramah pengguna, serta mudahnya melatih kumpulan data kustom dibandingkan kerangka kerja sebelumnya yaitu DarkNet., dan kode yang mudah dipahami,

sehingga mempermudah integrasi pada teknologi *computer vision*, serta dapat digunakan sebagai media pembelajaran dan referensi (Jiang dkk., 2021).

Selain YOLO, sistem pendeteksi label kardus juga menggunakan model kecerdasan buatan EasyOCR. Model EasyOCR merupakan model kecerdasan buatan bersifat *open-source*, yang menggunakan algoritma CRNN (*Convolutional Recurrent Neural Network*) (Smelyakov dkk., 2021). EasyOCR dilatih untuk dapat mengenali huruf latin yang ada pada suatu gambar. Berdasarkan kemampuan ini, maka model EasyOCR dipilih sebagai sarana pembaca label yang berada pada kardus.

Instrumen pengukur akurasi pendeteksian kardus yang digunakan adalah *confidence*. *Confidence* merupakan parameter yang digunakan pada sistem kecerdasan buatan untuk mengukur seberapa besar kemungkinan adanya objek pada frame gambar. *Confidence* memiliki besaran dari 0 hingga 1, yang mana 0 merupakan kemungkinan keberadaan objek terkecil dan 1 merupakan kemungkinan keberadaan objek terbesar. Namun demikian, nilai *confidence* 1 tidak berarti bahwa keberadaan objek adalah 100% pasti, begitu pula sebaliknya. Hal ini dikarenakan *confidence* merupakan nilai prediksi yang dihasilkan dari pemrosesan algoritma model kecerdasan buatan, dan umumnya memang nilai prediksi/perkiraan tidak ada yang 100% pasti. Selain *confidence*, pada saat melatih model YOLO dengan kumpulan data kustom, didapatkan beberapa parameter hasil pelatihan seperti nilai *loss* selama pelatihan dan nilai *mean Average Prediction* (mAP). Nilai *loss* pada model YOLO merupakan nilai error yang dihasilkan dari kesalahan prediksi pada saat proses pelatihan. Nilai mean AP (mAP) merupakan parameter yang digunakan untuk mengukur akurasi dari pendeteksi objek setiap kelas dalam suatu basis data (Padilla dkk., 2020).

Sistem pendeteksi label kardus juga dilengkapi sebuah kamera IoT yaitu ESP32-CAM. ESP32-CAM merupakan modul kamera IoT hasil pengembangan dari mikrokontroler ESP32. ESP32-CAM menggunakan chip 32-bit RISC Tensilica Xtensa LX106 yang didalamnya terdapat fitur pemrosesan sinyal digital dan FPU (*Floating Point Unit*). ESP32 memiliki kecepatan clock sebesar 240MHz dan SRAM sebesar 520KB (Dokic, 2020). Modul kamera yang digunakan pada ESP32-CAM adalah OV2640 (Ghifari dkk., 2020).

Rancang Bangun Sistem Deteksi Label Kardus Berbasis Model Kecerdasan Buatan YOLO dan EasyOCR serta ESP32-CAM



Gambar 1. Modul ESP32-CAM
(Sumber: hwlibre.com/id/kamera-esp32/)

