A Study of a Laser Engraving System Based on a Cartesian Robot with Image Processing

Thai-Duong Hoang [1], Manh-Dung Nguyen [2,*], Chi-Phat Pham [3], Thi-Ngoc-Thao Nguyen [4], Tan-Phat Nguyen [5], Anh-Son Tran [6], Quang-Thuan Le [7], Phuoc-Thinh Dang [8], Thai-Hiep Nguyen [9], Quang-Tung Trinh [10], Hoai-Bao-Nhan Nguyen [11], Huu-Nhan Nguyen [12], Thanh-Binh Nguyen [13]
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13] Ho Chi Minh City University of Technology and Engineering (HCM-UTE), Ho Chi Minh City (HCMC), Vietnam
Email: [1]211511090@student.hcmute.edu.vn, [2] 211511089@student.hcmute.edu.vn, [3] 20119191@student.hcmute.edu.vn, [4] thaontn@hcmute.edu.vn, [5] 22146190@student.hcmute.edu.vn, [6] 22144180@student.hcmute.edu.vn, [7] 22144204@student.hcmute.edu.vn, [8] 22146233@student.hcmute.edu.vn, [9] 22146121@student.hcmute.edu.vn, [10] 22146251@student.hcmute.edu.vn, [11] 22146185@student.hcmute.edu.vn, [12] 22146186@student.hcmute.edu.vn, [13] binhnt@hcmute.edu.vn
*Corresponding Author

***Abstract*—Traditional CNC laser engraving systems often face limitations in flexibility, requiring manual alignment and pre-defined G-code files. This paper proposes an advanced automated laser engraving system based on a 3-axis Cartesian robot that bridges the gap between industrial control reliability and modern computer vision. The core novelty of this research lies in the seamless integration of a Mitsubishi Q03UDE Programmable Logic Controller (PLC) with a Python-based image processing framework. By utilizing the OpenCV library for real-time edge detection and trajectory generation, the system can autonomously identify object positions and convert complex patterns into precise motion commands. Communication is established via the MC Protocol over Ethernet, ensuring high-speed data synchronization between the vision system and the servo-driven hardware. Experimental results demonstrate that the proposed system achieves high precision in engraving, significantly reduces setup time by eliminating manual calibration, and maintains the robust stability required for industrial environments. This approach provides a scalable solution for intelligent manufacturing and personalized production.**

***Keywords*—Laser Engraving; Cartesian Robot; Mitsubishi Q03UDE; PLC; Python; Image Processing; OpenCV; MC Protocol; Industrial Automation**

I. INTRODUCTION

In the modern industrial landscape, the integration of high-precision robotics and intelligent sensing has become a pivotal factor in achieving the goals of Industry 4.0 [1]. Cartesian robots, characterized by their linear axes and robust structural frames, remain a preferred choice for material processing due to their high positioning accuracy and straightforward kinematic modeling [2], [3]. Among various applications, laser engraving has evolved from simple marking to complex surface treatments, necessitating control systems that can handle both high-speed motion and intricate path planning [4].

However, conventional laser systems often rely on static G-code generation, which lacks the flexibility to adapt to dynamic manufacturing environments [5]. To address this, the incorporation of Computer Vision (CV) has been proposed to automate object recognition and trajectory mapping [6]. Recent studies have shown that using the OpenCV library allows for efficient real-time edge detection and contour extraction, which are essential for translating visual data into robotic motion [7], [8].

While many researchers utilize microcontrollers for prototyping, the harsh conditions of industrial environments demand the reliability and electromagnetic interference (EMI) resistance of Programmable Logic Controllers (PLCs) [9]. The Mitsubishi Q-series, in particular, has been highlighted for its high-speed processing capabilities and support for advanced motion control modules [10]. Furthermore, the communication between high-level vision algorithms (Python) and low-level hardware (PLC) is a critical bottleneck. The use of Ethernet-based protocols, such as the MC Protocol, has been proven effective for low-latency data synchronization in integrated systems [11], [12].

Advanced motion control techniques, including PID tuning and closed-loop servo systems, are vital to maintaining engraving quality across different material surfaces [13]. Recent advancements in hybrid control architectures have demonstrated that combining the computational power of PCs with the stability of PLCs can significantly reduce system response time and improve engraving throughput [14]. By eliminating the need for manual alignment and pre-defined coding, such intelligent systems provide a scalable solution for personalized production and high-volume industrial tasks [15]. The primary objectives of this study are twofold: first, to implement a real-time image processing pipeline that ensures immediate classification of objects on a moving conveyor; and second, to optimize system efficiency and accuracy. By replacing manual sorting with an automated robotic solution, the system aims to mitigate human-induced errors while significantly reducing operational costs and cycle times in industrial applications [16]-[22].

II. MODEL OF CARTESIAN ROBOTS

A. Properties of Cartesian Robots

1. Orthogonal Structure:

The most defining property is that all axes (X, Y, Z) are perpendicular (orthogonal) to one another. This rigid 90-degree alignment simplifies mechanical design and ensures predictable motion.

2. High Structural Stiffness:

Because the axes are typically supported at both ends (especially in "Gantry" configurations) rather than cantilevered like a human arm, they suffer from very little deflection. This makes them incredibly rigid.

3. Scalability:

They are modular. If you need a longer reach, you can simply use a longer rail for that specific axis without redesigning the entire robot. This is impossible with articulated arms, where extending reach requires larger motors and completely new linkage designs.

B. Mathematical model of a Cartesian Robot

1. Forward Kinematics:

For Cartesian robots, establishing a Denavit-Hartenberg (DH) table is not strictly mandatory, as the axes align directly with the Cartesian coordinate system and involve solely translational motion. However, to maintain consistency with standard robotic kinematic conventions, the system can be modeled as a series of three prismatic joints. Table 1 presents established DH parameters for the Cartesian robot.

