

Using artificial intelligence to detect fires in woodworking factory and designing the robotic arm control of the spray system in a mobile firefighting robot

Nguyen Khanh Nam

Nishimura Iron Works Co. Ltd., 286-4 Kakinase city, Ogi, Saga province, Japan

Sử dụng trí tuệ nhân tạo để nhận diện đám cháy xưởng gỗ và thiết kế điều khiển tay máy của lăng phun trong robot di động chữa cháy

Nguyễn Khánh Nam

Công ty TNHH thép Nishimura, 286-4 thành phố Kakinase, Ogi, tỉnh Saga, Nhật Bản

Corresponding author: khanhnamng@gmail.com

<https://doi.org/10.55250/jo.vnuf.10.1.2025.087-099>

ABSTRACT

The purpose of this study is to address the challenges of fire detection and suppression in woodworking factories, where the risk of fire is high due to flammable materials such as wood, wood waste, paint, and wood-based panels, as well as fire sources like burners and drying kilns. Firefighting is particularly difficult in these environments due to confined spaces and numerous obstacles. This study aims to enhance the speed and accuracy of fire detection while improving the efficiency of the fire suppression system, thereby increasing both safety and operational effectiveness. The research method utilizes deep learning techniques, specifically convolutional neural networks (CNNs), to detect fires quickly and accurately. Automated control algorithms guide the robotic system to precisely locate the fire and adjust parameters such as spray pressure, flow rate, and distance, reducing reliance on manual control. The results of the study demonstrate that the CNN-based system significantly improves fire detection accuracy and speed, while the automated spray system optimizes water usage and enhances suppression efficiency. The conclusion highlights that integrating artificial intelligence technology into firefighting systems can improve safety, reduce response times, and lower costs, while also expanding the potential applications of AI in other high-risk industries.

Article info:

Received: 07/02/2025

Revised: 12/03/2025

Accepted: 14/04/2025

Keywords:

Fire detection, fire prevention and control, mobile robot, robotic arm control, spray system, woodworking factory.

Từ khóa:

Điều khiển tay máy, lăng phun, mobile robot, nhận diện đám cháy, phòng và chống cháy, xưởng gỗ.

TÓM TẮT

Mục đích của nghiên cứu này là giải quyết thách thức trong việc phát hiện và dập tắt hỏa hoạn tại các nhà máy chế biến gỗ, nơi có nguy cơ cháy cao do vật liệu dễ cháy như gỗ, rác thải gỗ, sơn và ván gỗ nhân tạo, cùng với các nguồn gây cháy như lò đốt và lò sấy. Việc dập tắt cháy gặp khó khăn do không gian chật hẹp và nhiều vật cản. Nghiên cứu này nhằm cải thiện tốc độ và độ chính xác phát hiện cháy, đồng thời nâng cao hiệu quả hệ thống dập lửa, từ đó tăng cường an toàn và hiệu quả vận hành. Phương pháp nghiên cứu sử dụng kỹ thuật học sâu, đặc biệt là mạng nơ-ron tích chập (CNNs), để phát hiện cháy nhanh chóng và chính xác. Các thuật toán điều khiển tự động giúp hệ thống robot xác định chính xác vị trí cháy và điều chỉnh các thông số phun nước như áp suất, lưu lượng và khoảng cách, giảm sự phụ thuộc vào điều khiển thủ công. Kết quả nghiên cứu cho thấy hệ thống CNN cải thiện đáng kể độ chính xác và tốc độ phát hiện cháy, trong khi hệ thống phun nước tự động tối ưu hóa việc sử dụng nước và nâng cao hiệu quả dập tắt. Kết luận cho thấy việc áp dụng công nghệ trí tuệ nhân tạo vào hệ thống chữa cháy giúp cải thiện an toàn, giảm thời gian phản ứng và tiết kiệm chi phí, đồng thời mở rộng khả năng ứng dụng AI trong các ngành công nghiệp có nguy cơ cháy cao khác.

1. INTRODUCTION

Fire is one of the most devastating disasters, ranking second among common threats. In 2023, Vietnam recorded 3,440 fire incidents, resulting in 146 deaths, 109 injuries, and property damage amounting to 878 billion VND, along with the destruction of 236 hectares of forest [1]. Fires also pose serious risks to firefighting personnel. However, current fire prevention and control efforts remain largely ineffective. To enhance their effectiveness, several key issues must be addressed:

(i) Firefighter safety: Firefighting operations are still conducted directly by humans, exposing them to significant danger, especially in confined areas filled with flammable materials; (ii) Fire detection (presence and location): Traditionally, fire detection has relied on infrared, ultrasonic, laser, and optical sensors. These point sensors are only triggered when particles from a fire source reach the sensor itself, causing potential delays. Moreover, they are ill-suited for large, open, or dynamic environments. Fire and smoke possess both static and dynamic features, such as color and motion, which point sensors do not utilize in detection [2]; (iii) Fire suppression system control (adjusting spray distance, angle, and pressure): These adjustments are usually made manually, resulting in slow and inefficient responses [2].

Proposed solutions include:

(i) Utilizing computer vision-based smoke and flame detection systems to overcome the limitations of point-sensor methods, enabling faster fire detection, accurate localization, and quicker deployment of suppression measures; (ii) Applying artificial intelligence, particularly deep learning models such as convolutional neural networks (CNNs), to improve detection accuracy and speed. Control algorithms allow firefighting robots to determine optimal movement paths and precisely manage fire suppression nozzles, ensuring effective performance in high-risk environments [2].

