



IMPELMENTASI SISTEM PENGLIHATAN KAMERA SEDERHANA DALAM FORKLIFT ELEKTIK

*Mark Davisson Djunaedi ¹, Joni Fat ², Hugeng Hugeng ³

^{1,2,3}Elektro, Teknik, Universitas Tarumanagara, Jakarta, DKI Jakarta, Indonesia, 11470

*Email: mark.525200002@stu.untar.ac.id

ABSTRAK

Otomatisasi dalam industri memainkan peran krusial dalam meningkatkan efisiensi dan kemudahan pekerjaan. Salah satu contoh implementasinya adalah melalui integrasi sistem visi komputer pada *forklift* elektrik, yang memungkinkan pengangkutan barang berat secara otomatis. Dengan menggunakan sistem visi komputer sederhana untuk mengenali dan mengikuti jalur yang telah ditetapkan *forklift* bisa berkerja secara otomatis. *Forklift* elektrik juga dipasang dengan berbagai sensor untuk mencegah ada tabrakan selama *forklift* bergerak. Jalur yang digunakan adalah sebuah garis putih yang dikelilingi oleh dua garis hitam di sisi kanan dan kiri. *Forklift* elektrik dikendalikan melalui mikrokontroler Arduino Mega, yang berfungsi sebagai pusat kendali motor *forklift*. Sistem visi komputer dapat diintegrasikan dengan menggunakan PC mini seperti *Jetson Nano*, yang bertindak sebagai pemroses citra utama. Setelah proses visi komputer selesai, *Jetson Nano* dapat mengirimkan perintah kepada Arduino melalui koneksi serial kepada Arduino untuk menggerakkan *forklift* elektrik. Visi komputer menggunakan bahasa program *python* dan menggunakan beberapa library seperti *pyserial*, *opencv*, *numpy* dan lainnya. Selama tahap integrasi antara sistem visi komputer dan *forklift* elektrik, dilakukan serangkaian percobaan dan perbaikan berdasarkan hasil percobaan sebelumnya. Perubahan-perubahan yang diimplementasikan seperti penyesuaian parameter pergerakan ke arah kanan atau kiri, kecepatan *forklift*, dan lain sebagainya.

Kata Kunci : Arduino Mega, *electric forklift*, *Jetson Nano*, visi komputer.

ABSTRACT

Automation in the industry plays a crucial role in enhancing efficiency and facilitating work processes. One example of its implementation is through the integration of computer vision systems in electric forklifts, enabling the automated transportation of heavy goods. By utilizing a simple computer vision system to recognize and follow predefined paths, the forklift can operate autonomously. The electric forklift is also equipped with various sensors to prevent collisions during its movement. The path consists of a white line surrounded by two black lines on the right and left sides. The electric forklift is controlled by an Arduino Mega microcontroller, serving as the central motor control unit. The computer vision system can be integrated using a mini PC like Jetson Nano, acting as the primary image processor. Once the computer vision process is completed, Jetson Nano sends commands to the Arduino via a serial connection to drive the electric forklift. During the integration of the Jetson Nano computer vision system and the electric forklift, a series of experiments and adjustments are conducted based on previous trial results. Implemented changes include parameter adjustments for right or left movements, forklift speed, and other relevant factors.

Keywords: Arduino Mega, computer vision, *electik forklift*; *Jetson Nano*;

Submitted : Revision : Accepted :

