RESEARCH ARTICLE OPEN ACCESS

Manuscript received February 14, 2024; revised February 21, 2024; accepted March 17, 2024; date of publication March 19, 2024
Digital Object Identifier (**DOI**): https://doi.org/10.35882/teknokes.v17i1.679

How to cite: Biplov Paneru, Bishwash Paneru, Ramhari Poudyal, Krishna Bikram Shah, Khem Narayan Poudyal, and Yam Krishna Poudel, "Automated Environmental Stewardship: A Ribbon-Cutting Robot with Machine Vision for Sustainable Operation", Jurnal Teknokes, vol. 17, no. 1, pp. 8–19, March. 2024.

Automated Environmental Stewardship: A Ribbon-Cutting Robot with Machine Vision for Sustainable Operation

Biplov Paneru[1], Bishwash Paneru[2], Ramhari Poudyal[3], Krishna Bikram Shah[4], Khem Narayan Poudyal[2], and Yam Krishna Poudel[1]

[1] Department of Electronics & Communication Engineering, Nepal Engineering College Pokhara University, Bhaktapur, NEPAL
[2] Department of Applied Science Engineering, Tribhuvan University, Kathmandu, NEPAL
[3] Department of Electrical Engineering, Purbanchal University, Biratnagar, NEPAL
[4] Department of Computer Science and Engineering, Nepal Engineering College Pokhara University, Bhaktapur, NEPAL
Corresponding author: Biplov Paneru (e-mail: biplovp019402@nec.edu.np)

ABSTRACT This paper provides a novel way for automating ribbon-cutting rituals that use a specifically constructed robot with superior computer vision capabilities. The system achieves an outstanding 92% accuracy rate when assessing picture data by using a servo motor for ribbon identification, a motor driver for robot movement control, and nichrome wire for precision cutting. The robot's ability to recognize and interact with the ribbon is greatly improved when it uses a Keras and TensorFlow-based red ribbon identification model which obtained accuracy of about 93% on testing set before deployment in system. Implemented within a Raspberry Pi robot, the method exhibits amazing success in automating ceremonial activities, removing the need for human intervention. This multidisciplinary method assures the precision and speed of ribbon-cutting events, representing a significant step forward in the merging of tradition and technology via the seamless integration of robots and computer vision.

INDEX TERMS Robot, Color Detection, Tensorflow, Keras

I. INTRODUCTION

There are always ribbon-cutting rituals for a variety of occasions, from product launches to inaugurations. These occasions can be made more effective and memorable by streamlining the procedure and automating it with a ribbon-cutting robot [24]. The goal of this project is to create a flexible robot that can recognize and cut red ribbons on its own, revolutionizing customary ribbon-cutting events. Systems based on robotics and artificial intelligence are crucial for automating the world and paving the way for further advancement. Even though they have a lot of drawbacks, developing intelligent systems for sustainability requires the intelligent application of robots and AI-based methodologies [5]. Ribbon-cutting ceremonies are ceremonial occasions to usher in new buildings, facilities, or endeavors. Traditionally, this ceremony entails a human participant cutting a ribbon with scissors to mark the project's official start. The development of a computer vision-based ribbon-cutting robot was spurred by the increasing interest in automating ceremonial tasks in an era driven by technological advancements. Robotics can create a system that can visually perceive and interact with its environment by integrating computer vision with robotics. In this project, we investigate the real-time identification and location of the ribbon through the use of computer vision algorithms. A robotic control system then processes this data, directing the robot's manipulator to carry out the exact cutting motion.

Traditional ribbon-cutting ceremonies, despite their symbolic significance, are not without limitations (FIGURE 1). Human-led execution introduces the potential for errors, inconsistency in cuts, and safety concerns. Moreover, accessibility challenges may exclude certain individuals from participation, while logistical complexities and time constraints can hinder efficiency. These limitations underscore the need for innovation in ceremony execution. By employing autonomous ribbon-cutting robots equipped with machine vision, these challenges can be mitigated. Such robots ensure precise, consistent cuts, enhance safety measures, improve accessibility, and streamline the ceremony process. In doing so, they offer a more efficient and memorable experience for participants and spectators alike [24].