The forward kinematics problem determines the position and orientation of the end-effector based on known joint variables. Since the robot undergoes only translational motion and the orientation vector remains constant relative to the base frame, the general homogeneous transformation matrix T is expressed as in (1) follows:

$$T = \begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

Hence, the position of the end-effector is a direct result of the joint variables, represented as $P = [q_1, q_2, q_3]^T$. This linear relationship demonstrates that the motions of the X, Y, and Z axes are completely decoupled and independent of one another.

Table 1. DH Parameters for 3-Axis Cartesian Robot

Link(i)	A_i	θ_i	D_i	a_i
1	0	0	$d_1 = x$	0
2	0	0	$d_2 = y$	0
3	0	0	$d_3 = z$	0

2. Inverse Kinematics

The inverse kinematics problem necessitates the determination of the joint variable set(s) corresponding to a specific pose (position and orientation) of the end-effector. The aggregation of all joint values constitutes the robot's configuration within the joint space.

While deriving inverse kinematics from forward kinematics presents significant computational challenges for articulated robotic systems, Cartesian robots offer a distinct exception. Due to the orthogonality and kinematic decoupling of their motion axes, the inverse kinematics solution is trivial to derive.

The governing equations for positional inverse kinematics are formulated as in (2)-(4) follows:

$$q_x = q_1 \quad (2)$$

$$q_y = q_2 \quad (3)$$

$$q_z = q_3 \quad (4)$$

a) Jacobian Matrix

By differentiating the robot's position equations, the following equations (5) and (6) is derived, establishing the relationship between the differential motion of the joints and the differential motion of the end-effector (arm):

$$\begin{bmatrix} d_x \\ d_y \\ d_z \\ \delta_x \\ \delta_y \\ \delta_z \end{bmatrix} = [JACOBI] \begin{bmatrix} d_{\theta_1} \\ d_{\theta_2} \\ d_{\theta_3} \\ d_{\theta_4} \\ d_{\theta_5} \\ d_{\theta_6} \end{bmatrix} \quad (5)$$

$$[D] = [J][D_\theta] \quad (6)$$

where:

- dx, dy, dz represent the differential translation of the end-effector along the x, y, and z axes.
- δ_x, δ_y,δ_z represent the differential rotations about the x, y,and z axes.
- [D] represents the differential motion of the joints.

b) Application to the Problem

Since the Cartesian robot consists solely of prismatic (translational) joints and lacks revolute (rotational) joints, the angular velocity of the robot is zero. Consequently, the system possesses only linear velocity, characterized by dx, dy, dz.

To derive the Jacobian matrix for this three-degree-of-freedom (3-DOF) Cartesian robot, it is necessary to utilize the forward kinematics position equations (7) established in the preceding section:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (7)$$

Finally, by applying the aforementioned expressions, we obtain (8):

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} \quad (8)$$

3. Motion Trajectory Planning

Trajectory planning for Cartesian robots is a critical component in ensuring precise motion control according to specific requirements. This process involves generating a motion path that allows the robot to transit from an initial point to a target point under varying parameters. When defining the trajectory, it is essential to analyze the robot's kinematic profiles, specifically position, velocity, and acceleration, as variations in velocity and acceleration constraints result in distinct motion characteristics.

For Cartesian robots, point-to-point motion is typically implemented. Specifically, the trajectory is modeled using a third-degree (cubic) polynomial subject to specific boundary conditions.

At the initial time, we obtain (9):

$$P_0 = [x_0, y_0, z_0] \tag{9}$$

At the final time, we obtain (10):

$$P_1 = [x_1, y_1, z_1] \tag{10}$$

a) Linear Trajectory Planning

The trajectory for each axis is defined as in (11)-(13) follows:

$$x(t) = x_0 + \frac{(x_1 - x_0)}{T}t \tag{11}$$

$$y(t) = y_0 + \frac{(y_1 - y_0)}{T}t \tag{12}$$

$$z(t) = z_0 + \frac{(z_1 - z_0)}{T}t \tag{13}$$

The trajectory for each independent axis is defined by the following polynomial equation (14) below:

$$q(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \tag{14}$$

where: $q_{(t)}$is the position of the axis at time t; a_0, a_1, a_2, a_3 are the coefficients determined by the boundary conditions.

4. System Dynamics

Robot dynamics is a critical aspect of the design and simulation phases, serving to determine the forces and torques required for the robot to execute prescribed motions.

Dynamic analysis examines the interrelationships between acceleration, payload, mass, and inertial forces. In the context of a Cartesian system, the fundamental principle is that subjecting a specific mass to a linear acceleration generates a corresponding linear force

Since Cartesian robots utilize exclusively translational (prismatic) joints rather than rotary joints, they do not generate angular acceleration. Consequently, significant rotational moments are not induced within the robot's kinematic structure (in (15)):

$$\sum \bar{F} = m\bar{a} \tag{15}$$

The equations of motion govern the behavior of a mechanical system subjected to applied forces. The general dynamic formulation is represented by the (16) below:

$$M(q)\ddot{q} + C(q,\dot{q}) + G(q) = \tau \tag{16}$$

where: M(q): Inertia matrix (or mass matrix) of system; q: The joint position vector; $\dot{q}$: The joint velocity vector; $\ddot{q}$: The joint acceleration vector; C(q,$\dot{q}$): The matrix of Coriolis and centrifugal forces; G(q): The gravity vector; τ: The vector of generalized forces and torques acting on the system.