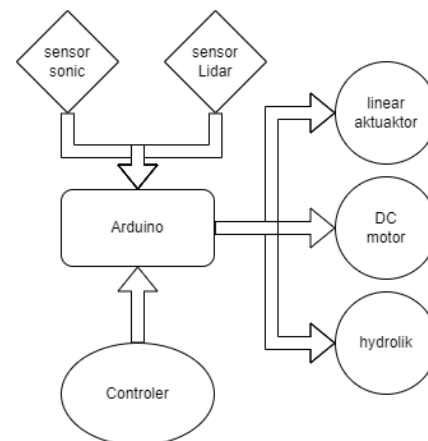
PENDAHULUAN

Otomatisasi dalam bidang industri telah menjadi kunci utama dalam menghadapi tantangan dan meningkatkan efisiensi dalam berbagai sektor. Dengan kemajuan teknologi dan perkembangan sistem cerdas, otomatisasi telah mengubah cara kerja dan produksi di industri modern. Kelancaran proses pekerjaan sangat diutamakan dalam suatu industri agar dapat diperoleh hasil yang maksimal sehingga tidak menimbulkan kerugian (Tjandra, Hugeng, & Nurwijayanti, 2019). Dengan menggabungkan sistem komputer, sensor, dan kontrol, otomatisasi menjadi sangat mudah untuk diimplementasikan dalam hampir seluruh proses produksi atau pun proses pemeliharaan. Dalam artikel ini akan membahas hasil dari perencanaan dan pengembangan dari *forklift* elektrik yang diintegrasikan dengan *computer vision* (penglihatan komputer) sederhana.

Computer vision adalah kecerdasan buatan (AI) yang memungkinkan komputer dan sistem memperoleh informasi bermakna dari gambar digital, video, dan input visual lainnya — dan mengambil tindakan atau membuat rekomendasi berdasarkan informasi tersebut (Howse & Minichino, 2020; Voulodimos, Doulamis, Doulamis, & Protopapadakis, 2018). Dalam konteks integrasi *forklift* elektrik, *computer vision* memainkan peran penting dalam meningkatkan efisiensi dan keamanan operasi *forklift*. Kemampuan *forklift* untuk “melihat” dapat mengoptimalkan pekerjaan *forklift* dalam mengangkat barang, navigasi *forklift* dan menghindari tabrakan. Pemasangan fitur *computer vision* meskipun dimulai dengan fungsi sederhana seperti mengikuti suatu garis hitam putih.

Artikel ini bertujuan untuk membahas hasil dari integrasi antara sistem *forklift* elektrik dengan penggunaan *computer vision* sederhana. Dalam metode *computer vision* sederhana ini, digunakan sebuah kamera untuk mendeteksi jalur yang telah ditentukan dalam bentuk garis. Meskipun metode ini masih sangat sederhana, namun mampu memberikan kontribusi yang signifikan dalam mempermudah pekerjaan yang berat. Penambahan *computer vision* juga diharapkan meningkatkan keakuratan, presisi dan efisiensi dari pergerakan *forklift*. Dengan adanya penelitian dan pengembangan yang terus dilakukan, diharapkan teknologi *computer vision* dalam robotika dan otomatisasi industri dapat terus berkembang dan memberikan kontribusi yang lebih besar dalam meningkatkan efisiensi dan produktivitas di berbagai sektor industri.

Dalam perencanaan, dilakukan pengkajian yang membantu dalam membimbing pengintegrasian *forklift* dan komputer vision. Salah satu panduan yang digunakan untuk mengintegrasikan *forklift* elektrik dan komputer vision adalah "A Computer vision System for Autonomous Forklift Vehicles in Industrial Environments" (Syu et al., 2017). Dalam artikel ini, komputer vision digunakan dengan memproses warna untuk mengidentifikasi jalur yang telah dibuat sebelumnya. Setelah kamera mengambil satu *frame*, komputer akan mengubah warna dari BGR (Biru Hijau Merah) menjadi HSV (Hue Saturation Value). Hal ini dilakukan untuk memproses dan menganalisis gambar dengan lebih baik dalam hal segmentasi warna, deteksi objek, dan pemrosesan visual yang lebih tahan terhadap perubahan pencahayaan (Szeliski, 2010). Setelah dilakukan perubahan tersebut, komputer akan mencari fitur-fitur khusus yang dapat dijadikan titik-titik untuk mendeteksi bentuk yang dicari. Titik-titik ini kemudian dianalisis oleh komputer dan ditentukan apakah bentuk tersebut sesuai dengan yang diinginkan oleh komputer.