II. LITERATURE REVIEW

The robot and mobile application can communicate wirelessly thanks to the Bluetooth module. IC L293D is utilized to obtain the Stabilized output. The bot arm structure was constructed from polymer material, which reduces the "total weight" without changing any parameters. Instead of using the conventional setup of four DC motors, just two are used in the prototype to reduce costs without sacrificing efficiency.

FIGURE 1. Illustration of robotic technology in ribbon cutting for inauguration

As a result, the developed system is more adaptable, and the prototype can help surgeons by selecting and positioning surgical instruments to reduce the possibility of human contact and contamination [1]. The complete system depends on building a prototype that is fixed to a four-wheeled robotic car chassis with an Arduino controller. Within Ribbons are cut in this instance with a nickel wire, which can be paired through an app or Bluetooth with an Android device. The automaton is among the initial stages of creating an independent, cognitive robots and are applicable to a range of occasions [2]. Both the developed control algorithm and the entire estimation process are independent of time. The independent variable is a quantity related to wheel rotations that is naturally suitable. This decision has the effect of allowing the vehicle speed to be specified without reference to the estimation and control algorithms. Reference paths are "taught" by guiding the car along the intended path by hand. The geometry of the path is then represented by estimates obtained during this training session using the extended Kalman filter. Controlling the vehicle's position and orientation in relation to the reference path allows for the tracking of taught reference paths. The requirement for time-independence path tracking has led to the creation of a new, geometry-based method for moving along the reference path [3]. Other supplemental navigation systems, like computer-vision-based positioning, are used for more precise localization and mapping in the event that the GPS signal becomes unstable. However, harsh environmental conditions may also degrade the quality of data obtained from AV's sensors, which inevitably results in a decline in navigation performance. The authors took into account the Arctic region use case, which presents additional difficulties for the AV's navigation and may use artificial visual landmarks to improve the localization quality, which we used for the computer vision training, in order to validate our computer-vision-based localization system design. Affine transformations were used to further improve our data and make it more diverse. For their system design, they went with the YOLOv4 image detection architecture since our experiments showed it to perform the best [4].

The authors are demonstrating in this paper how the Raspberry Pi is used to implement the small-scale robotic vehicle model that is automated and how data can be transferred to the cloud. in order for someone to be able to monitor and maintain the vehicle's movement, speed, and distance traveled on city roads. This model is based on the safety precautions that need to be implemented for these unmanned vehicles, which will soon be a thing of the past. The Raspberry Pi development board serves as the main processor, the camera module identifies traffic signals, and the ultrasonic sensor detects any vehicles in the vicinity. If a vehicle is detected, its direction of travel changes, and the remaining data about the data uploaded to the cloud is comprehensively explained in the paper [5].

A novel framework for managing mobile robotics is explained. The robot can operate at progressively higher levels of proficiency thanks to layers of control system architecture. Asynchronous modules that communicate over low-bandwidth channels make up layers. Every module is an example of a reasonably basic computer system. By suppressing their outputs, higher-level layers can take on the responsibilities of lower-level layers. Higher levels are added, though, and lower levels stay operational. A reliable and adaptable robot control system is the end result. A mobile robot that roams freely through computer rooms and unrestricted lab spaces has been operated by the system [6].