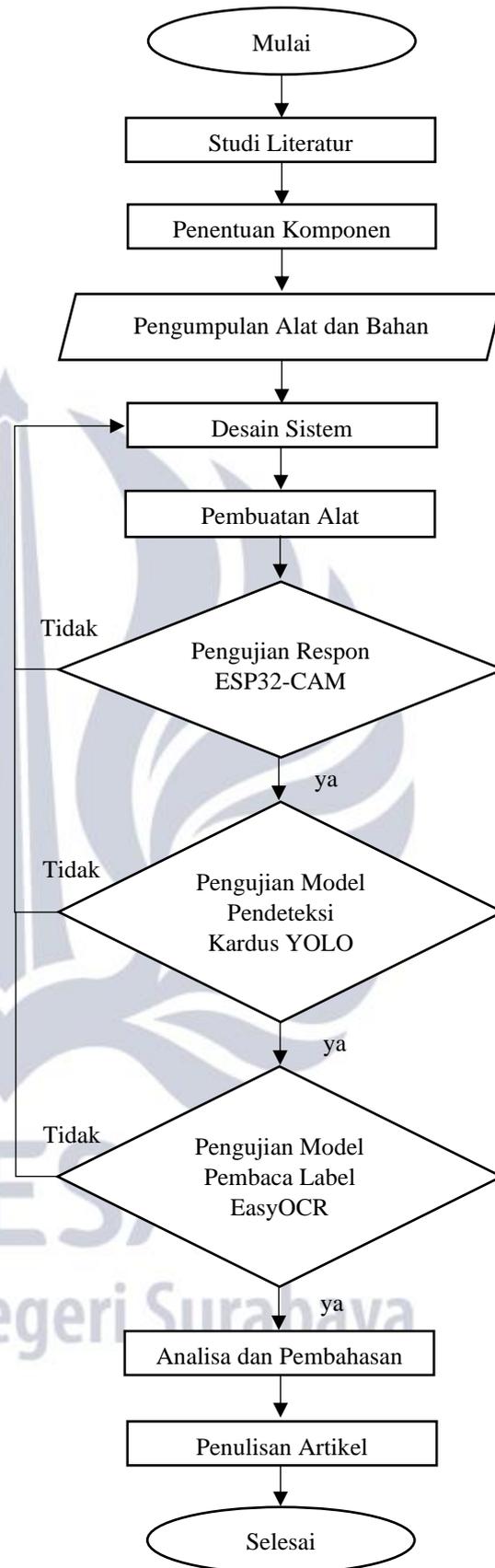
Disamping bagian utama sistem deteksi label kardus yang sudah dijelaskan sebelumnya, penelitian ini ditunjang dengan pembuatan sabuk konveyor sederhana. Pembuatan sabuk konveyor sederhana digunakan untuk mempermudah proses pemindahan kardus dari sisi satu ke sisi lainnya. Komponen sabuk konveyor sederhana ini yaitu berupa stepper motor 28BJY-48 sebanyak 2 buah yang berfungsi untuk menggerakkan sabuk konveyor. Stepper motor 28BJY-48 merupakan motor penggerak 4 fasa yang hanya membutuhkan tegangan masuk sebesar 5 Volt. Untuk mempermudah pemrograman stepper motor 28BJY-48 digunakan driver motor ULN2003. ULN2003 mempermudah konfigurasi antara stepper motor dan Arduino selaku mikrokontroler sabuk konveyor. Modul Arduino Uno R3 merupakan mikrokontroler yang menggunakan chip ATmega 328 (Bahrin, 2017). Pemilihan Arduino Uno R3 sebagai mikrokontroler sabuk konveyor didasari dari kemudahan penggunaan serta konfigurasinya.

Penelitian ini bertujuan untuk menghasilkan sistem pendeteksi label kardus yang memiliki akurasi yang tinggi dalam hal pendeteksian kardus maupun label pada kardus. Selain akurasi, penelitian ini juga diharapkan menghasilkan sebuah sistem yang cepat dan efisien dalam memproses tiap gambar yang masuk.

METODE

Tahapan Penelitian

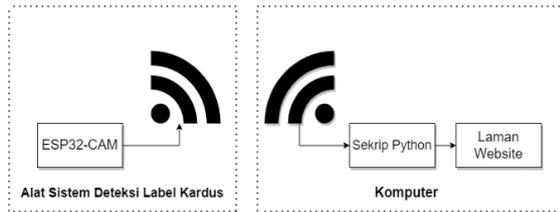
Tahapan penelitian sistem pendeteksi label kardus digambarkan melalui diagram alir Gambar 2.



Gambar 2. Diagram Alir Tahapan Penelitian

Desain Sistem

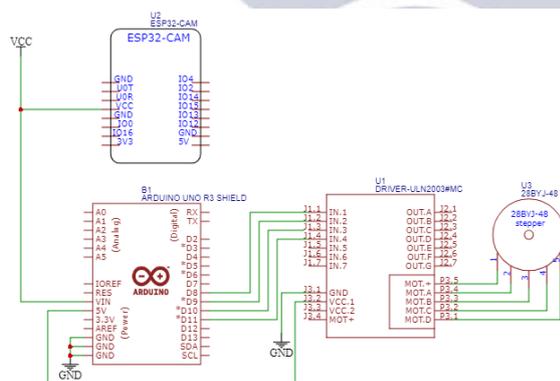
Desain sistem pada alat pendeteksi label kardus divisualisasikan pada Gambar 3.



Gambar 3. Visualisasi Desain Sistem.

Gambar 3 merupakan visualisasi dari desain sistem alat pendeteksi label kardus yang terdiri dari blok alat pendeteksi label kardus dan blok komputer yang digunakan untuk memproses data dari alat. Blok alat sistem deteksi label kardus menggunakan ESP32-CAM sebagai komponen utama yang berfungsi untuk mengirimkan rekaman video secara *real-time* menggunakan protokol WiFi. Blok komputer akan mengolah data berupa rekaman video dari ESP32-CAM dan memproses pendeteksian kardus dan label menggunakan skrip *Python*. Hasil data yang diolah oleh skrip *Python* lalu ditampilkan pada laman website untuk mempermudah pembacaan hasil deteksi label kardus.

Skematik perangkat keras pendeteksi label kardus dapat dipahami secara lebih dalam dengan melihat Gambar 4.