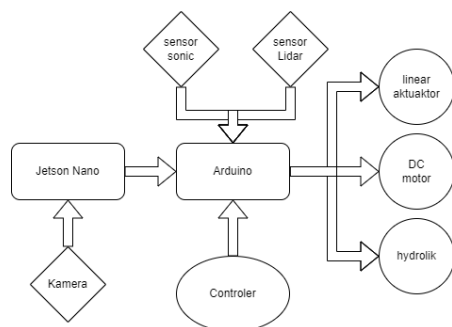


Gambar 1. Diagram *forklift* elektrik

Perencanaan untuk mengintegrasikan *forklift* elektrik dengan komputer vision dimulai dengan pemahaman terhadap kemampuan *forklift* itu sendiri. *Forklift* elektrik dikendalikan oleh Arduino Mega sebagai pengendali yang akan membaca semua sensor. Untuk pergerakan *forklift*, terdapat tiga komponen penggerak yaitu motor DC untuk maju dan mundur, aktuator linear untuk mengontrol kemampuan belok roda ke kanan dan kiri, serta sistem hidrolik untuk mengangkat garpu dan badan *forklift*. Pada *forklift*, juga dipasang tiga sensor utama, yaitu sensor sonic untuk menghindari tabrakan, dan sensor lidar untuk mendeteksi ketinggian garpu *forklift*. Dalam pengoperasiannya,

forklift dapat dikendalikan secara manual melalui sebuah pengontrol jika terjadi kerusakan atau saat melakukan pengisian baterai.

Dalam integrasi komputer vision sederhana yang dijelaskan dalam artikel ini, peran mini PC seperti *Jetson Nano* sangat penting sebagai pengolah data dari kamera. Mini PC ini bertanggung jawab untuk menerima input visual dari kamera yang terpasang pada *forklift*. Setelah menerima data gambar, *Jetson Nano* akan memprosesnya menggunakan algoritma dan teknik komputer vision. Setelah diproses *Jetson Nano* akan mengirimkan perintah kepada Arduino untuk menggerakkan DC motor atau linear aktuaktor sehingga *forklift* akan berjalan sesuai dengan jalur yang sudah ditentukan.



Gambar 2. *Forklift* elektrik yang diintegrasikan dengan *computer vision*

METODOLOGI

Metodologi integrasi *forklift* elektrik dengan teknologi komputer vision dapat dibagi menjadi beberapa tahap. Berikut adalah metodologi yang tepat untuk mengintegrasikan *forklift* elektrik dengan teknologi komputer vision:

Tahap 1: Persiapan Hardware

Pemasangan Perangkat Hardware, Pasang Jetson Nano Developer Kit pada *forklift*, Sumbungkan USB Webcam Logitech V-Ubm46 ke Jetson Nano, Pasang Arduino Mega, sensor sonic, dan sensor lidar pada *forklift*, Hubungkan semua perangkat menggunakan kabel yang sesuai, Pengaturan Daya, Gunakan baterai 12 volt dan UBEC untuk menyediakan daya pada Jetson Nano, Pastikan baterai dan daya listrik lainnya memadai untuk menggerakkan *forklift*.

Tahap 2: Pengaturan Perangkat Lunak

Instalasi Sistem Operasi:

Unduh sistem operasi Ubuntu dari situs resmi NVIDIA dan instal pada kartu SD 64 GB untuk Jetson Nano, Konfigurasi Jetson Nano, Atur Jetson Nano menggunakan JetPack SDK. Instal library Python seperti OpenCV, NumPy, dan pyzbar untuk komputer visi, Pemrograman Arduino Mega Instal library pyserial pada Arduino Mega, Bangun program sederhana dengan dua mode (manual dan scan otomatis), Tambahkan logika untuk menerima perintah dari Jetson Nano dan joystick.

Tahap 3: Pengujian Computer Vision

Pengujian Proses Computer Vision:

Gunakan program pada Jetson Nano untuk membaca dan mengolah gambar dari webcam, Uji kemampuan sistem dalam mengenali jalur putih pada lantai, Lakukan percobaan untuk memastikan perubahan warna, identifikasi bentuk, dan pembacaan garis berhasil.

Tahap 4: Pengujian Komunikasi

Pengujian Komunikasi Jetson Nano dan Arduino, Uji komunikasi antara Jetson Nano dan Arduino Mega, Pastikan perintah dari Jetson Nano dapat dibaca dengan baik oleh Arduino Mega.