A set of equations that transparently define the phase space of admissible motion limited by path geometry and joint torques are produced by transforming the equations of motion to this one DOF. A field of extremals bound by a maximum velocity curve, which functions as a trajectory source or sink, makes it simple to identify the time optimal solution representing the maximum mobility of the path-manipulator configuration. Due to its characteristics, an algorithm can be developed to determine the time-minimum curve given a series of accelerating and decelerating extremals. Bellman's principle is used to study additional optimizing criteria [7]. The author’s major objective was to acquire a single low-cost, multifunctional robot that could be utilized for research, teaching, and public outreach initiatives. According to the authors this robot would be more than ten times less expensive than platforms that are currently on the market. A detailed description is given of the ExaBot's body, sensors, actuators, processing units, and control board. For the benefit of the robotics community, the printed circuit board and software created for this project are open source, enabling upgrades to the current version. Lastly, various ExaBot configurations are displayed, showcasing a number of applications that meet the specifications for which this robotic platform was created [8]. The primary objective of the authors was to create a low-cost mobile robot that could be used for research and education in addition to other purposes. This robot was to be cheaper than the majority of commercial robots on the market today. In order to create this mobile robot, a chassis with a Raspberry Pi controller that was remotely controlled over a Wi-Fi network using a Debian Linux-based operating system and a C programming was designed and implemented. An Android smartphone was equipped with a mobile application that featured an intuitive graphical user interface (GUI). Video footage is wirelessly transferred to Android smartphones via Wi-Fi and stored in the device's local memory [9].

These days, artificial intelligence is being used to create increasingly sophisticated robots. One such robot is the navigation robot, which is controlled by its neural network. As a result, a range of sensors must be dispersed throughout the robot's neural network in order for it to receive commands for navigation. During the process, make plans for the route, take in outside information, and adjust our position. As a result, the Internet of Things-based multi-sensor information fusion technology is crucial. With the help of IOT information fusion technology, it can enhance the robot navigation system and provide a better information base for the robot. It can also increase accuracy [10].

Many researches have been carried out with Bluetooth based interfacing. A high need of a machine vision based low-cost special purpose robot is there [16] and many researches are ongoing for developing machine vision powered intelligent systems. This robot prototype we have developed utilizes a computer vision and OpenCV based model for ribbon detection and cutting ribbon implementing CNN+Keras algorithm. Our robot prototype is unique in the field of wireless communication since it makes use of path detection system and fulfills a research gap in color detection specific purpose robotics development. This technology facilitates easy communication over short distances and is well-known for its dependability and broad usage. Although other options such as Wi-Fi or Zigbee may provide longer ranges or faster data transfer rates, OpenCV based tracking red ribbon and directional movement of robot in the tracked trajectory makes it especially suitable for the short-range interactions needed in our robot prototype. Our design's selection of range detection indicates a sensible and approachable method for wireless control and data exchange (FIGURE 2).

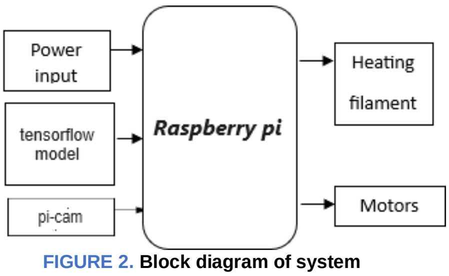

FIGURE 2. **Block diagram of system**

This robot prototype's computer vision part uses the OpenCV library, which is a strong and flexible option. Acknowledged for its extensive collection of image processing and computer vision tools and algorithms, OpenCV is an industry-standard open-source library. Although there are other frameworks with more sophisticated deep learning capabilities, like TensorFlow and PyTorch, OpenCV is a better option because of its ease of use and versatility, especially for real-time applications. The incorporation of OpenCV indicates an emphasis on useful application and effective visual data processing. Our robot prototype advances object recognition technology by incorporating a Tensorlow keras based algorithm for ribbon detection. They are now considered state-of-the-art for tasks involving image recognition because of their high accuracy in identifying and categorizing objects in images. By using a deep leraning model to detect ribbons, we have taken a sophisticated approach to visual analysis, enabling the robot to pick up on and recognize complex ribbon features.