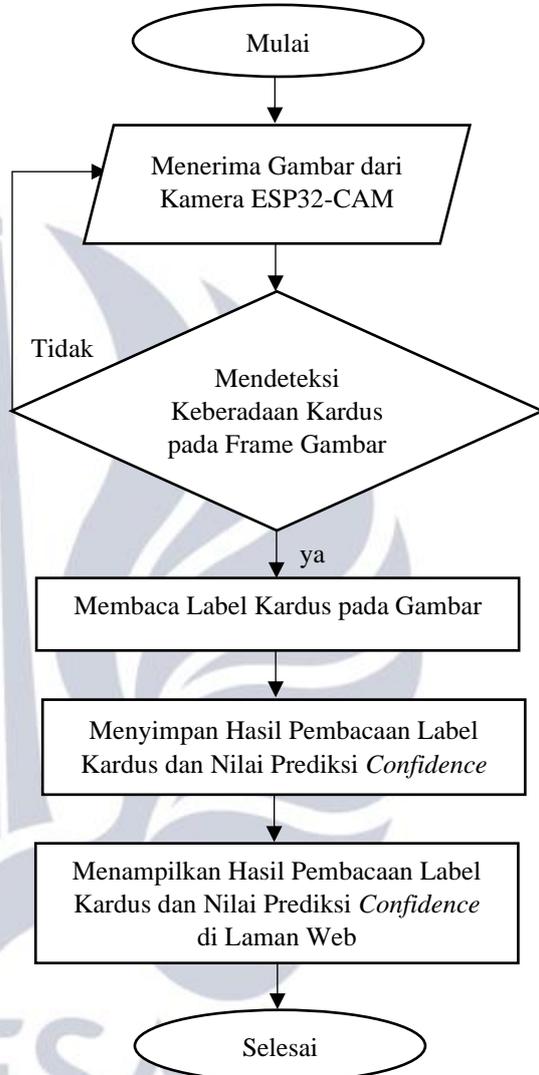


Gambar 4. Skematik Sistem

Skematika sistem deteksi label kardus menunjukkan bahwa ESP32-CAM dan Arduino Uno R3 mendapatkan tegangan masuk yang sama. Driver ULN2003 mendapat masukan digital dari pin D8 hingga D11 Arduino Uno R3 yang masuk ke pin IN1 hingga pin IN4 pada ULN2003 secara berurutan. Tegangan yang masuk ke ULN2003 berasal dari pin 5V Arduino Uno, dan ULN2003

mengatur gerak motor melalui pin motor yang sudah dihubungkan ke pin driver ULN2003.

Simulasi *software* sistem pendeteksi label kardus dapat divisualisasikan melalui diagram alir pada Gambar 5.



Gambar 5. Diagram Alir Simulasi *Software*

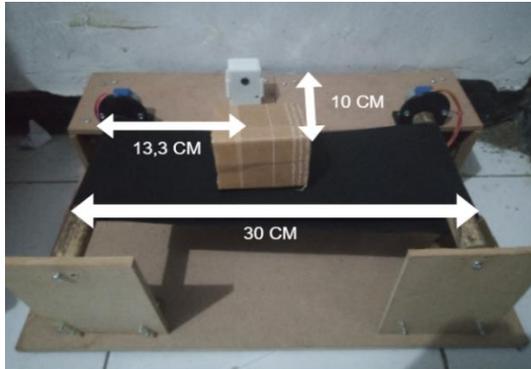
Diagram alir pada Gambar 5 menjelaskan cara kerja *software* pendeteksi label kardus. Simulasi dimulai dengan mengolah rekaman video yang telah diterima dari ESP32-CAM. Ketika sistem mendeteksi keberadaan kardus pada frame video, maka frame tersebut akan diproses dan menjadi masukan sistem pembacaan label kardus. Hasil pembacaan label kardus kemudian akan disimpan bersamaan dengan tingkat kepercayaan pembacaan keberadaan kardus dan label kardus. Hasil keluaran yang telah disimpan kemudian ditampilkan pada laman web untuk memudahkan pembacaan hasil simulasi *software* pendeteksi label kardus.

Rancang Bangun Sistem Deteksi Label Kardus Berbasis Model Kecerdasan Buatan YOLO dan EasyOCR serta ESP32-CAM

HASIL DAN PEMBAHASAN

Pemodelan Sabuk Konveyor

Sabuk konveyor dibuat dengan tujuan menciptakan suasana simulasi yang asli pada sistem deteksi label kardus. Gambar 6 menunjukkan tampilan sabuk konveyor yang sudah dibangun.