Tahap 5: Uji Coba Forklift pada Jalur

Uji Coba Forklift pada Jalur yang Ditentukan, Program *forklift* untuk bergerak lurus dan berbelok sesuai jalur yang telah dibuat, Lakukan uji coba untuk melihat sejauh mana *forklift* dapat mengikuti jalur dengan akurasi dan presisi, Perhatikan respons *forklift* terhadap perintah dari Jetson Nano dan sensor sonic.

Tahap 6: Penyesuaian dan Optimalisasi

Penyesuaian Parameter:

Lakukan penyesuaian parameter pada program komputer vision untuk meningkatkan keakuratan dalam mengenali jalur dan bentuk, Sesuaikan parameter kendali *forklift* untuk mengoptimalkan pergerakan dan responsnya.

Tahap 7: Dokumentasi dan Evaluasi

Dokumentasi Hasil dan Pengalaman: Catat hasil dari setiap tahap dan perubahan yang dilakukan.

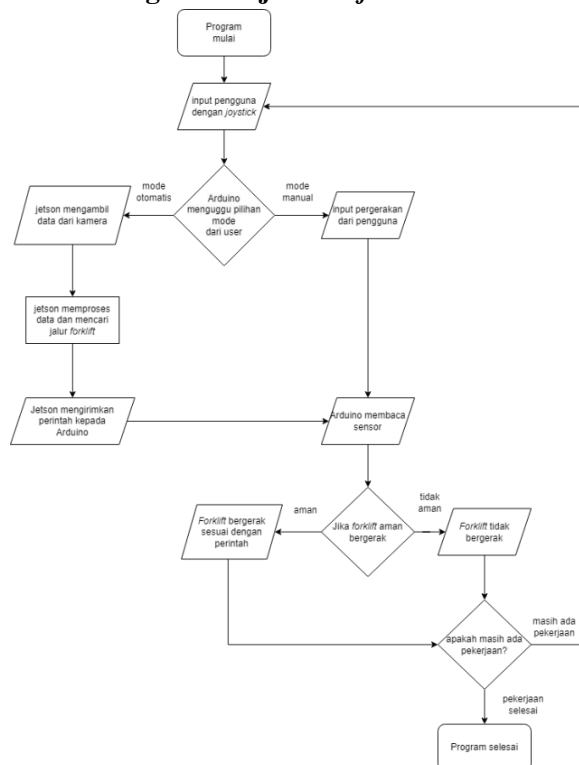
Evaluasi Keseluruhan: Evaluasi keseluruhan performa *forklift* berdasarkan tujuan awal integrasi, Identifikasi potensi pengembangan dan peningkatan lebih lanjut.

Metodologi ini mencakup langkah-langkah dari persiapan hardware hingga uji coba praktis, memastikan integrasi yang efektif antara forklift elektrik dan teknologi computer vision. Selain itu, tahap penyesuaian dan optimalisasi menjadi kunci untuk mencapai performa yang optimal pada aplikasi ini.

Deskripsi Konsep

Dengan menambahkan teknologi *computer vision*, *forklift* elektrik dapat bergerak dengan akurasi dan presisi yang lebih baik. Jalur yang digunakan terdiri dari garis putih dengan garis putih tambahan di kedua sisinya. *Jetson Nano* akan menggunakan garis putih ini sebagai patokan utama yang diikuti oleh *forklift*. Untuk meningkatkan keamanan, sensor sonic dipasang pada *forklift* untuk mendeteksi potensi tabrakan dengan objek di sekitarnya. Selain itu, dalam perancangan ini digunakan *Jetson Nano*, USB Webcam Logitech V-Ubm46, Arduino Mega, baterai, sensor sonic, dan sensor lidar. Dengan kombinasi ini, diharapkan *forklift* elektrik dapat bergerak dengan akurasi tinggi, mengikuti jalur yang ditentukan, dan menghindari tabrakan dengan objek sekitarnya.

Diagram Kerja Forklift



Gambar 3. Diagram kerja *forklift* elektrik

Forklift dilengkapi dengan program sederhana menggunakan Arduino untuk mempermudah pengembangan selanjutnya.