Based on the aforementioned dynamic principles, the robot's dynamics can be modeled using the Euler-Lagrange formulation, expressed as in (17) follow:

$$\frac{d}{dt}\left(\frac{\partial y}{\partial \dot{q}_1}\right) - \frac{\partial y}{\partial \dot{q}_1} = \tau_i \tag{17}$$

The Lagrangian function L (in (18)) is defined as the difference between the total kinetic energy (K) and the total potential energy (P) of the system:

$$L = K - P \tag{18}$$

Let $\boldsymbol{\tau}$ denote the generalized force or torque acting at the i-th joint. The kinetic energy of the joints moving along the X, Y, and Z axes is determined as follows:

a) Link along the X-axis:

The Kinetic Energy Equation is shown in (19):

$$K_1 = \frac{1}{2}m_1\dot{x}^2 \tag{19}$$

The Potential Energy Equation is shown in (20):

$$P_1 = 0 \tag{20}$$

b) Link along the Y-axis:

The Kinetic Energy Equation is shown in (21):

$$K_3 = \frac{1}{2}m_3(\dot{z}^2 + \dot{y}^2) \tag{21}$$

The Potential Energy Equation is shown in (22):

$$P_2 = m_2 gz \tag{22}$$

c) Link along the Z-axis:

The Kinetic Energy Equation is shown in (23):

$$K_3 = \frac{1}{2}m_3(\dot{z}^2 + \dot{y}^2) \tag{23}$$

The Potential Energy Equation is shown in (24):

$$P_3 = m_3 gz \tag{24}$$

Lagrangian operator is obtained as in (25):

$$\begin{aligned} L &= (K_1 + K_2 + K_3) - (P_1 + P_2 + P_3) \\ &= \frac{1}{2}m_1\dot{x}^2 + \frac{1}{2}m_2\dot{y}^2 \\ &\quad + \frac{1}{2}m_3(\dot{z}_2 + \dot{y}^2) \\ &\quad - (m_2 + m_3)gz \\ &= \frac{1}{2}m_1\dot{x}^2 + \frac{1}{2}(m_2 + m_3)\dot{y}^2 + \frac{1}{2}m_3\dot{z}_2 - (m_2 \\ &\quad + m_3)gz \end{aligned} \tag{25}$$

where: $\frac{d}{dt}\frac{\partial L}{\partial \dot{x}} = m_1\ddot{x}$; $\frac{d}{dt}\frac{\partial L}{\partial \dot{y}} = (m_2 + m_3)\ddot{y}$; $\frac{d}{dt}\frac{\partial L}{\partial \dot{z}} = m_3\ddot{z}$; $\frac{\partial L}{\partial x} = 0$; $\frac{\partial L}{\partial y} = 0$; $\frac{\partial L}{\partial z} = -(m_2 + m_3)g$.

Euler-Lagrange Equations have the form as in (26):

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial \dot{q}} = F \tag{26}$$

where: $F=[F_1\ F_2\ F_3]^T$

The inverse dynamics formulation is shown in (27)-(29):

$$m_1 x'' = F_1 \tag{27}$$

$$(m_2 + m_3)y'' = F_2 \tag{28}$$

$$m_3 z'' + (m_2 + m_3)g = F_3 \tag{29}$$

Deriving the equations of motion, we obtained (30) below:

$$\begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 + m_3 & 0 \\ 0 & 0 & m_3 \end{bmatrix}\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ (m_2 + m_3)g \end{bmatrix} = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix} \tag{30}$$

C. Structure of the Cartesian Robot

An H-frame Gantry structure combined with a ball screw drive was selected to maximize structural rigidity and positional accuracy. Although this configuration involves a trade-off in speed and presents challenges regarding the synchronization of parallel axes, it effectively eliminates the mechanical deflection commonly observed in cantilever-type robots. Consequently, this design is the optimal choice for applications requiring high payload capacities and minimal error tolerances.

Structure: The proposed system features a gantry-style architecture consisting of two parallel linear guides and lead screws acting as the base (Y-axis). A transverse axis (X-axis), also driven by a lead screw mechanism, is mounted atop these parallel Y-axes. The Z-axis is subsequently attached to the X-axis carriage.

Actuation: Linear motion is achieved using Ball Screws (for high precision) or Lead Screws.

Characteristics: The payload of the transverse X-axis is distributed evenly across the two Y-axis supports. This creates a rigid, portal-like structure (Gantry) offering superior stability compared to single-arm designs.

D. Motion Control Implementation via SSCNET III

The motion control architecture is built upon a Mitsubishi Q03UDE paired with a dedicated Motion CPU (or Simple Motion Module) to command MR-J3-10B servo drives via the high-speed SSCNET III (Servo System Controller Network) protocol. Unlike traditional pulse-train control, this fiber-optic communication ensures absolute immunity to electromagnetic interference (EMI) and enables high-speed data synchronization with a cycle time of 0.88 ms. Target coordinates (X, Y) received from the Python-based vision module via MC Protocol are processed using the PLC's dedicated interpolation instructions. For a 2-axis linear path, Motion CPU performs vectorial decomposition to compute the instantaneous velocity for each axis as in (31), (32):

$$V_{x(t)} = V_{\{feed\}} \cdot cos(\theta) \tag{31}$$

$$V_{y(t)} = V_{\{feed\}} \cdot sin(\theta) \tag{32}$$

where v_{feed} represents the programmed engraving speed and θ is the calculated path angle.

This synchronization mechanism ensures that both the X and Y axes reach the target coordinates simultaneously, maintaining a Constant Velocity (CV) profile. Such precision is essential for ensuring uniform laser energy distribution across the workpiece, thereby preventing over-charring or inconsistent engraving depths.

III. Control System Design

Automated control system utilizes a Mitsubishi Programmable Logic Controller (PLC) to regulate the X, Y, and Z axes through three sets of AC servo motors and drivers. Position and velocity control algorithms are programmed to ensure precise and synchronized multi-axis motion.

Additionally, limit switches and force sensors are integrated to verify positioning accuracy and ensure operational safety.