Although other architectures such as YOLO and Faster R-CNN are also good at object detection, we have selected a CNN based on the rigorous needs of our particular application. Our robot prototype's integration of a deep learning tensorflow for object detection, OpenCV-based computer vision, and Bluetooth-based control positions it as a sophisticated and well-rounded robotic system. The smooth integration of these technologies enables the robot to process visual information intelligently for accurate ribbon detection and cutting, in addition to receiving commands and sending data wirelessly. The potential uses for our robot prototype are numerous and demonstrate the flexibility and adaptability of the device. They include packaging and automated gift wrapping among other industries.
Prospective directions for advancement and growth encompass honing the tensorflow and keras model to achieve even greater precision in ribbon identification, streamlining the robot's cutting mechanism for maximum effectiveness, and investigating supplementary features. Innovation in tasks requiring a combination of wireless communication and advanced computer vision capabilities may be made possible by ongoing research and development in the field of robotic automation.

III. METHODOLOGY

The main aim is to implement a tensorflow and keras model and use a four-wheeler robotic car frame to make a body with the help of wooden sticks to support the nichrome wire used for the ribbon cutting. The ribbon tracking algorithm is then put into practice by utilizing the OpenCV library's capabilities. This algorithm finds and tracks the red ribbon by analyzing every frame that has been taken. The particular specifications and characteristics of the ribbon-cutting robot would determine the specifics of the tracking algorithm. This perpetual monitoring loop guarantees instantaneous adjustment to variations in the ribbon's location. Implementation includes safety precautions as a fundamental component. An emergency stop button or an obstacle detection system can be incorporated into the system to guarantee safe operation. By preventing accidental cutting and preparing for unforeseen events during a ribbon-cutting event, this provides an extra layer of safety. At this point, testing and optimization become crucial. To verify the precision of the ribbon tracking algorithm and the dependability of the cutting mechanism, extensive testing is carried out. Testing results are used to fine-tune parameters, ensuring optimal performance in a variety of scenarios. Since the Raspberry Pi is a component of our multifunctional system, integration is the next step. The robot's overall control system incorporates both the ribbon tracking and cutting logic. Together, they guarantee a smooth operation, enabling the robot to recognize and cut the ribbon on its own at ceremonial occasions.

A. SYSTEM ALGORITHM AND FLOWCHART

The purpose of this algorithm is to direct the creation of a ribbon-cutting robot based on a Raspberry Pi. The robot tracks the distance of the ribbon with a camera and cuts it when it reaches the cutting range.

1. Setup and Calibration:
   a) Connect the Raspberry Pi camera module to the Raspberry Pi.
   b) Adjust camera settings for optimal performance (focus, field of view).
2. Image Capture and Processing:
   a) Capture continuous images or video frames using the camera.
   b) Implement image processing to detect and track the ribbon in the frames.
3. Distance Measurement:
   a) Estimate the distance between the camera and the ribbon using image features or depth information.
4. Control Logic:
   a) Implement logic to determine when the ribbon is within cutting range.
   b) Set a threshold distance for triggering the cutting mechanism.
5. Relay Control:
   a) Connect the relay module to the Raspberry Pi GPIO pins.
   b) Write code to control the relay, activating the cutting mechanism.
6. Cutting Mechanism:
   a) Design a cutting mechanism with a nichrome wire.
   b) Ensure the mechanism is securely mounted and aligned with the ribbon's position.
7. Safety Measures:
   a) Implement an emergency stop button or obstacle detection to ensure safe operation.
   b) Consider adding features for unintended cutting prevention.
8. Testing and Calibration:
   a) Test the entire system thoroughly.
   b) Calibrate distance measurements and fine-tune control parameters for optimal performance.

Different methods for measuring distance were performed concurrently. Estimating the distance between the ribbon and the camera was vital to improve the accuracy of the ribbon-cutting procedure, depending on the setup. This may entail calculating the ribbon's distance from the camera using its known dimensions (Eq. (1)).