Program ini memiliki dua mode, yaitu mode manual dan mode scan otomatis. Pada mode manual, Arduino menerima perintah langsung dari *joystick* untuk menggerakkan *forklift*. Sementara pada mode scan otomatis, Arduino menerima perintah dari *Jetson Nano* yang menggunakan kamera dan library OpenCV untuk mencari bentuk persegi panjang berwarna hitam dalam gambar. Setelah menemukan bentuk dan warna yang diinginkan, *Jetson Nano* memberikan perintah kepada Arduino untuk mengikuti jalur yang telah ditentukan. Program ini memberikan dasar yang kuat untuk pengembangan lebih lanjut pada *forklift*.

Selain itu, kedua mode di atas juga dilengkapi dengan program untuk membaca sensor sonic guna mengantisipasi tabrakan. Jika perintah yang diterima akan menyebabkan tabrakan, Arduino akan menghentikan eksekusi perintah tersebut, memberikan perlindungan keamanan bagi *forklift* dan pengguna. Sensor lidar juga dibaca untuk memastikan penanganan barang oleh *forklift* dapat dilakukan dengan baik dan presisi. Program ini secara keseluruhan memberikan fitur-fitur yang penting dan dapat digunakan sebagai dasar untuk pengembangan lebih lanjut pada *forklift*.

Tahapan Integrasi Forklift

Proses integrasi *forklift* elektrik dengan *computer vision* membutuhkan serangkaian tahapan yang penting dan harus dijalankan dengan cermat. Tahapan-tahapan ini mencakup langkah awal dalam melakukan pengaturan *Jetson Nano* yang akan digunakan sebagai sistem pengolahan gambar, hingga uji coba *forklift* pada jalur yang telah dirancang sebelumnya. Dalam artikel ini, setiap tahapan akan dijelaskan secara mendetail untuk memberikan pemahaman yang baik tentang proses integrasi ini.

Pengaturan Awal Jetson Nano

Dalam integrasi *forklift* otomatis dengan *computer vision*, diperlukan kekuatan pemrosesan yang cukup tinggi namun tetap efisien dalam penggunaan daya listrik dari baterai. Untuk alasan ini, dipilihlah mini PC *Jetson Nano Developer Kit* yang menawarkan spesifikasi yang sesuai. *Jetson Nano Developer Kit* dilengkapi dengan prosesor NVIDIA Quad-core ARM Cortex-A57 MPCore, GPU NVIDIA Maxwell dengan 128 inti CUDA, dan memori LPDDR4 sebesar 4GB. *Jetson Nano* juga didukung oleh JetPack SDK yang menyediakan beragam perangkat lunak dan library untuk pengembangan aplikasi kecerdasan buatan (AI) dan komputer visi.

Jetson Nano memiliki kemampuan untuk

menjalankan *framework* deep learning populer seperti TensorFlow, PyTorch, dan Caffe. Hal ini memungkinkan pengguna untuk mengembangkan dan mengimplementasikan model AI dengan mudah. *Jetson Nano* juga memiliki konsumsi daya yang rendah, hanya membutuhkan tegangan 5 VDC untuk operasinya, yang sejalan dengan Arduino. Selain itu, *Jetson Nano* juga memiliki dua mode daya, yaitu mode 10 watt dan mode 5 watt, yang memberikan fleksibilitas dalam penggunaan baterai sesuai dengan kebutuhan. Dengan spesifikasi di atas dan fitur-fitur yang efisien, *Jetson Nano Developer Kit* menjadi pilihan yang ideal untuk integrasi *forklift* otomatis dengan *computer vision*.



Gambar 4. *Jetson Nano Developer Kit*

Dalam hal penyimpanan data, dipilihlah kartu SD dengan kapasitas 64 GB yang sudah diformat dan di-flash dengan sistem operasi Ubuntu yang diunduh dari situs web resmi NVIDIA [5]. Untuk memberikan daya kepada *Jetson Nano*, digunakan baterai array 12 volt yang akan diatur tegangannya menggunakan *Universal Battery Elimination Circuit* (UBEC). Selain itu, untuk mencegah terjadinya panas berlebih, *Jetson Nano* dilengkapi dengan kipas kecil yang dipasang di atas heat sink-nya.