These approaches have already shown promise in practical applications, contributing to better fire prevention and control.

Woodworking factory fires pose specific challenges due to several factors:

(i) Most woodworking facilities in Vietnam are small- to medium-sized with low levels of automation [3], making detection and response more difficult. Limited automation increases operational errors, human negligence, and risks associated with outdated equipment; (ii) The materials commonly used - wood, MDF, particleboard, adhesives, coatings, sawdust, and wood shavings - are highly flammable, with low ignition temperatures and rapid combustion rates [4]; (iii) Internal heat sources such as drying kilns, processing machinery, and sanding equipment further elevate fire risk; (iv) Cramped working conditions and disorganized layouts obstruct firefighting equipment and personnel, delaying response times and complicating suppression efforts; (v) Fires in these environments often occur under fluctuating lighting conditions, shadows, or among objects resembling flames, which makes manual fire location and traditional detection methods based on feature extraction (fixed sensors or cameras) time-consuming and prone to error [2]. Despite these risks, no targeted research has yet been conducted specifically for fire prevention in woodworking factories. This study addresses that gap by: (i) Applying AI for fire detection, and (ii) Implementing control algorithms to manage a firefighting robot's arm (e.g., positioning to the fire center, adjusting spray angle) under the unique conditions found in woodworking facilities.

2. RESEARCH METHODS

2.1. Fire detection in woodworking factories

2.1.1. Fire detection in woodworking factories

The fire detection process is carried out through the following steps: (i) The camera installed at the woodworking factory collects sufficient image data of fires occurring in the

factory and using a labeling application to annotate the collected images. ; (ii) Training the fire detection model using YOLOv8 on Google Colab; (iii) Detecting fires based on the trained model.

2.1.2. Fire Location Identification Relative to the Camera for Controlling the robotic arm control of the spray system in a mobile firefighting robot

The process of determining the fire's position relative to the camera to control the robotic arm of spray system mounted on the robot is carried out through the following steps: (i) Determining the fire's center: First, the camera on the robot is connected to detect the fire, and the result is used to determine the fire's center based on XYZ coordinates in the image; (ii) Determining the camera's rotation angle: The camera is fixed along the Oz axis at an angle relative to the wall; (iii) Calculating the distance from the camera to the fire's center: After aligning the bounding box containing the detected fire with the camera's center, the distance from the camera to the fire is calculated based on the following parameters: Camera placement position, Fire center position, Fire pixel position relative to the floor, Angle of the camera relative to the wall, Distance from the camera to the fire, and Camera height above the ground

2.1.3. Determining the Rotation Angle (ϑ) of the Firefighting Robot's spray system

The spray system's rotation angle is determined based on the following parameters: Camera placement position. Fire position in the image and actual fire position; Camera focal length, and Distance from the camera center to the fire center in the image.

2.2. Design of the robotic arm control of the spray system of a mobile firefighting robot

Based on the dynamic equations, the control problem involves moving the spray system from its initial position (0,0) to a specific angle (θ_1 , θ_2) after a certain period. These

angle values are determined from image analysis. The design of the robotic arm control of the spray system control follows the PD control law and the PD + G control law [5].

3. RESULTS AND DISCUSSION

3.1. Fire Detection

Fire detection must answer three key questions: (i) Is there a fire or not? (ii) If there is a fire, what are its coordinates (relative to the firefighting equipment)? and (iii) What is the extent of the fire? Previously, fire detection relied on thermal, smoke, and laser sensors, which often lacked accuracy. Additionally, manually determining the fire's coordinates was prone to errors [6]. Nowadays, the use of AI for fire detection is becoming an effective trend. AI can analyze data from sensors and surveillance cameras to detect signs of fire, such as temperature changes or smoke, and provide immediate alerts. Moreover, AI can quickly and accurately determine the fire's central coordinates and its extent, transmitting this information to the central processing unit attached to the firefighting robot. For detecting fires in a woodworking factory, the following steps were taken: (i) Step 1: Collecting a sufficient number of images. A total of 2,911 images in .jpg format with a resolution of 640 pixels were pictured from fire incidents in the woodworking factory; (i) Step 2 - Labeling bounding box data for each image: Bounding box data was labeled for each image using a labeling app. Based on the obtained data, the fire data in each image was identified. The output file contains the fire coordinates and is saved in .txt format. In the .txt file, the first number represents the class.

The diagram helps create a prior bounding box with a predefined width p_w and height p_h from a grid cell at coordinates (c_x, c_y) . The center coordinates (b_x, b_y) , width and height b_h , b_w of the bounding box are then calculated using the formulas above.

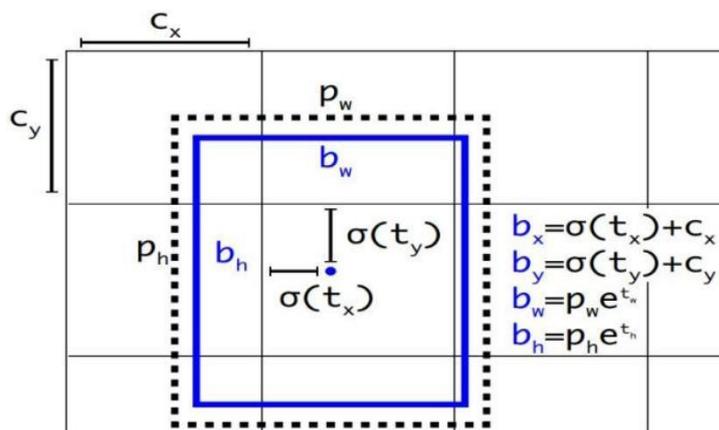


Figure 1. Bounding Box Creation Diagram

(ii) Step 3 - Training the Detection Model with YOLOv8 to obtain a model with stable accuracy:

YOLO (You Only Look Once) is a CNN-based model for object detection, recognition, and classification. It is built by combining convolutional layers (Conv) and fully connected layers. The convolutional layers extract image features, while the fully connected (FC) layers predict the object's probability and coordinates. The model takes an image as input, detects whether an object is present, and determines its coordinates within the image. YOLOv8 uses a Darknet-53 neural network architecture to extract image features and applies the YOLOv8 object detection algorithm to those features. Each YOLO output head performs two tasks: classification and regression. The shared head processes both

tasks on the same branch, while the separate head handles them on two (or more) different branches. A drawback of the shared head is the conflict between classification and regression, which affects overall model accuracy. That's why most object detection models, whether single-stage or two-stage, use separate heads. A shared head reduces detection performance, while replacing it enhances model convergence speed. The separation process involves replacing a $1 \times 11 \times 1$ conv layer into four $1 \times 1 \times 1$ conv layers and four $3 \times 3 \times 3$ conv layers. Steps for training the YOLOv8 Detection Model:

- (a) Create a ZIP file named "data" in Google Drive, containing image files and .txt annotation files.
- (b) Set up the necessary libraries and dependencies for YOLOv8.

```
!pip install ultralytics
!pip install git+https://github.com/ultralytics/ultralytics.git@main
```

(c) unzip image file

```
%cd /content/drive/MyDrive/data
!unzip /content/drive/MyDrive/data/firedata.zip
```

(d) Organize .jpg and .txt files into separate folders: images and labels.

```
%cd /content/drive/MyDrive/data
!mkdir train
!mkdir train/images
!mkdir train/labels
!mv *.jpg train/images
!mv *.txt train/labels
```

(e) Create a yaml file containing the necessary information for training.

(f) Use code to train with epochs = 100.

```
!yolo task=detect mode=train model=yolov8n.pt data=data/mydataset.yaml epochs=100
imgsz=640
!echo 'train: /content/drive/MyDrive/data/train' >> data/mydataset.yaml
!echo 'val: /content/drive/MyDrive/data/train' >> data/mydataset.yaml
!echo 'nc: 1' >> data/mydataset.yaml

!echo "names: ['fire']" >> data/mydataset.yaml
```

With the requirement that the file with the .pt extension contains the trained model with stable accuracy. After training, the results obtained are as follows:

- The file best.pt is stored in the directory: runs/detect/train
- Evaluation of Model Accuracy

(a) Confusion Matrix

A confusion matrix is commonly used to evaluate the performance of a model by comparing its predicted values with the actual values in the test data. From this, we can assess metrics like

```
Speed: 0.4ms preprocess, 3.1ms inference, 0.0ms loss, 3.6ms postprocess per image
Results saved to runs/detect/train
```

💡 Learn more at <https://docs.ultralytics.com/modes/train>

accuracy, sensitivity, specificity, etc. The confusion matrix typically looks like this:

		Predicted Positive	Predicted Negative
Actual	Positive	True Positive (True Prediction is Positive)	False Negative (False Prediction is Negative)
	Negative	False Positive (False Prediction is Positive)	True Negative (True Prediction is Negative)

Figure 2. Confusion Matrix with 4 Metrics

(a) Using the confusion matrix, you can calculate P, R, F Score

$$P_r = \frac{TP_0}{TP_0 + FP_0} \quad (1)$$

$$R_e = \frac{TP_0}{TP_0 + FN_e} \quad (2)$$

$$F_\beta = (1 + \beta^2) \frac{P_r \cdot R_e}{\beta^2 \cdot P_r + R_e} \quad (3)$$

Where,

- P_r : Precision;
- TP_0 : True Positive;
- FP_0 : False Positive;
- FN_e : False Negative;
- R_e : Recall

From (3), can see that: When $\beta > 1$, R_e is given more importance than P_r ; when $\beta < 1$, P_r

is given more importance.

Two commonly used values for β are $\beta = 2$ and $\beta = 0.5$. Here, β is chosen as 0.5.

(b) Actual evaluate the model performance using the confusion matrix

Substituting the values into equations (1) and (2):

$$P_r = \frac{360}{360 + 50} \approx 0.878$$

$$R_e = \frac{360}{360 + 94} \approx 0.793$$

and $F_{0.5}$ score is:

$$F_{0.5} = (1 + 0.5^2) \frac{0.878 \cdot 0.793}{0.5^2 \cdot 0.878 + 0.793} \approx 0.859$$

Thus, the model ensures accuracy.