Gambar 6. Tampilan Sabuk Konveyor

Sabuk konveyor yang sudah dibangun memiliki beberapa spesifikasi yang dijabarkan pada Tabel 1. Kecepatan sabuk konveyor pada Tabel 1 didapatkan dengan menghitung Persamaan 1.

$$\text{Kecepatan} = \frac{\text{Panjang Sabuk Konveyor (m)}}{\text{Waktu 1 rotasi Sabuk Konveyor (s)}} \dots [1]$$

Tabel 1. Spesifikasi Sabuk Konveyor

No	Besaran	Nilai	Satuan
1	Panjang Sabuk Konveyor	30	cm
2	Jarak Titik Awal Panjang ke ESP32-CAM	13,3	cm
3	Jarak Sabuk Konveyor ke ESP32-CAM	10	cm
4	Waktu Satu Rotasi Sabuk Konveyor	18,45	s
5	Kecepatan Sabuk Konveyor	1,626	cm/s

Spesifikasi sabuk konveyor yang disediakan pada Tabel 1 membantu proses perancangan alat, terutama ESP32-CAM menjadi jauh lebih efisien. Berdasarkan tabel spesifikasi, dapat ditentukan waktu yang dibutuhkan oleh kardus dari titik awal hingga ke ESP32-CAM. Nilai waktu yang dibutuhkan oleh kardus dari titik awal hingga ke ESP32-CAM dapat diketahui dengan menyelesaikan kalkulasi Persamaan 2. Nilai waktu tersebut memungkinkan penambahan delay yang signifikan pada proses penerimaan gambar di

software sehingga sistem jauh lebih ringan karena tidak membutuhkan banyak RAM jika dibandingkan sistem tanpa delay.

$$\text{Waktu} = \frac{\text{Jarak Titik Awal Panjang ke ESP32-CAM}}{\text{Kecepatan Sabuk Konveyor}} \dots [2]$$

Pengujian Respon ESP32-CAM

ESP32-CAM dikonfigurasi untuk mengirim gambar via protokol WiFi. Pengujian respon ESP32-CAM dibutuhkan untuk mengetahui apakah sistem ESP32-CAM dapat bekerja dengan baik di skrip Python. Pada proses pengujian, baik delay pada skrip Python maupun delay pada ESP32-CAM dikonfigurasi menjadi 0 untuk mengetahui nilai delay pada pemrosesan gambar dari ESP32-CAM menuju ke skrip Python. Tabel 2 menunjukkan hasil respon sistem pemrosesan gambar dari ESP32-CAM menuju ke skrip Python. Pengambilan data respon waktu dibantu oleh software Arduino IDE. Respon waktu dihitung dari saat kamera ESP32-CAM mengambil gambar hingga gambar terkirim ke skrip Python. Gambar 7, 8, 9, dan 10 merupakan contoh tampilan gambar yang sudah dikirimkan dari ESP32-CAM ke skrip Python.



Gambar 7. Hasil Tangkapan Gambar Kardus "AYAM"



Gambar 8. Hasil Tangkapan Gambar Kardus "SAWI"



Gambar 9. Hasil Tangkapan Gambar Kardus “JERUK”



Gambar 10. Hasil Tangkapan Gambar Kardus “APEL”

Pengujian Model YOLO

Model YOLO dilatih dengan menggunakan kumpulan data kustom yang mengandung 130 gambar kardus untuk fase pelatihan dan 49 gambar kardus untuk fase validasi. Pengembang model YOLO menyediakan tutorial sederhana yang dapat digunakan untuk melatih kumpulan data kustom melalui Google Colab. Model YOLO dilatih pada aplikasi Google Colab untuk mempermudah proses pelatihan model YOLO karena Google Colab menyediakan GPU virtual yang dapat diakses ketika pengguna sudah terkoneksi dengan server *cloud* Google Colab. Pelatihan model YOLO dilakukan sebanyak 40 epoch/iterasi dengan menggunakan beban buatan YOLO dalam bentuk PyTorch. Gambar 11 merupakan salah satu contoh hasil pelatihan model YOLO dengan kumpulan data kustom. Hasil terbaik pelatihan model YOLO tersimpan pada file *best.pt* yang berbentuk PyTorch. Hasil *best.pt* merupakan modul yang akan digunakan pada simulasi *software* sistem pendeteksi label kardus.



Gambar 11. Hasil Pelatihan Model YOLO

Tabel 2. Hasil Pengiriman Gambar

No	Status Gambar	Waktu
1	Terkirim	0,73 detik
2	Terkirim	0,71 detik
3	Terkirim	0,72 detik
4	Terkirim	0,85 detik
5	Terkirim	0,66 detik
6	Terkirim	0,8 detik
7	Terkirim	0,82 detik
8	Terkirim	0,82 detik
9	Terkirim	0,71 detik
10	Terkirim	0,80 detik
Rata-Rata Respon Waktu		0,762 detik