A. Control Algorithm Implementation

1. Independent Joint Control Strategy

The control strategy employed for the Cartesian robot is based on the Independent Joint Control approach. Unlike articulated manipulators, where dynamic coupling and non-linearities are significant, the orthogonal configuration of the Cartesian robot implies that the coupling effects between the X, Y, and Z axes are negligible. Consequently, the system is decoupled into three independent Single-Input Single-Output (SISO) subsystems. A decentralized Proportional-Integral-Derivative (PID) controller is implemented for each axis:

- Proportional (P) term ensures rapid transient response.
- Integral (I) term eliminates steady-state error, which is particularly critical for the Z-axis to compensate for gravitational forces holding the payload.
- Derivative (D) term provides damping to minimize overshoot and suppress mechanical oscillation.

2. Trajectory Generation

To ensure smooth motion and mitigate the "jerk" phenomenon (abrupt changes in acceleration) that leads to mechanical wear and vibration, the reference trajectory is not a simple trapezoidal profile but is generated using a Cubic Polynomial profile (or S-curve velocity profile). This method guarantees continuity in both velocity and acceleration, ensuring stable operation even at high speeds.

3. Hierarchical Control Architecture

The hardware control structure operates on a cascade hierarchy:

a) Centralized Controller (PLC):

The Mitsubishi PLC functions as the trajectory planner. It calculates the instantaneous position setpoints q_d based on the cubic polynomial function and transmits these commands

to the servo drives via high-speed pulse trains (or fieldbus communication).

b) Inner-Loop Controller (Servo Drives):

The AC Servo Drivers execute the high-speed inner control loops. They receive the position commands from the PLC and regulate the motor's torque (current) and velocity to track the desired trajectory precisely, rejecting external disturbances.

B. Flowchart of the Control Algorithm

The algorithm flowchart of the model is illustrated in Fig. 1. The system operates through a closed-loop workflow, initiated by the acquisition of input image data. In this phase, specialized image processing algorithms perform analysis, feature extraction, and surface digitization to transform the raw image into a set of spatial coordinates (X, Y) and corresponding control parameters.

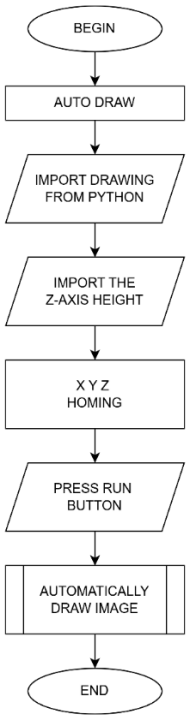

Fig. 1. Algorithm Flowchart

Upon completion of data processing, the generated trajectory (path planning) is transmitted to the central PLC. During the execution phase, the controller actuates drive axes to guide the engraving head along a predefined trajectory. Simultaneously, the pulsed laser source is triggered in synchronization with the traverse speed to scan and interconnect the pixels. This process proceeds continuously and precisely until the image is fully reproduced on the material surface.

At mode AUTO RUN, the system functions in a sequential manner:

- Image Acquisition: In the first step, the target image for laser engraving is imported into the custom-developed image processing interface. Fig. 2 depicts the Graphical User Interface.
- Digital Image Processing (DIP): Next, the loaded image is processed through an image processing pipeline to generate the required (X,Y)coordinates.
- Data Transmission: Upon successful coordinate extraction, the data is transferred from the Python environment to the Mitsubishi Q03UDE PLC using Ethernet communication.
- Motion & Laser Control: The PLC performs 2-axis linear interpolation to drive the X and Y axes synchronously. Concurrently, it synchronizes the laser switching with the motion to execute the engraving task."
- Completion: Once the system verifies that the engraving process is fully completed, the laser head automatically retracts to the 'Home' position for safe product unloading.

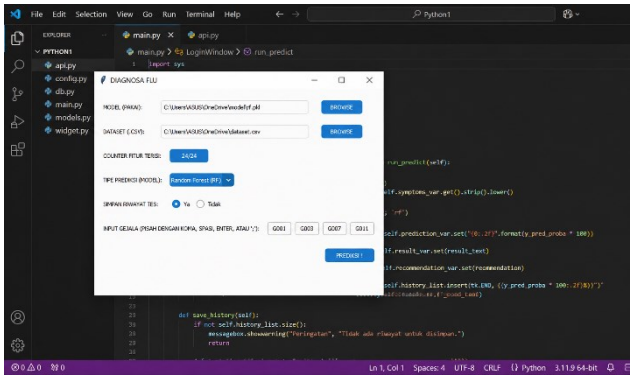

Fig. 2. Graphical User Interface (GUI)

C. Digital Image Processing (DIP)

Following the digital image processing pipeline depicted in Fig. 3.

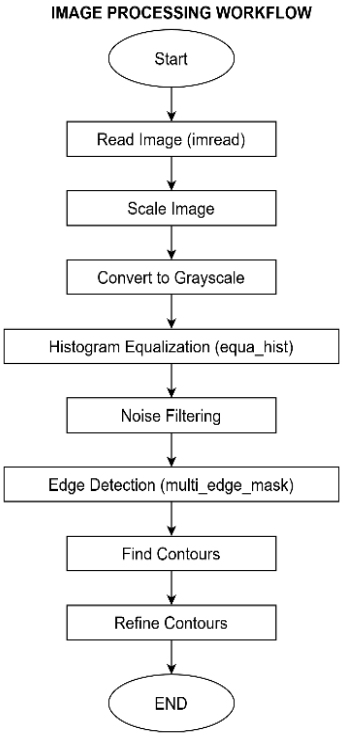

Fig. 3. Image processing workflow

1. Image Reading (imread)

This is a mandatory initial step. Computers do not perceive images as humans do; they require the conversion of data from file formats (e.g., .jpg, .png) into a numerical matrix (pixels) to perform arithmetic operations.