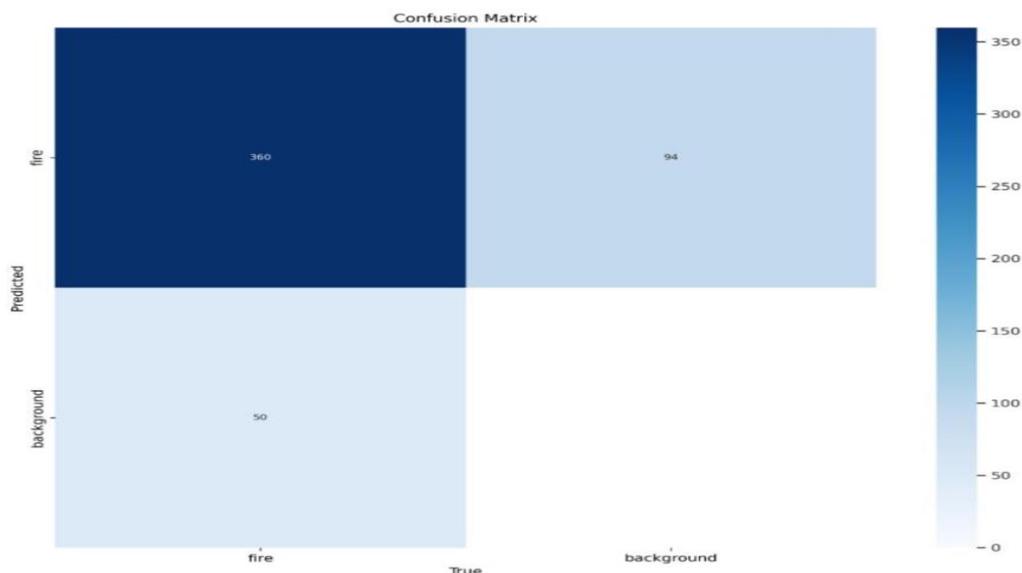


Figure 3. Model evaluation results using the confusion matrix

3.2. Determining the Fire location relative to the camera to control the robotic arm of spray system mounted on the robot

To determine the fire's position relative to the camera to control the robotic arm of spray system mounted on the robot, the following

(ii)

```

model = YOLO('D:/New folder/pythonProject/best.pt')

# Open a connection to the camera
cam = cv2.VideoCapture(0)

# Check if the camera is opened successfully
if not cam.isOpened():
    print("Error: Couldn't open camera.")
    exit()

# Create a file to store bounding box information
output_file_path = "D:/New folder/pythonProject/result/bounding_boxes.txt"
    
```

From the obtained results, the center of the fire will be determined:

$$objX = (\det[\theta,0] + \det[\theta,2])/2$$

$$objY = (\det[\theta,1] + \det[\theta,3])/2$$

(iii) Determine the camera rotation angle:

The camera is fixed along the Oz axis at an angle of 45° to the wall.

steps are required:

(i) **Identify the Fire Center:**First, connect the camera using the command "cam = cv2.VideoCapture (0)"; then, the camera detects the fire and returns the results to a .txt file.

(iv) Determine the distance (L) from the camera to the fire center.

After applying the bounding box containing the detected fire to the center of the camera, the distance (L) from the camera to the fire will be calculated (Figure 4).

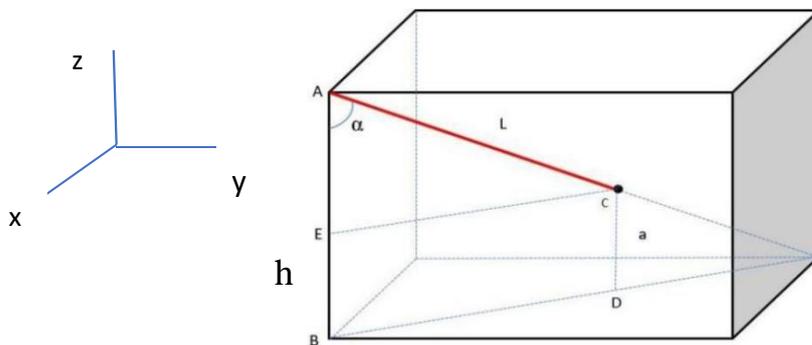


Figure 4. The distance from the webcam to the fire

Where, A is the location of the webcam; B is the location of the floor; C is the center of the fire; D is the pixel point of the fire located below the floor; α is the angle formed by the camera relative to the wall; L is the distance from the camera to the fire; and α is It is the camera installation angle (the angle formed between the camera's axis and the wall where the camera is mounted)
 $AB = h = 5m$ is the height of the camera from the ground.

From Figure 4, the distance L can be determined:

$$L = \frac{h - a}{\cos \alpha} \quad (4)$$

3.3. Design of the robotic arm of the spray system Control

3.3.1. Dynamics of the robotic arm of the spray system

3.3.1.1. Coordinate System of the robotic arm of the spray system

According to the coordinate system placement rule, the origin of the i-th coordinate system is attached to the i-th joint and is placed at the intersection of the common perpendicular line between the i+1-th joint axis and the i-th joint axis with the i+1-th joint axis

itself. In the case where the axes of the joints intersect, the origin will be placed at that intersection point. If the two axes are parallel, the origin will be chosen as any point on the i+1-th joint axis. The Zi-axis of the i-th coordinate system runs along the i+1-th joint axis. The Xi-axis of the i-th coordinate system runs along the common perpendicular line directed from the i-th joint to the i+1-th joint. In the case where the two axes intersect, the Xi-axis will align with the vector direction $Z_i \times Z_{i-1}$, which is perpendicular to the plane containing Z_i and Z_{i-1} . By applying this coordinate system placement principle, we have the coordinate system of the robotic arm of the spray system as shown in Figure 5.