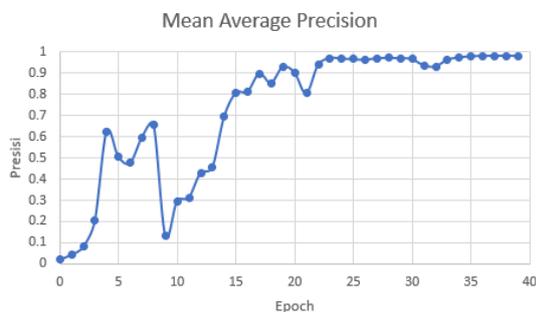
Pelatihan model YOLO dengan kumpulan data kustom sebanyak 40 epoch menghasilkan model yang memiliki loss dalam pendeteksian objek yang rendah. Pada iterasi terakhir, tercatat bahwa model YOLO memiliki loss sebesar 0,00997. Loss paling kecil terjadi pada iterasi 33 yang mencatatkan loss sebesar 0,00985. Gambar 12 merupakan hasil plotting data loss objek deteksi yang dialami saat pelatihan model YOLO. Grafik yang ditampilkan menunjukkan bahwa jumlah iterasi berhubungan erat dengan tingginya loss objek. Berdasarkan grafik, semakin tinggi jumlah *epoch* maka terjadi trend penurunan angka loss objek deteksi saat pelatihan.

Rancang Bangun Sistem Deteksi Label Kardus Berbasis Model Kecerdasan Buatan YOLO dan EasyOCR serta ESP32-CAM



Gambar 12. Grafik Loss Objek Deteksi YOLO

Pelatihan model YOLO dengan dataset kustom sebanyak 40 iterasi juga menghasilkan rata-rata mean presisi yang tinggi. Tercatat bahwa mAP model YOLO adalah sebesar 0,982 dari nilai penuh yaitu 1. Nilai tertinggi mAP terjadi pada iterasi 39 yang mencatatkan nilai mAP sebesar 0,983. Gambar 13 merupakan hasil plot data mAP selama masa pelatihan model YOLO dengan dataset kustom. Grafik memperlihatkan bahwa pada 20 iterasi awal nilai mAP tidaklah stabil. Hal tersebut terjadi karena model YOLO sedang melakukan proses *tuning*. Pada 20 iterasi terakhir, nilai mAP lebih stabil sehingga tidak terjadi banyak peningkatan perkembangan.



Gambar 13. Mean Average Precision (mAP)

Pelatihan model YOLO dengan kumpulan data kustom telah membuahkan sebuah model YOLO yang dapat mendeteksi kardus dengan nilai mAP yang tinggi dan nilai loss yang rendah. Pembuktian keberhasilan proses pelatihan dapat dilakukan dengan mengaplikasikan model YOLO yang sudah dilatih ke skrip *Python*.

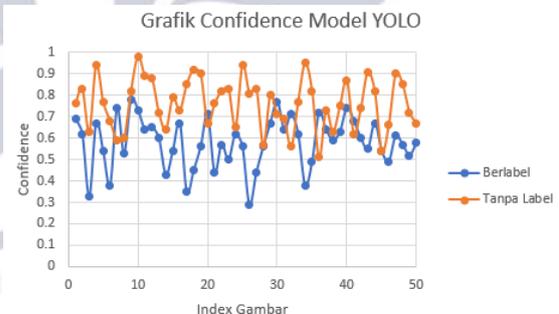
Model YOLO yang sudah dilatih selanjutnya diuji akurasi dengan cara memasukkan 50 gambar kardus dengan label dan 50 gambar kardus tanpa label yang sebelumnya telah ditangkap oleh ESP32-CAM. Gambar 14 menunjukkan grafik hasil *confidence* model YOLO. Berdasarkan Gambar 14, dapat diketahui bahwa gambar tanpa label memiliki nilai bawah yang lebih tinggi dibandingkan dengan

nilai bawah gambar berlabel. Tercatat bahwa nilai bawah gambar berlabel yaitu sebesar 0,29 sedangkan nilai bawah gambar tak berlabel yaitu sebesar 0,51. Pada nilai atas, gambar tanpa label memiliki nilai yang jauh lebih tinggi yaitu sebesar 0,98 sedangkan gambar berlabel memiliki nilai tinggi sebesar 0,78. Rata rata nilai *confidence* dan persentasenya dapat ditemukan dengan menghitung persamaan 3 dan 4.

$$\text{Rata-rata} = \frac{\text{total nilai confidence}}{\text{jumlah gambar}} \dots [3]$$

$$\text{Persentase confidence} = \text{rata} - \text{rata} \times 100 \dots [4]$$

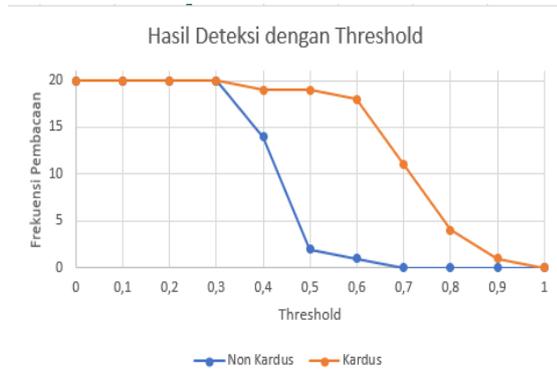
Berdasarkan perhitungan Persamaan 3 dan 4 maka ditemukan persentase rata rata nilai *confidence* pada gambar berlabel sebesar 58,002% sedangkan persentase rata rata nilai *confidence* pada gambar tak label adalah sebesar 76,04%. Terdapat perbedaan sebesar 18,038% pada nilai rata rata *confidence* kedua kumpulan gambar. Diketahui bahwa kumpulan gambar yang digunakan untuk melatih model YOLO merupakan gambar kardus tak berlabel. Oleh karena itu, kumpulan data yang digunakan untuk melatih model tentunya berperan besar terhadap perbedaan nilai yang didapat antara gambar berlabel dan tak berlabel.