2. *Image Resizing (Scaling)*

- Speed: High-resolution images cause slow processing. Downscaling accelerates computation speed.
- Consistency: Ensuring all input images share a standard size allows parameters in subsequent steps (such as noise filters) to function stably and accurately.

3. *Grayscale Conversion*

- Data Reduction: Color images (RGB) contain three color channels, whereas grayscale images contain a single channel representing light intensity.
- Focus on Structure: For contour detection, color is often less critical than variations in brightness. Converting to grayscale reduces the computational load by $2/3$ while preserving the object's morphological features.

4. *Histogram Equalization (equal_hist)*

- Contrast Enhancement: If an image is underexposed (too dark) or overexposed (too bright), details become obscured.
- Object Highlighting: Histogram equalization redistributes light intensity, clarifying dark regions and preventing glare in bright regions, which facilitates subsequent edge detection steps.

5. *Noise Filtering*

- Removing Digital Artifacts: Images often contain grain (noise) caused by camera sensors or poor lighting conditions.
- Preventing Misinterpretation: Without filtering, the computer might mistake noise for edges or small objects, leading to erroneous results.

6. *Edge Detection (multi_edge_mask)*

- Boundary Definition: This is the most critical step to identify the object's "skeleton." It detects areas with sudden gradients in intensity (lines, table edges, product borders, etc.).
- Eliminating Redundant Information: Instead of processing entire regions, the computer now focuses solely on outlines.

7. *Contour Retrieval*

- Shape Digitization: Following the detection of discrete edges in the previous step, this process connects them to form closed loops (contours).
- Object Identification**:** Each contour represents a specific shape or a potential object that requires recognition.

8. *Contour Filtering*

- Extracting Relevant Information: An image may contain hundreds of contours (cracks, shadows, foreign objects).
- Selection Criteria**:** This step utilizes area, perimeter, or aspect ratio to eliminate insignificant contours (residual noise) and retain only the objects of actual interest.

Digital image processing yielded results shown in Fig. 4.

Fig. 4. Image output after scaling and binary conversion

IV. System Implementation and Experiments

System development workflow proceeded from conceptual design and 3D CAD modeling to component selection, electrical circuit design, mechanical assembly, and software programming.

A. *3D CAD Modeling*

Upon completing the design in SolidWorks, we obtained a machine model as shown in Fig. 5.

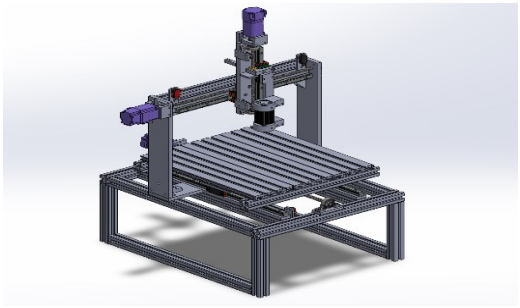

Fig. 5. Mechanical assembly drawing of the Cartesian system

The proposed system is a 3-axis Cartesian robot designed using SolidWorks. The mechanical structure utilizes aluminum extrusion profiles for the frame, ensuring rigidity and ease of assembly.

- Motion Transmission: The linear motion for X, Y, and Z axes is achieved using servo motors coupled with lead screws
- Guidance System: To ensure precision and smooth movement, linear guide rails are employed on all axes.
- Configuration: The robot features a fixed-gantry and moving-table configuration. The Z-axis assembly is mounted on the X-axis carriage, allowing for vertical positioning of the end-effector.

B. *Component Selection*

The control system is built around a PLC combined with necessary expansion modules for I/O management. Precision motion is achieved using servo motors, while the end-effector is equipped with a high-power laser module for processing.

The selection of the Q03UDE PLC is justified by its ability to perform multi-axis interpolation and its Ethernet connectivity for seamless system integration.

Reasons for choosing the QD75MH4 Module:

- Capacity: Controls up to 4 axes (suitable for X, Y, Z motion).
- Reliability: Uses SSCNET III optical fiber for noise-immune and high-speed communication.
- Precision: Supports high-precision linear and circular interpolation.
- Connectivity: Easy integration with Mitsubishi MR-J3/J4 Servo Amplifiers via simple daisy-chain wiring.

Reasons for selecting HF-KP13 (100W, 0.32Nm):

- Optimal Torque Matching: Rated torque of 0.32 Nm satisfies the calculated load requirements.
- High Dynamic Response: Low inertia design ensures fast acceleration/deceleration.
- Compact Form Factor: Lightweight and small, ideal for mounting on moving axes.
- Compatibility: Seamless integration with the MR-J3 amplifier and QD75MH4 controller.

C. Electrical Circuit Design

The connection diagram was designed in Fig. 6 to ensure the system operates in accordance with the algorithm flowchart.

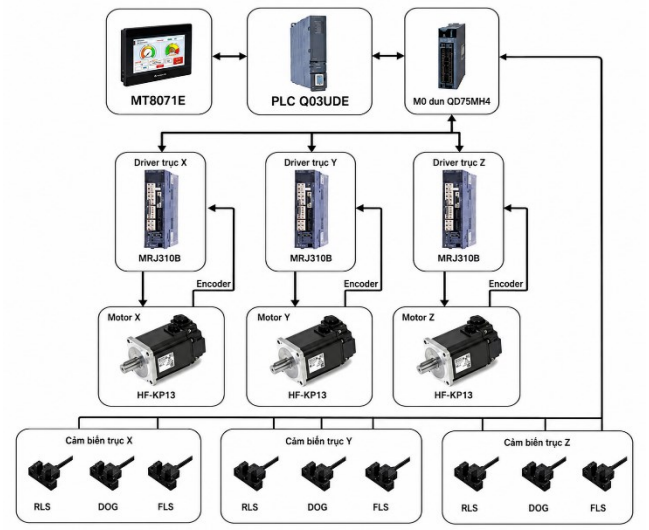

Fig. 6. Electrical circuit design

1. Control System Architecture

Control system for the 3-axis Cartesian robot is designed based on a hierarchical architecture, utilizing industrial-grade components from Mitsubishi Electric to ensure high precision and reliability. The system configuration is illustrated in Fig. 7 and comprises three main levels:

a) Supervisory and Control Level

- Human-Machine Interface (HMI): A Weintek MT807iE touch panel serves as the operator interface, facilitating parameter configuration, real-time monitoring, and manual control commands.
- Central Processing Unit: The core logic control is executed by a Mitsubishi Q03UDE Programmable Logic Controller (PLC). This unit coordinates the overall system operation and processes high-level commands from the HMI.
- Motion Control Module: Precision motion tasks are managed by the QD75MH4 positioning module. This unit is capable of controlling up to four axes and is responsible for generating high-speed motion profiles.