Bahasa pemrograman yang digunakan oleh *Jetson Nano* untuk *computer vision* adalah *Python*. Dalam *Python*, Anda perlu menginstal beberapa library untuk komputer visi, seperti *opencv-contrib-python*, *numpy*, dan *pyzbar*. Untuk *Jetson Nano* dapat berkomunikasi dengan Arduino Mega perlu menggunakan *pyserial*. Dengan menggunakan *pyserial* Arduino dan *Jetson Nano* dapat berkeja sama mengerjakan *forklift*.

Pengaturan Awal Arduino Mega



Gambar 5. Arduino Mega

Arduino Mega adalah sebuah papan pengembangan mikrokontroler yang didasarkan pada mikrokontroler ATmega2560. Arduino Mega menyediakan 54 pin digital dan 16 pin analog sebagai antarmuka untuk berbagai perangkat dan sensor eksternal. Selain itu, Arduino Mega memiliki total 4 port serial (UART) yang dapat digunakan untuk berkomunikasi secara serial dengan perangkat eksternal. Setiap port serial memiliki pin TX (transmit) dan RX (receive), memungkinkan pengiriman dan penerimaan data secara serial.



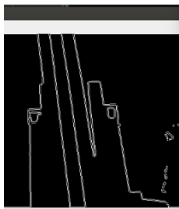

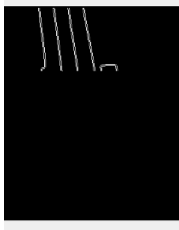

Dalam pembahasan ini digunakan salah satu port serial sebagai saluran komunikasi antara Arduino Mega dan *Jetson Nano*. Dari 4 port serial yang ada port serial 0 dipilih sebagai jalur komunikasi. Pemilihan port serial 0 didasarkan pada kemudahan *Jetson Nano* dalam berkomunikasi melalui kabel USB Type B. Untuk membaca data serial yang diterima dari *Jetson Nano*, digunakan fungsi Arduino bernama *Serial.readStringUntil()*. Fungsi ini memungkinkan Arduino Mega untuk membaca data serial sampai karakter tertentu ditemukan.

HASIL DAN PEMBAHASAN

Percobaan pertama adalah menguji kemampuan *computer vision* pada *Jetson Nano*. Program ini dirancang untuk membaca jalur yang telah dibuat dan memberikan perintah yang sesuai. Jalur tersebut terdiri dari satu garis putih yang dikelilingi oleh dua garis hitam di sebelah kanan dan kiri garis putih. Dalam percobaan ini, *Jetson Nano* akan menghasilkan *frame* gambar yang telah diambil dan sedang diproses, sehingga dapat dilihat apa yang telah dilakukan oleh sistem dan apa yang dapat diperbaiki.

Tabel 1. Pengujian proses *computer vision*

Parameter keberhasilan proses <i>computer vision</i>	Frame yang ditunjukkan oleh <i>Jetson Nano</i>	Keterangan
Jetson		Berhasil

mengambil <i>frame</i> dari kamera		
Jetson mengubah warna dari BGR ke dalam HSV		Berhasil
Mengidentifikasi kasi bentuk dari perbedaan yang ada dan menandainya dengan garis putih		Berhasil
Jetson membuat <i>frame</i> menandakan tempat yang akan dianalisis		Berhasil
Jetson hanya mengambil garis yang berada dalam <i>frame</i> yang diinginkan		Berhasil
Jetson membaca garis dari sebelumnya dan merepresentasikannya dalam <i>frame</i> awal.		Berhasil

Proses *computer vision* yang digunakan dalam *Jetson Nano* dimulai dengan pengambilan *frame* menggunakan fungsi `cv2.VideoCapture` dari

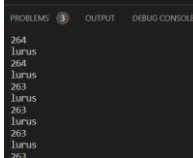
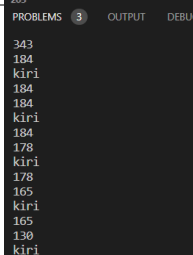

library OpenCV (Howse & Minichino, 2020). *Frame* yang telah diambil akan mengalami transformasi warna dari BGR (Blue Green Red) ke HSV (Hue Saturation Value). Tujuan dari perubahan warna ini adalah agar komputer dapat dengan lebih mudah mendeteksi warna tertentu dan mengurangi ketergantungan pada kondisi pencahayaan yang spesifik.