Gambar 18. Ploting Data Confidence YOLO

Pengukuran *threshold* selanjutnya dilakukan setelah pengukuran *confidence*. *Threshold* merupakan nilai pembatas yang digunakan untuk menyaring hasil data yang tidak sempurna. Pengukuran *threshold* dilakukan dengan tujuan untuk mendapatkan model YOLO yang dapat diaplikasikan secara ideal dengan nilai error yang rendah. Nilai error yang dimaksud adalah terdeteksinya objek lain selain kardus. Untuk itu telah dilakukan pengukuran untuk mendapatkan *threshold* yang ideal, dengan membandingkan hasil pendeteksian kardus dan objek non-kardus. Gambar 15 merupakan grafik pengukuran *threshold* yang

membandingkan hasil pendeteksian kardus dan non-kardus. Adapun objek non-kardus yang digunakan dalam pengukuran adalah berupa masker, gelas es krim, kepala *charger* telepon genggam, dan lakban. Beberapa objek tersebut dipilih karena kemiripan karakteristiknya dengan kardus.



Gambar 15. Perbandingan Objek Terdeteksi pada Threshold

Gambar 15 merupakan hasil pengukuran pendeteksian objek kardus dan non-kardus pada *threshold* yang ditentukan. Berdasarkan hasil pengukuran, dapat diketahui bahwa *threshold* 0,5 merupakan *threshold* paling ideal dengan rasio akurasi: error sebesar 19:1 dan hasil deteksi objek non kardus sebanyak 2/20 atau 10%. *Threshold* 0,6 juga dapat menjadi opsi yang hampir sama akurasinya dengan rasio akurasi: error sebesar 18:2 dan hasil deteksi objek non kardus sebanyak 1/20 atau 5%. Gambar 16, 17, 18, dan 19 merupakan beberapa hasil pendeteksian kardus dengan model YOLO yang sudah dilatih dengan kumpulan data kustom.



Gambar 16. Hasil Deteksi Label Kardus “AYAM”



Gambar 17. Hasil Deteksi Label Kardus “APEL”



Gambar 18. Hasil Deteksi Label Kardus “JERUK”



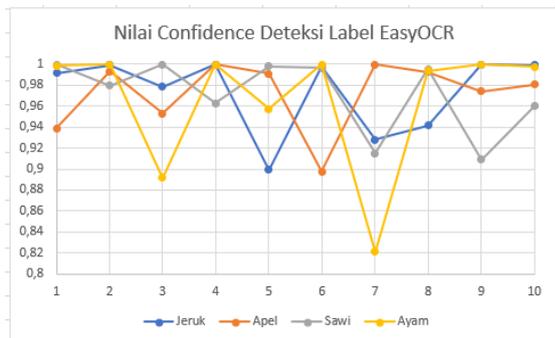
Gambar 19. Hasil Deteksi Label Kardus “SAWI”

Pengujian Model EasyOCR

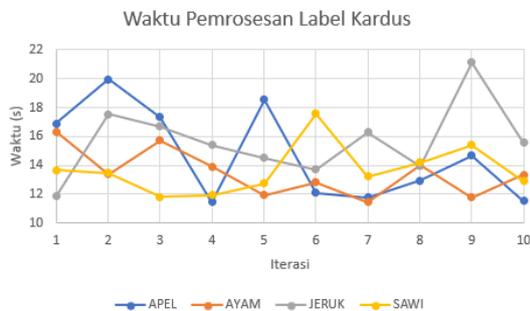
Pengujian model EasyOCR dilakukan untuk menentukan keakuratan model. Pengujian label dilaksanakan dengan menggunakan 4 buah kardus yang memiliki label berbeda. Setiap jenis label memiliki 10 gambar yang berbeda yang diambil menggunakan ESP32-CAM, sehingga total instrumen yang diujikan yaitu sebanyak 40 buah gambar. Gambar yang akan diproses terlebih dahulu sudah dimasukkan ke model YOLO untuk mendeteksi keberadaan kardus. Gambar 20 dan 21

Rancang Bangun Sistem Deteksi Label Kardus Berbasis Model Kecerdasan Buatan YOLO dan EasyOCR serta ESP32-CAM

menunjukkan hasil nilai *confidence* dan waktu yang dibutuhkan untuk memproses satu gambar. Pencatatan data dibantu oleh skrip *Python*.