b) Drive and Actuation Level

- Servo Amplifiers: The system employs three MR-J3-10B servo amplifiers corresponding to the X, Y, and Z axes. These drivers communicate with the positioning module via the high-speed SSCNET III optical network, which significantly reduces electrical noise interference and ensures superior synchronization compared to traditional pulse-train control.
- Servo Motors: Actuation is provided by HF-KP13 low-inertia AC servo motors. These motors are well-suited for dynamic applications requiring rapid acceleration.
- Feedback Mechanism: Each motor is equipped with a high-resolution encoder that provides real-time position and speed feedback to the driver, establishing a closed-loop control system to maintain positional accuracy.

c) Sensor and Safety Interface

To ensure operational safety and precise calibration, the system integrates a set of digital sensors for each axis (X, Y, Z). These signals are fed directly into the motion controller:

- FLS (Forward Limit Switch) & RLS (Reverse Limit Switch): Prevent the robot from exceeding its mechanical travel limits.
- DOG (Near-point Dog): Used for the homing sequence to establish the machine's zero coordinate.

2. Wiring Diagram

The wiring schematic was developed from the connection diagram, as illustrated in Fig. 7. Based on the connection architecture, a detailed electrical wiring diagram was developed for the system components. To ensure operational reliability, a 20 A Miniature Circuit Breaker (MCB) is dedicated to each axis (X, Y, and Z) to provide independent protection. This isolation strategy ensures that a fault occurring in a single axis does not trigger a system-wide shutdown, thereby facilitating efficient fault diagnosis and maintenance.

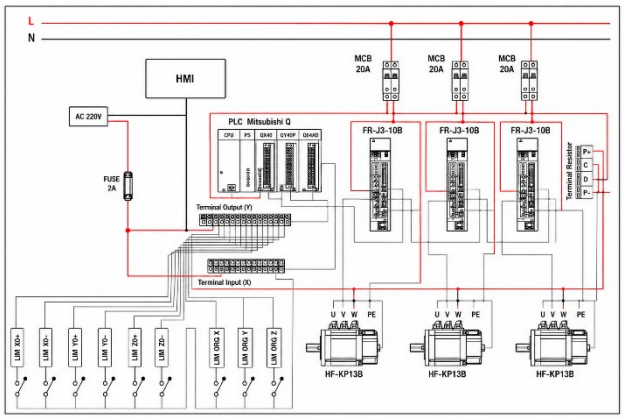

Fig. 7. Writing Diagram

D. HMI Configuration

The HMI interface is illustrated in Fig. 8 below.

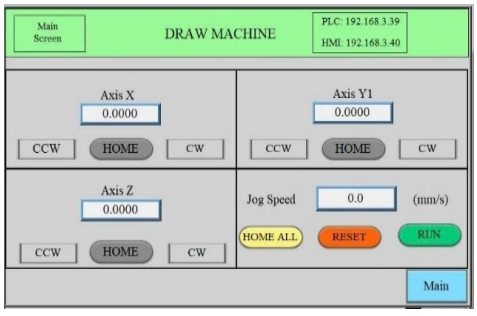

Fig. 8. HMI Configuration

1. Individual Axis Control:

Interface provides dedicated control blocks for Axis X, Axis Y1, and Axis Z. Each block features a high-precision numerical display (four decimal places) for real-time position monitoring. Operators can manually manipulate each axis using the CW (Clockwise) and CCW (Counter-Clockwise) buttons for jogging, or initiate a homing sequence using the HOME button.

2. System Operation Panel:

The bottom-right quadrant contains global operational parameters. This includes a field for setting the Jog Speed (in mm/s) and essential command buttons: HOME ALL for simultaneous homing of all axes, RESET for error clearance, and RUN to initiate the automatic operation cycle.

E. Experiments

Upon completion of the mechanical fabrication, the assembled prototype is shown in Fig. 9 and Fig. 10.

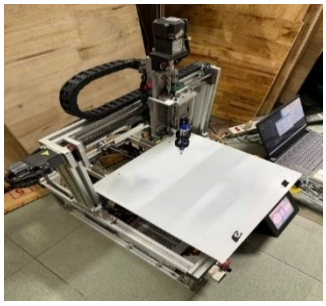

Fig. 9. Machine Prototype

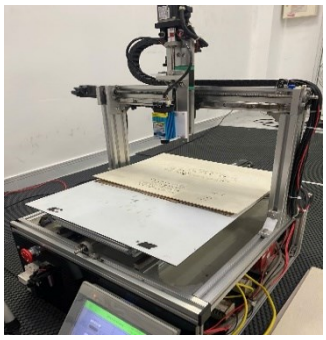

Fig. 10. Final Assembly

1. System Evaluation and Summary

The experimental model of the Cartesian laser robot has been successfully fabricated and integrated, demonstrating a high level of industrial standardization.