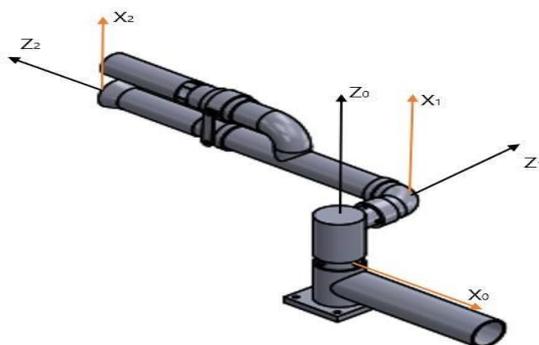


Figure 5. Coordinate system for constructing the dynamic equation of the robotic arm of the spray system

Based on the mechanical structure of the robotic arm [7], the dynamic equation is constructed using the coordinate system shown in Figure 6. The robotic arm has two joints: Joint 1 rotates around the z-axis, and

Joint 2 rotates around the y-axis, with gravitational acceleration directed downward along the z-axis. The physical quantities of the robot arm are defined as shown in Figure 6.

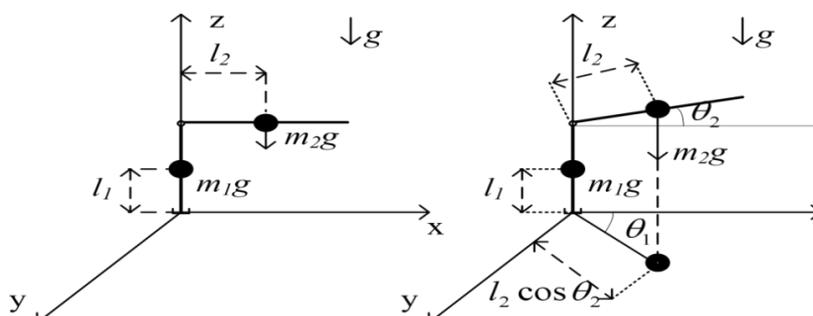


Figure 6. Coordinate system for defining the physical quantities of the robotic arm

Where,

• m_1, m_2 are the masses of arm1 and arm2 of the robotic arm, placed at the center of mass of each arm. The mechanical system is idealized by placing the center of mass of each arm at the midpoint of each arm.

- l_1, l_2 are the length from the pivot of each arm to the center of mass of the arm;
- θ_1, θ_2 are the rotation angles of Joint 1 and Joint 2 of the arm;
- J_1 and J_2 are the moments of inertia of arm 1 and arm 2;
- R_1 is the radius of the cylindrical arm 1 (idealized as a thin cylindrical tube)

The values of the parameters m_1, m_2, l_1, l_2 are shown in Table 1

Table 1. Values of the physical quantities of the robotic arm [7]

No	Quantity	Unit	Value	N	Quantity	Unit	Value
1	m_1	kg	3	4	l_2	m	0.25
2	m_2	kg	12	5	R	m	0.03
3	l_1	m	0,05				

3.3.1.2. Dynamics of the robotic arm of the spray system

According to the Lagrange method, the dynamics equation is as follows:

- The total kinetic energy of the system.

$$\sum K = K_1 + K_2 \tag{5}$$

Kinetic energy of arm 1

$$K_1 = \frac{1}{2} J_1 \dot{\theta}_1^2 = \frac{1}{2} m_1 R^2 \dot{\theta}_1^2 \tag{6}$$

Kinetic energy of arm 2

$$K_2 = \frac{1}{2} m_2 l_2^2 (\dot{\theta}_2^2 \sin^2 \theta_2 + \dot{\theta}_1^2 \cos^2 \theta_2) + \frac{1}{6} m_2 l_2^2 \dot{\theta}_2^2 \tag{7}$$

Substituting K_1 and K_2 into (5), we obtain the total kinetic energy of the system:

$$\sum K = K_1 + K_2 = \frac{1}{2} m_1 R^2 \dot{\theta}_1^2 + \frac{1}{2} m_2 l_2^2 (\dot{\theta}_2^2 \sin^2 \theta_2 + \dot{\theta}_1^2 \cos^2 \theta_2) + \frac{1}{6} m_2 l_2^2 \dot{\theta}_2^2 \tag{8}$$

- The total potential energy of the system:

$$\sum P = P_1 + P_2 = m_1 g l_1 + m_2 g (2l_1 + l_2 \sin \theta_2) \tag{9}$$

- The Lagrangian equation of the system:

$$L = \sum K - \sum P$$

$$= \frac{1}{2} m_1 R^2 \dot{\theta}_1^2 + \frac{1}{2} m_2 l_2^2 (\dot{\theta}_2^2 \sin^2 \theta_2 + \dot{\theta}_1^2 \cos^2 \theta_2) + \frac{1}{6} m_2 l_2^2 \dot{\theta}_2^2 - m_1 g l_1 - m_2 g (2l_1 + l_2 \sin \theta_2) \quad (10)$$

- The equations of motion of the system will have the form:

$$\frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = M_i \quad (11) \quad \frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{\theta}_2} \right) - \frac{\partial L}{\partial \theta_2} = M_2 \quad (12)$$

Where q_i is the generalized coordinate of the system ($i=1,2$), M_i is the moment applied by the motor to joint 1 and joint 2

- From (10), (11), and (12), the following can be calculated:

◆ The equations of motion at joint 1:

$$(m_1 R^2 + m_2 l_2^2 \cos^2 \theta_2) \ddot{\theta}_1 - m_2 l_2^2 \dot{\theta}_1 \dot{\theta}_2 \sin 2\theta_2 = M_1 \quad (13)$$