Setelah transformasi warna, Jetson akan mencari garis-garis putih pada *frame* yang telah diubah. Selanjutnya, kita akan membuat sebuah "mask" atau masker yang akan membatasi area di dalam *frame* yang hanya berisi jalur di depan *forklift*, dan menghilangkan objek seperti garpu *forklift* atau kaki *forklift*. Mask ini akan diberi warna putih sesuai dengan garis putih yang telah dibuat sebelumnya, untuk membedakan area yang akan diproses. Selanjutnya, Jetson akan membandingkan hasil perbedaan warna dengan masker yang telah dibuat, dan hanya akan mengambil garis-garis putih yang terletak dalam area masker tersebut. Jetson akan membaca dan menyimpan koordinat dari garis-garis putih yang terdeteksi.

Dalam pengujian *computer vision* yang tercantum dalam tabel di atas, sistem berhasil mengenali jalur yang telah dibuat. Dalam pengujian tersebut, terdapat lebih dari satu garis yang terdeteksi karena sistem *computer vision* juga mengenali beberapa garis lainnya yang ada pada lantai percobaan yang berwarna putih. Untuk mengatasi variasi ini, parameter-parameter tersebut dihitung berdasarkan rata-rata dari 25 data, sehingga *Jetson Nano* dapat lebih baik dalam membaca keberadaan garis putih.

Setelah pengujian *computer vision* selesai, dilakukan pengujian komunikasi antara *Jetson Nano* dan Arduino. Pengujian ini akan dilakukan sebanyak 5 kali untuk setiap perintah yang ada. Tujuan dari pengujian ini adalah untuk memastikan bahwa komunikasi antara *Jetson Nano* dan Arduino berjalan dengan baik dan stabil, sehingga pada pengujian selanjutnya tidak perlu dilakukan pemeriksaan komunikasi secara berulang.

Tabel 2. Pengujian komunikasi *Jetson Nano* dan Arduino Mega

Perintah dari <i>Jetson Nano</i>	Pembacaan Arduino	Keterangan
lurus		Berhasil
kiri		Berhasil
kanan		Berhasil

Dengan melakukan pengujian komunikasi yang memadai, dapat dipastikan bahwa sistem komunikasi antara *Jetson Nano* dan Arduino telah teruji dengan baik dan dapat diandalkan. Hal ini akan memberikan kepercayaan dan meminimalkan risiko terjadinya masalah komunikasi pada tahap-tahap selanjutnya.

Pengujian selanjutnya melibatkan pergerakan *forklift* yang mengikuti jalur yang telah disiapkan. Hasil pengujian menunjukkan bahwa *forklift* mampu mengikuti jalur dengan baik dan melakukan perubahan arah dengan lancar saat mencapai tikungan. Hasil pengujian ini membuktikan kemampuan *forklift* dalam menggunakan *computer vision* dan integrasi dengan perangkat lainnya, memberikan dasar yang kuat untuk pengembangan selanjutnya.

Tabel 3. Pengujian *Forklift* berjalan dalam jalur

Parameter keberhasilan	Hasil Pengujian	Keterangan
------------------------	-----------------	------------

Forklift berjalan lurus



Berhasil

Forklift bisa berjalan melewati jalur berliku-liku



Berhasil

Dalam pengujian yang dilakukan, terdapat beberapa hal yang perlu dibahas sebelum mencapai keberhasilan yang telah disebutkan sebelumnya. Salah satu aspek yang penting adalah pengaturan parameter, yang memiliki dampak signifikan terhadap keberhasilan pengujian, terutama dalam menentukan kapan *forklift* harus memberikan perintah untuk berbelok ke kanan atau kiri. Untuk menemukan parameter yang optimal dalam hal ini, dilakukan beberapa percobaan dan penyesuaian.