- Mechanical Integrity: The robot utilizes a robust aluminum gantry structure driven by precision lead screws on all axes (X, Y, Z). This design choice significantly minimizes mechanical backlash and vibration, ensuring high-fidelity engraving results compared to belt-driven alternatives.
- Control Reliability: The integration of a Mitsubishi Q-Series PLC combined with the QD75MH4 positioning module and MR-J3 AC Servo systems provides a stable, closed-loop control architecture. This setup offers superior noise immunity and real-time responsiveness essential for precision machining.
- Operational Safety: The system is equipped with a comprehensive safety feature, including hardware limit switches (FLS/RLS) and emergency stop mechanisms. The HMI provides a user-friendly interface for seamless monitoring and parameter adjustment.

In essence**:** The fabricated prototype meets the design requirements for rigidity, precision, and operational safety, serving as a valid platform for further research into laser engraving algorithms and automated manufacturing applications.

F. Laser Engraving Results Produced by the System

The experimental results, showing the transition from the original input image to the final engraving on the wooden workpiece, are presented in Fig. 11 to Fig. 13.

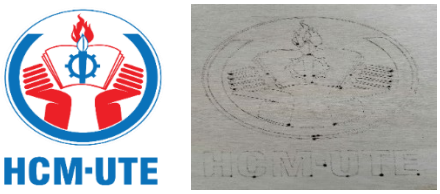

Fig. 11. Original HCMUTE logo and the laser-engraved product

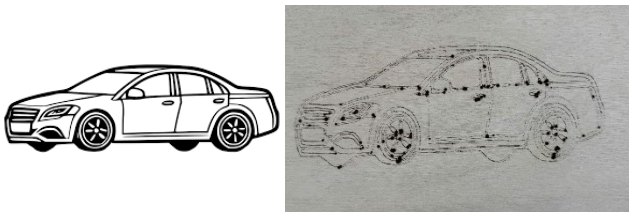

Fig. 12. The original car image vs. the engraved result

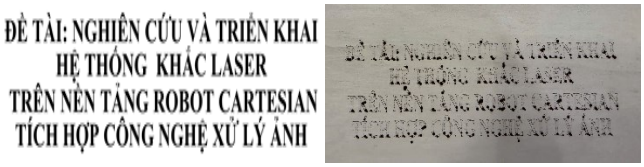

Fig. 13. Project Title (in Vietnamese) and Experimental Results

2. Evaluation of Experimental Results

Upon comparing the original source files with the laser-engraved specimens on wood, the following observations regarding the system's performance were established:

- Geometric Fidelity: The experimental results, specifically the automotive sketch and the HCMUTE logo, demonstrate exceptional geometric fidelity. Curved trajectories, such as the concentric circles in the logo and the wheel arches of the car, are rendered smoothly without visible stair-step artifacts or jagged edges. This validates the mechanical rigidity of the lead screw transmission system and the efficacy of the motion controller's interpolation algorithms.
- Dimensional Accuracy: The aspect ratio of the engraved patterns remains consistent with the original design. There is no evidence of axial distortion (e.g., circular geometries appearing elliptical), indicating precise calibration of the servo drive steps per millimeter ratio for both X and Y axes.
- Surface Finish and Definition: The text engraving sample shows high legibility with sharp character definition. The path following is precise, with no deviation from the intended baseline. Minor carbonization is observed at the vertices of the vector paths, which is a typical thermal interaction of diode lasers during deceleration phases, suggesting potential for further optimization in laser power modulation.

Conclusion: The fabricated Cartesian robot prototype successfully meets the design criteria for precision laser machining. The combination of industrial servo control and rigid mechanical guidance ensures high repeatability and stable operation for complex graphical engravings.

V. CONCLUSION

This study successfully designed, fabricated, and experimentally evaluated a 3-axis Cartesian robot for laser engraving, distinguished by an industrial control system (Mitsubishi Q-Series PLC, MR-J3 Servo) combined with a lead screw transmission mechanism to ensure high reliability and sharp engraving quality on wood. Although the system achieved superior precision and rigidity compared to microcontroller-based solutions, the use of industrial components and lead screws resulted in limitations regarding high investment costs, low raster engraving speeds, limited diode laser power for metals or acrylics, and frequent maintenance requirements due to dust accumulation. To address these drawbacks, future work will focus on integrating a Computer Vision system for automatic workpiece localization and optimizing the laser power control algorithm to enhance product uniformity. Additionally, an automatic feeding system will be developed to fully automate the production process.

ACKNOWLEDGEMENT

This research was funded by Ho Chi Minh City University of Technology and Engineering (HCM-UTE), Vietnam, under grant No. SV2026-437. We want to give thanks to Assoc. Prof. Van-Phong Vu (HCM-UTE) due to his supervision and PhD. Van-Dong-Hai Nguyen (HCM-UTE) due to his help in editing the paper. We, authors, are grateful to these supports.

REFERENCES

[1] L. Li Da Xu Li, Eric L. Xu, and Li, "State of the Art and Future Trends," *International Journal of Production Research*, vol. 56, no. 8, pp. 2941–2962, 2018, https://doi.org/10.1109/TII.2018.2828141.

[2] A. Gasparetto and G. Rosati, "Design and Implementation of a Cartesian Robot," in *AMST'02 Advanced Manufacturing Systems and Technology*, pp. 539–544, 2002, https://doi.org/10.1007/978-3-7091-2555-7_61.

[3] J. Zhang, D. Cao, and Y. Wu, "Kinematic analysis and motion planning of cable-driven rehabilitation robots," *Applied Sciences (Switzerland)*, vol. 11, no. 21, 2021, https://doi.org/10.3390/app112110441.

[4] L. Cao *et al.*, "2 µm Passively Q-switched all-solid-state laser based on WSe2 saturable absorber," *Optics and Laser Technology*, vol. 113, pp. 72–76, 2019, https://doi.org/10.1016/j.optlastec.2018.12.015.