◆ The equations of motion at joint 2:

$$m_2 l_2^2 (\sin^2 \theta_2 + \frac{1}{3}) \ddot{\theta}_2 + m_2 l_2^2 \dot{\theta}_2^2 \sin 2\theta_2 - \frac{1}{2} m_2 l_2^2 \sin 2\theta_2 (\dot{\theta}_2^2 - \dot{\theta}_1^2) + m_2 g l_2 \cos \theta_2 = M_2 \quad (14)$$

From (13) and (14), The system of equations of motion of the robotic arm is:

$$\begin{cases} (m_1 R^2 + m_2 l_2^2 \cos^2 \theta_2) \ddot{\theta}_1 - m_2 l_2^2 \dot{\theta}_1 \dot{\theta}_2 \sin 2\theta_2 = M_1 \\ m_2 l_2^2 (\sin^2 \theta_2 + \frac{1}{3}) \ddot{\theta}_2 + m_2 l_2^2 \dot{\theta}_2^2 \sin 2\theta_2 - \frac{1}{2} m_2 l_2^2 \sin 2\theta_2 (\dot{\theta}_2^2 - \dot{\theta}_1^2) + m_2 g l_2 \cos \theta_2 = M_2 \end{cases} \quad (15)$$

Convert the system of equations (15) into matrix form:

$$\begin{bmatrix} M_1 \\ M_2 \end{bmatrix} = \begin{bmatrix} d_{11} & 0 \\ 0 & d_{22} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} -m_2 l_2^2 \dot{\theta}_1 \dot{\theta}_2 \sin 2\theta_2 \\ +m_2 l_2^2 \dot{\theta}_2^2 \sin 2\theta_2 - \frac{1}{2} m_2 l_2^2 \sin 2\theta_2 (\dot{\theta}_2^2 - \dot{\theta}_1^2) \end{bmatrix} + \begin{bmatrix} 0 \\ m_2 g l_2 \cos \theta_2 \end{bmatrix} \quad (16)$$

Where,

$$d_{11} = (m_1 R^2 + m_2 l_2^2 \cos^2 \theta_2); d_{22} = m_2 l_2^2 (\sin^2 \theta_2 + \frac{1}{3})$$

The matrix is simplified to:

$$\bar{M} = D(\bar{\theta}) \ddot{\bar{\theta}} + C(\bar{\theta}, \dot{\bar{\theta}}) + G(\bar{\theta}) \quad (17)$$

From (17), The following comment can be made:

◆ $D(\bar{\theta})$ is the term characteristic of the joint's inertia; $C(\bar{\theta}, \dot{\bar{\theta}})$ is the term characteristic of the coupling between the two joints; $G(\bar{\theta})$ is the term characteristic of the gravitational force acting on the two joints; \bar{M} is the torque generated by the motor applied to the two joints.

◆ Equations (16) and (17) are the dynamics equations of the robotic arm, which form the basis for designing the motion control of the robotic arm.

3.3.2. Design of Robotic arm control System

3.3.2.1. Determining the Rotation angle of the spray system

After determining the distance from the camera position to the fire (L) (according to formula (4)), the rotation angle of the spray system (θ) is determined.

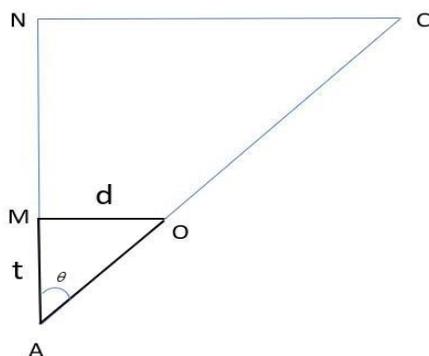


Figure 7. Rotation angle θ of the spray system.

Where,

A is the position of the camera

O is the position of the flame in the image

C is the position of the actual flame

$AM = t = 3.6 \text{ mm}$ is the camera's focal length

$OM = d$ is the distance from the camera center to the center of the flame in the image

From figure 7, Rotation angle θ is determined:

$$\theta = \arccos\left(\frac{t}{d}\right) \quad (18)$$

3.3.2.2. The Problem of the robotic arm of spray system Control

Based on the dynamic equation (18), the control problem of moving the spray system from the initial position (0,0) to a specific angle (θ_1, θ_2) after a certain period of time will be controlled. This value is obtained from the image analysis problem.

3.3.2.3. Design of Fire Fighting the Robotic Arm Control

- PD Control Law

The n-DOF robotic arm has the general dynamic equation:

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = M \quad (19)$$

Where, $D(q)$ is a positive definite, symmetric square matrix ($n \times n$) representing the inertia matrix; $C(q, \dot{q})$ is a vector ($n \times 1$) representing the centripetal and viscous force components.

When neglecting the effects of the gravitational component and external disturbances, using a PD controller with the control law as in equation (19), it is possible to satisfy the stability requirements of the closed-loop control system when the trajectory input is a fixed, predefined point.

$$M = K_d \dot{\epsilon} + K_p \epsilon \quad (20)$$

With K_d, K_p are positive definite diagonal matrices, and $\epsilon = q_d - q$ is the deviation.