Gambar 20. Nilai Confidence Hasil Deteksi Label EasyOCR

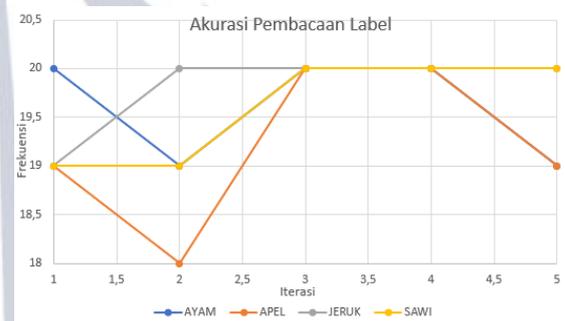


Gambar 21. Waktu Pendeteksian Label EasyOCR

Berdasarkan data yang telah ditampilkan pada gambar 20 dan 21 maka dapat diketahui bahwa model EasyOCR memiliki nilai *confidence* yang tinggi dan respon waktu yang cepat. Nilai *confidence* tertinggi dicatatkan oleh label sawi dengan nilai *confidence* sebesar 0,999869 pada masukkan ke 3. Nilai *confidence* terendah dicatatkan oleh label ayam pada masukkan ke 7 dengan nilai *confidence* sebesar 0,821043. Nilai rata-rata *confidence* terendah dicatatkan oleh label ayam dengan nilai rata rata *confidence* sebesar 0,965835. Nilai rata-rata *confidence* tertinggi dicatatkan oleh label jeruk dengan nilai rata rata *confidence* sebesar 0,973364. Waktu pemrosesan gambar terlama dicatatkan oleh label jeruk pada masukkan ke-9 yang membutuhkan waktu pemrosesan sebesar 21,11 detik. Waktu pemrosesan gambar tercepat dicatatkan oleh label ayam dengan kecepatan pemrosesan sebesar 11,47 detik pada masukkan ke-8. Rata-rata waktu pemrosesan tercepat dicatatkan oleh label ayam dengan catatan waktu rata-rata yang diperlukan sebesar 13,469 detik sedangkan waktu pemrosesan gambar terlama dicatatkan oleh label

jeruk dengan waktu rata-rata pemrosesan gambar sebesar 15,676 detik.

Pengukuran akurasi pembacaan label selanjutnya dilakukan untuk mendeterminasikan kelayakan model EasyOCR sebagai model pembaca label kardus. Gambar 22 merupakan grafik hasil pengukuran akurasi model EasyOCR yang digunakan sebagai pembaca label. Pengukuran akurasi pembacaan label ditujukan untuk mengetahui nilai error yang terjadi pada saat pembacaan label. Adapun error yang dimaksud adalah kesalahan dalam pembacaan label sehingga hasil pembacaan label tidaklah sempurna.



Gambar 22. Akurasi Pembacaan Label

Gambar 22 merupakan grafik yang mengukur akurasi pembacaan label. Pada 1 iterasi pembacaan label, terdapat 20 masukan gambar kardus berlabel. Berkat pengukuran akurasi pembacaan label, dapat diketahui nilai error hasil pembacaan label. Rata-rata error pembacaan label diketahui yaitu sebesar 2,25%. Rata-rata error terbesar diperoleh dari label apel dengan rata-rata error sebesar 4% dan rata-rata error terkecil diperoleh label jeruk dengan nilai error sebesar 1%.

Gambar 23, 24, 25, dan 26 merupakan contoh hasil pendeteksian modul EasyOCR. Dapat diketahui berdasarkan Gambar 23 bahwa hasil pembacaan label yaitu adalah ayam dengan tingkat *confidence* sebesar 0,9998302 dari nilai utuh 1. Pada Gambar 24, hasil pembacaan label yaitu apel dengan nilai *confidence* sebesar 0,9386910. Pada Gambar 25, hasil pembacaan label yaitu jeruk dengan nilai *confidence* sebesar 0,998306. Pada Gambar 26, hasil pembacaan label yaitu sawi dengan nilai *confidence* sebesar 0,9998693. Gambar 27 menunjukkan hasil deteksi label kardus beserta nilai *confidence* pembacaan label dan pendeteksian kardus pada laman website.



Gambar 23. Hasil Deteksi EasyOCR Label "AYAM"



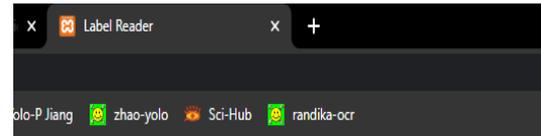
Gambar 24. Hasil Deteksi EasyOCR Label "APEL"



Gambar 25. Hasil Deteksi EasyOCR Label "JERUK"



Gambar 26. Hasil Deteksi EasyOCR Label "SAWI"



Input Terakhir=AYAM

Nilai Confidence Label=0.8210431933403015

Nilai Confidence Kardus=0.82368

Gambar 27. Laman Website via Localhost

PENUTUP

Simpulan

Berdasarkan hasil penelitian sistem deteksi label kardus berbasis model kecerdasan buatan YOLO dan EasyOCR serta ESP32-CAM maka dapat disimpulkan bahwa ESP32-CAM dapat bekerja pada sistem dengan baik, dibuktikan dari hasil respon waktu pemrosesan gambar oleh ESP32-CAM yang cepat yaitu dengan nilai rata-rata sebesar 0,762 detik. Model EasyOCR mencatatkan nilai rata-rata respon waktu pemrosesan gambar tercepat yaitu sebesar 13,469 detik.