[5] I. G. S. M. Diyasa, A. Fauzi, M. Idhom, and A. Setiawan, "Multi-face Recognition for the Detection of Prisoners in Jail using a Modified Cascade Classifier and CNN," *Journal of Physics: Conference Series*, vol. 1844, no. 1, p. 12005, 2021, https://doi.org/10.1088/1742-6596/1844/1/012005.

[6] A. S. Shaikat, S. Akter, and U. Salma, "Computer Vision Based Industrial Robotic Arm for Sorting Objects by Color and Height," *Journal of Engineering Advancements*, vol. 01, no. 04, pp. 116–122, 2020, https://doi.org/10.38032/jea.2020.04.002.

[7] A. Pushpa Latha, R. Tharun Teja, B. Vamsi, P. Uha, P. Varshini, and M. Gnana Priya, "OpenCV-Enabled Real-Time Object Tracking Robotic Manipulator with Eye-Gaze Control for Assistive Applications," *International Journal of Emerging Research in Science, Engineering, and Management*, vol. 2, no. 4, pp. 109–113, 2026, https://doi.org/10.58482/ijersem.v2i4.15.

[8] Z. Jin and H. Yang, "Real time object tracking using deep learning and OpenCV," *Applied and Computational Engineering*, vol. 35, no. 1, pp. 272–279, 2024, https://doi.org/10.54254/2755-2721/35/20230406.

[9] N. F. Mohamad, A. R. Hidayu, A. A. Sherif, and A. S. A. K. Sharifah, "Characteristics of bituminous coal, sub-bituminous coal and bottom ash from a coal-fired power plant," in *BEIAC 2013 - 2013 IEEE Business Engineering and Industrial Applications Colloquium*, pp. 571–573, 2013, https://doi.org/10.1109/BEIAC.2013.6560193.

[10] L. Van Dai, N. N. Bon, and L. C. Quyen, "Deep Learning Method for Fault Diagnosis in High Voltage Transmission Lines: A Case of the Vietnam 220kV Transmission Line," *International Journal on Electrical Engineering and Informatics*, vol. 14, no. 2, pp. 254–275, 2022, https://doi.org/10.15676/ijeei.2022.14.2.1.

[11] Z. Zhaolei, M. Zhiqiang, and S. Lei, "The performance of typical QKD scheme under the condition of quantum measurement noise," *Journal of Physics: Conference Series*, vol. 1634, no. 1, p. 12117, 2020, https://doi.org/10.1088/1742-6596/1634/1/012117.

[12] H. Yoshida, Y. Hasegawa, M. Matsushima, T. Sugiyama, T. Kawabe, and M. Shikida, "Miniaturization of Respiratory Measurement System in Artificial Ventilator for Small Animal Experiments to Reduce Dead Space and Its Application to Lung Elasticity Evaluation," *Sensors*, vol. 21, no. 15, p. 5123, 2021, https://doi.org/10.3390/s21155123.

[13] C-Y. Lin and Y-C. Liu, "Precision tracking control and constraint handling of mechatronic servo systems using model predictive control," *IEEE/ASME Transactions on Mechatronics*, vol. 17, no. 4, pp. 593–605, 2012, https://doi.org/10.1109/TMECH.2011.2111376.

[14] P. Li, V. Marik, L. Gao, and W. Shen, "Editorial for the Special Issue on Intelligent Manufacturing," *Engineering*, vol. 5, no. 4, pp. 595–596, 2019, https://doi.org/10.1016/j.eng.2019.07.004.

[15] C. Sharma, S. K. Sharma, and D. Gill, "Holistic Smart Cities Viewed Through the Lens of Performance Evaluation Schemes," in *3rd IEEE 2022 International Conference on Computing, Communication, and Intelligent Systems, ICCCIS 2022*, pp. 983–989, 2022, https://doi.org/10.1109/ICCCIS56430.2022.10037622.

[16] Z. H. Khan *et al.*, "Development of a Low-Cost CNC Machine Laser Engraver," in *2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, IEEE, pp. 0376–0382, 2021, https://doi.org/10.1109/UEMCON53757.2021.9666661.

[17] A. Patwardhan, A. Prakash, and R. G. Chittawadigi, "Kinematic Analysis and Development of Simulation Software for Nex Dexter Robotic Manipulator," *Procedia Computer Science*, vol. 133, pp. 660–667, 2018, https://doi.org/10.1016/j.procs.2018.07.101.

[18] Z. Liu, M. Li, D. Fu, and S. Zhang, "Design of intelligent controller for obstacle avoidance and navigation of electric patrol mobile robot based on PLC," *Scientific Reports*, vol. 14, no. 1, p. 13476, 2024, https://doi.org/10.1038/s41598-024-63810-5.

[19] E. A. Gyasi, A. Antila, P. Owusu-Ansah, P. Kah, and A. Salminen, "Prospects of Robot Laser Cutting in the Era of Industry 4.0," *World Journal of Engineering and Technology*, vol. 10, no. 03, pp. 639–655, 2022, https://doi.org/10.4236/wjet.2022.103042.

[20] R. Qi *et al.*, "A new Cartesian cutting robot with laser height control system," in *2011 IEEE International Conference on Mechatronics and Automation, ICMA 2011*, pp. 2153–2158, 2011, https://doi.org/10.1109/ICMA.2011.5986314.

[21] X. Xie, S. Ge, M. Xie, F. Hu, and N. Jiang, "An improved industrial sub-pixel edge detection algorithm based on coarse and precise location," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 5, pp. 2061–2070,2020, https://doi.org/10.1007/s12652-019-01232-2.

[22] Y. Li, T. Huang, and D. G. Chetwynd, "An approach for smooth trajectory planning of high-speed pick-and-place parallel robots using quintic B-splines," *Mechanism and Machine Theory*, vol. 126, pp. 479–490, 2018, https://doi.org/10.1016/j.mechmachtheory.2018.04.026.