The simulation result of the PD control law for the spray system is as follows:

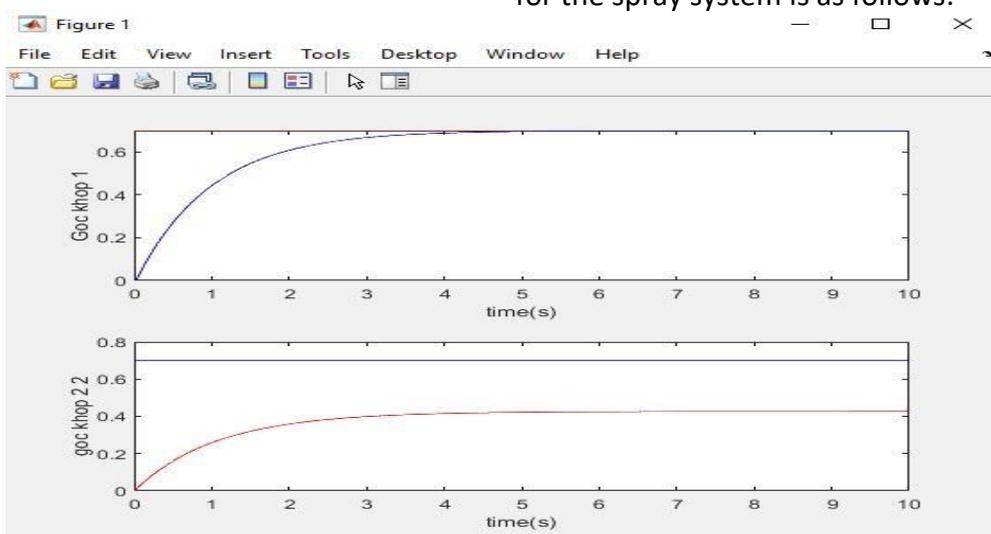


Figure 8. Simulation result of the PD control law for the spray system

Clearly, there is a certain position deviation in the control results due to the gravitational force, which causes the position to be offset.

- PD + G Control Law

When the robot is subjected to gravitational forces, using the PD law with gravitational compensation (PD + G) can satisfy the requirement of controlling the joint trajectory to a predefined target point:

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = M \quad (21)$$

Where $G(q)$ is the gravity matrix (representing the effect of gravitational forces on the robot).

So, the Control law PD+G:

$$M = K_d \dot{\epsilon} + K_p \epsilon + \hat{G}(q) \quad (22)$$

Where $\hat{G}(q)$ is the gravitational compensation component. The simulation result of the PD+G control law for the spray system as follows:

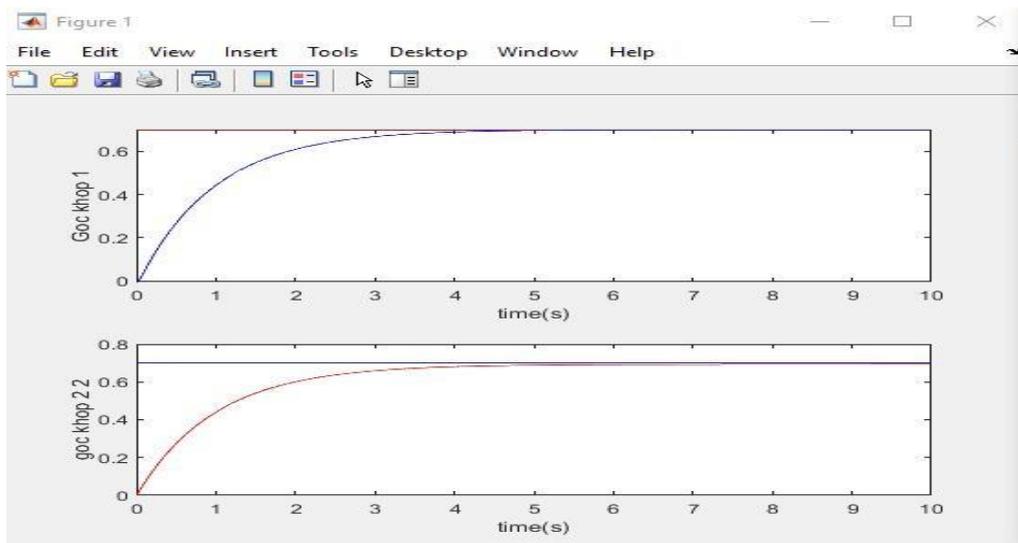


Figure 9. Simulation result of the PD+G control law for the spray system

3.3.2.4. Simulation Diagram of the Robotic Arm control System

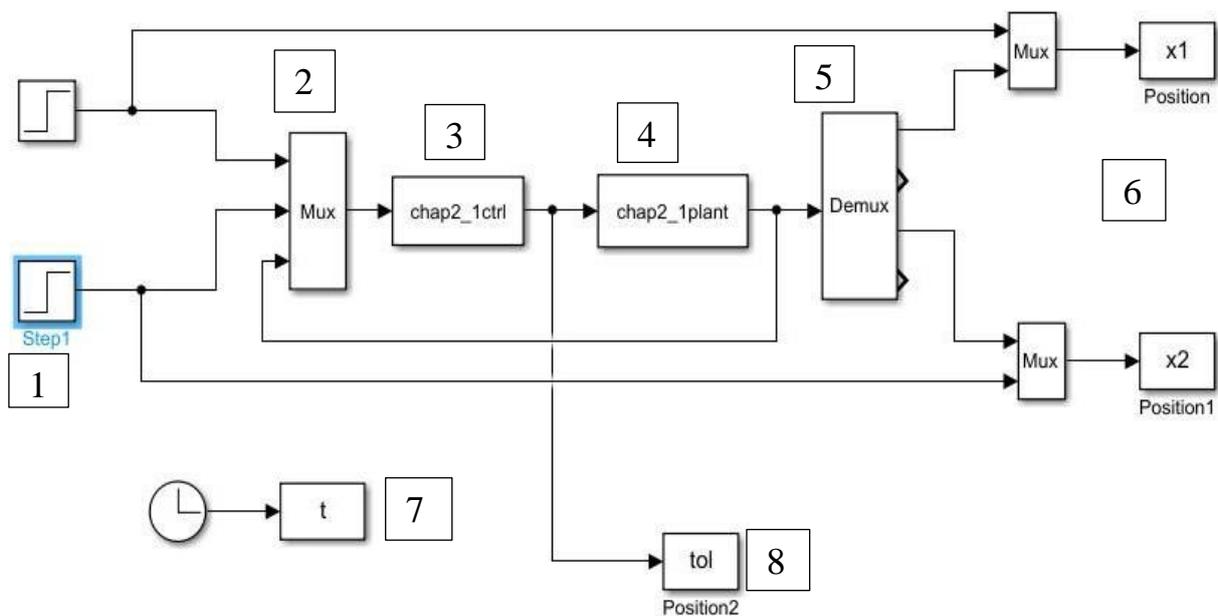


Figure 10. Diagram of the simulation of the robotic arm control of spray system.

Where,

Block 1: Step Input block - This is an input signal block that can be used to test the system's response to a sudden input signal.

Block 2: Mux block - The Multiplexer block is responsible for combining multiple input signals into a single signal to be transmitted to the control system.

Block 3: chap2_1ctrl control block – This can be a controller block responsible for adjusting the input control signal based on the system's feedback.

Block 4: chap2_1plant - This block simulates the system's dynamics that need to be controlled.

Block 5: Demux output signal separation block - This block is responsible for separating the output signal from the system into multiple distinct signals.

Block 6: x1, x2 - These blocks represent the system's position according to the coordinate system (after control).

Block 7: t (Clock) - The time clock, which can be used to track the simulation time.

Block 8: tol (Position2) - This block represents the system's position according to the coordinate system (before control).

The operation of the control system is as follows: (i) Initialization of input signals: **The Step1 block** provides a step signal, which serves as the system's input. When the simulation starts, the value of this signal increases from 0 to a specified value at a specific point in time; this input signal is then combined using Mux and sent to the controller. (ii) Control processing: **The chap2_1ctrl block** (controller) uses the PD+G algorithm to receive the input signal from Mux, then calculates and provides the appropriate control signal to adjust the system. (iii) System modeling (Plant): **The chap2_1plant block** (the model of the fire fighting system) receives the control signal from the controller and simulates the real system's response according to the dynamic equations. (iv) Output processing: After passing through the fire fighting system block, the output signal is separated into various components by Demux. These output signals may include position, speed, or other states of the system. (v) Displaying results: The two main output signals (x1 and x2) are combined using Mux blocks and displayed through the "Position" and "Position1" blocks. The tol block (Position2) is another status signal to check the

system's deviation or tolerance; the time clock t can be used to track changes in the system over time. (vi) Feedback loop: If the system has a feedback mechanism, the output signal will be sent back to the controller to adjust the control signal and ensure that the system operates stably.

4. CONCLUSION

The use of robots in fire prevention is an inevitable trend that enhances the effectiveness of this work while minimizing risks to firefighting personnel. With the rapid development of fire prevention technology, artificial intelligence is increasingly being integrated into fire prevention robots. A woodworking factory, which contains many flammable materials and has limited space, is an ideal setting for the use of robots in fire prevention. The research titled "*Using Artificial Intelligence (AI) to Detect Fires in Woodworking Factories and Designing the Robotic Arm Control for the Spray System in a Mobile Firefighting Robot*," which applies artificial intelligence, has significantly improved fire detection in woodworking factories. It enables quick identification of fires, accurate determination of their locations, and

calculation of the distance from the fire's center to the firefighting robot. Furthermore, the research employs automatic control algorithms to design a control system for the robotic arm with a spraying nozzle, allowing the robot to operate the spray system efficiently and precisely. However, due to the varying scales and types of products in woodworking factories, the research results need to be practically tested for necessary adjustments.

REFERENCES

- [1]. Fire Prevention and Rescue Police Department (2023). Press release on fire prevention and fighting and search and rescue (CNCH) nationwide in 2023 (Vietnamese).
- [2]. Nguyen Thuc Anh, Hoang Son, Nguyen Xuan Nguyen, Nguyen Huy Manh, Nguyen Thanh & Dan Duc (2022). Fire Detection for Automated Firefighting Robot

by using Efficient Det. Impact Factor 3.582 Case Studies Journal ISSN (2305-509X). 11(8): 9-18.

[3]. Nguyen Phan Thiet, Nguyen Van Dien, Nguyen Trong Kien & Vu Manh Tuong (2017). Report on the capabilities of Vietnamese wood furniture enterprises, JICA (Vietnamese).

[4]. Nguyen Dinh Tung, Do Chi Dung & Nguyen Khac Thong (2017). Study on the combustion process of wood biomass and biogas in solid fuel boilers. Journal of Rural Industry, Vietnam Association of Agricultural Mechanics. 25: 63-71 (Vietnamese).

[5]. Le Tien Dung (2019). Research on the design of a synchronous adaptive controller for a planar robotic arm. Ministry of Education and Training (Vietnamese)

[6]. Hoang Son (2002). Textbook Industrial Robots. Publishing House Science and Technique, Ho Chi Minh City (Vietnamese).

[7]. Nguyen Khanh Nam (2024). Design of a Firefighting Robot for a Wood Workshop. Graduation Thesis, Hanoi University of Science and Technology (Vietnamese).