Model kecerdasan buatan YOLO dan EasyOCR juga bekerja dengan baik di sistem. Nilai *threshold* model YOLO yang ideal untuk mendeteksi kardus yaitu 0,5 dengan rasio akurasi: error sebesar 19:1 dan hasil deteksi objek non kardus sebanyak 2/20 atau 10%. Model YOLO mencatatkan persentase rata-rata nilai *confidence* pada gambar kardus berlabel sebesar 58,002%, sedangkan pada model EasyOCR rata-rata nilai *confidence* tertinggi dicatatkan oleh label jeruk dengan nilai sebesar 0,973364 dan rata-rata nilai *confidence* terendah dicatatkan oleh label ayam dengan nilai sebesar 0,965835. Model EasyOCR mencatatkan rata-rata error pembacaan label sebesar 2,25%.

Saran

Berdasarkan penelitian yang telah dilakukan, ada beberapa saran yang dapat diimplementasikan dalam pengembangan sistem deteksi label kardus berbasis model YOLO dan EasyOCR. Saran tersebut yaitu dengan meningkatkan jumlah kumpulan data dalam fase pelatihan model YOLO. Kumpulan data yang digunakan diharapkan mengandung gambar kardus berlabel sehingga meningkatkan nilai *confidence* pendeteksian kardus. Selain itu, memperpanjang sabuk konveyor dapat dilakukan untuk memperbaiki kekurangan sabuk konveyor.

Rancang Bangun Sistem Deteksi Label Kardus Berbasis Model Kecerdasan Buatan YOLO dan EasyOCR serta ESP32-CAM

DAFTAR PUSTAKA

- Bahrin. 2017. *Sistem kontrol penerangan menggunakan arduino uno pada Universitas Ichsan Gorontalo*. ILKOM Jurnal Ilmiah Vol. 9 No. 3 hal. 282-289.
- Chang, Jie. 2016. *Research on and implementation of the logistics warehouse management system*. Makalah disajikan dalam 2nd International Conference on Social Science and Technology Education (ICSSTE 2016), Cina, 14-15 Mei.
- Chen, Zhimi, Siu Liazhen, Song Wei, dan Du Chao. 2017. *Research on warehouse management system based on association rules*. Makalah disajikan dalam 6th International Conference on Computer Science and Network Technology (ICCSNT 2017), Cina, 21-22 Oktober.
- Dokic, Kristian. 2020. *Microcontrollers on the Edge – is ESP32 with camera ready for machine learning?*. Lecture Notes of Computer Science (LNCS) 12119, pp. 213–220.
- Ghifari, Hummam Gassan, Denny Darlis, dan Aris Hartaman. 2020. *Pendeteksi golongan darah manusia berbasis tensorflow menggunakan ESP32-CAM*. ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika. Vol 9, pp. 359-357.
- Hamdy, Walaa, Noha Mostafa, dan Hesham Alawady. 2020. *An intelligent warehouse management system using the internet of things*. The Egyptian International Journal of Engineering Sciences and Technology. Vol. 32 (2020) 59–65.
- Jiang, Peiyuan, Ergu Daji, Liu Fangyao, Cai Ying, dan Ma Bo. 2021. *A review of yolo algorithm developments*. Makalah disajikan dalam 8th International Conference on Information Technology and Quantitative Management (ITQM 2020 & 2021), Cina, 9-11 Juli.
- Padilla, Rafael, Netto Sergio L., dan Da Silva Eduardo A.B.. 2020. *A survey on performance metrics for object-detection algorithms*. Makalah disajikan dalam 2020 International Conference on System, Signals and Image Processing (IWSSIP), Brazil, 1-3 Juli.
- Redmon, Joseph, Divvala Santosh, Girschick Ross, dan Farhadi Ali. 2016. *You Only Look Once: unified, real-time object detection*, (Online), (<https://doi.org/10.48550/arXiv.1506.02640> diunduh 10 April 2022).
- Smelyakov, Kirill, Chupryna Anastasya, Dmytro Darahan, dan Midina Serhii. 2021. *Effectiveness of modern text recognition solutions and tools for common data sources*. Makalah disajikan dalam 5th International Conference on Computational Linguistics and Intelligent Systems (ICOLINS-2021), Ukraina, 22-23 April.