Selain itu, penting juga untuk memperhatikan sudut perputaran *forklift* elektrik. *Forklift* membutuhkan waktu tertentu untuk dapat berbelok secara penuh, dengan waktu sekitar 3 detik. Dalam pengujian, *forklift* diprogram untuk memiliki 5 kondisi roda yang berbeda, yaitu roda berputar penuh ke kiri, setengah ke kiri, lurus, setengah ke kanan, dan penuh ke kanan. Hal ini dilakukan agar *forklift* dapat meluruskan posisinya pada jalur lurus tanpa perlu melakukan perputaran penuh, jika tidak *forklift* bisa menjadi terlalu miring ke arah yang berlawanan.

KESIMPULAN

Dari hasil pengimplementasian ini, beberapa kesimpulan dapat ditarik, yaitu:

1. Penggunaan metode *computer vision* sederhana, seperti mengidentifikasi garis, telah terbukti efektif dalam menciptakan program line follower yang sukses. Meskipun diperlukan beberapa uji coba untuk mengendalikan parameter dengan tepat, kesuksesan dalam mengikuti jalur menunjukkan potensi besar dari penggunaan teknologi ini dalam industri.

2. Komunikasi antara *Jetson Nano* dan *Arduino* berjalan dengan lancar menggunakan *pyserial*. Pengiriman perintah antara *Jetson* dan *Arduino* dapat dilakukan dengan mudah dan andal, memungkinkan koordinasi yang efisien antara sistem visi komputer dan kendali motor *forklift*.

3. *Forklift* elektrik berhasil berjalan sesuai dengan jalur yang ditentukan. Dengan bantuan sistem visi komputer dan sensor-sensor yang dipasang, Hal ini membuktikan kemampuan *forklift* dalam melakukan tugas pengangkutan barang secara otomatis dan akurat.

4. Pengimplementasian ini telah membawa pemahaman yang lebih baik tentang integrasi antara computer vision dan *forklift* elektrik. Melalui serangkaian percobaan dan perbaikan, parameter-parameter seperti penyesuaian pergerakan dan kecepatan *forklift* dapat diatur dengan lebih baik, meningkatkan kinerja dan efisiensi sistem secara keseluruhan.

Dengan kesimpulan ini, dapat dilihat bahwa integrasi computer vision pada *forklift* elektrik memberikan potensi besar dalam meningkatkan otomatisasi dan efisiensi dalam industri. Penelitian dan pengembangan lebih lanjut di bidang ini dapat membawa kemajuan yang signifikan dalam penggunaan teknologi ini dalam konteks industri modern.

UCAPAN TERIMA KASIH

Ucapan terimakasih diberikan kepada pembimbing dan keran kerja dalam Pt. IUS yang sudah memberikan kesempatan untuk didanai pembuatan alat dalam artikel ini.

DAFTAR PUSTAKA

- Howse, J., & Minichino, J. (2020). *Learning OpenCV 4 Computer Vision with Python 3: Get to grips with tools, techniques, and algorithms for computer vision and machine learning*. Packt Publishing Ltd.
- Syu, J. L., Li, H. T., Chiang, J. S., Hsia, C. H., Wu, P. H., Hsieh, C. F., & Li, S. A. (2017). A computer vision assisted system for autonomous *forklift* vehicles in real factory environment. *Multimedia Tools and Applications*, 76(18), 18387–18407. <https://doi.org/10.1007/s11042-016-4123-6>
- Szeliski, R. (2010). *Computer Vision: Algorithms and Applications*.

- Tjandra, N., Hugeng, H., & Nurwijayanti, N. (2019). Alat Pemantau Jumlah Hasil Produksi dalam Industri Sepatu. *TESLA: Jurnal Teknik Elektro*, 20(1), 16–26. <https://doi.org/10.24912/tesla.v20i1.2817>
- Voulodimos, A., Doulamis, N., Doulamis, A., & Protopapadakis, E. (2018). Deep learning for computer vision: A brief review. *Computational Intelligence and Neuroscience*, 2018.