$$\text{Distance}=Pixel\ Width * \frac{Known\ Width}{Focal\ Length} \tag{1}$$

Several critical equations play an important role in robotic motion calibration, with Equation 1 standing out as the most basic. This equation serves as the foundation for determining the distance between the camera and the item of interest, in this case a ribbon [22]. It is based on four key parameters: distance, known width, focal length, and pixel width. Distance reflects the actual distance between the camera and the ribbon, whereas Known Width refers to the ribbon's predefined width. Focal Length captures the camera's optical characteristic and indicates its focal length. Pixel Width is the measured width of the ribbon in pixels within the recorded picture. Distance Traveled by Each Wheel (d) as shown in Eq. (2):

$$d = 2\pi r\theta/360 \tag{2}$$

where the letters 'd', 'r', and 'θ' represent the distance traveled by each wheel, the radius of the wheel, and the intended angle of rotation [21] . By combining these variables, this equation allows you to calculate the amount of wheel movement required to produce a specific angular rotation. This calibration procedure assures the precise execution of robotic operations, which is critical for activities ranging from navigation to manipulation. With these equations at the forefront of robotic calibration approaches, engineers may fine-tune and improve robotic systems for a wide range of applications, increasing efficiency and reliability in robotic operations. Distance Traveled by the Robot Center (D) is calculated by Eq. (3):

$$D = dr + \frac{dl}{2} \tag{3}$$

In this equation, 'D' represents the distance traveled by the robot center, whilst 'dr' and 'dl' represent the distances covered by the right and left wheels. By combining these variables, this equation makes it easier to synchronize wheel motions in order to accomplish the necessary center displacement of the robot. This calibration procedure is critical for maintaining precise navigation and placement of the robot, which is required for activities ranging from path planning to obstacle avoidance. With these equations as the foundation of robotic motion control, engineers may enhance and optimize the performance of robotic systems across a wide range of applications, promoting agility and precision in robotic operations. Angle of Rotation for the Robot Center (α) is calculated by using Eq. (4):

$$\alpha = dr - \frac{dl}{w} \tag{4}$$

Here, 'α' denotes the angle of rotation for the robot center, while 'W' represents the wheelbase, marking the distance between two wheels. By combining these characteristics, this equation allows for the accurate determination of the angle of rotation necessary to perform specified robot movements or orientations [21]. This calibration procedure is vital to providing precise rotational motions, which are required for activities like as navigation, alignment, and object manipulation. Engineers may use this equation as the foundation of robotic motion control to improve the performance and adaptability of robotic systems, allowing them to function efficiently in a variety of situations and jobs.

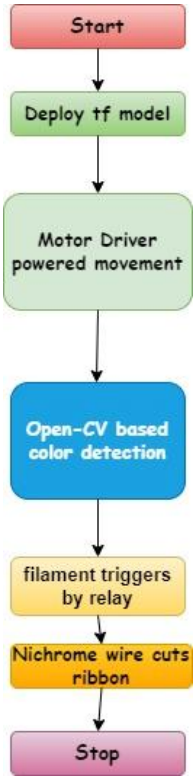

FIGURE 3. **Proposed method for system development**

Using these equations, we can calculate the required wheel movements (distance or rotation) to reach a specific position or orientation. This is a representation and we assume a straight-line motion for development of this prototype. In a more complex scenario, involving navigation and obstacle avoidance, additional factors such as wheel slippage, turning radius, and the robot's initial pose would need to be considered. As a result, the robot can use the angular rotation calculation equation if it cannot detect the ribbon with a straight line of sight. It can also move in the direction of the ribbon when it is free to do so, and when it does, the filament heats up as a relay triggers. Adding GPIO (General Purpose Input/Output) pins for relay control is the next step. The cutting mechanism is connected to the relay, which is activated by control logic that is put in place to set off the relay under certain circumstances. These circumstances might be finding the ribbon or traveling a certain distance before starting the cutting operation. Easy control of the relay, an essential part of the ribbon-cutting mechanism, is made possible by the GPIO library. The li-po high power battery operating at 12v is chosen due to its highly stable supply, rechargable feature and its ability to withstand the filament heating requirements.

As can be seen in FIGURE 3, we employ an OpenCV based color detection model based on CNN, and after the ribbon is identified, it may be cut using a Nichrome wire-based frame. When the nichrome recognizes the ribbon, it heats up and automatically cuts it, enabling a robotic way to be used for the event's inauguration

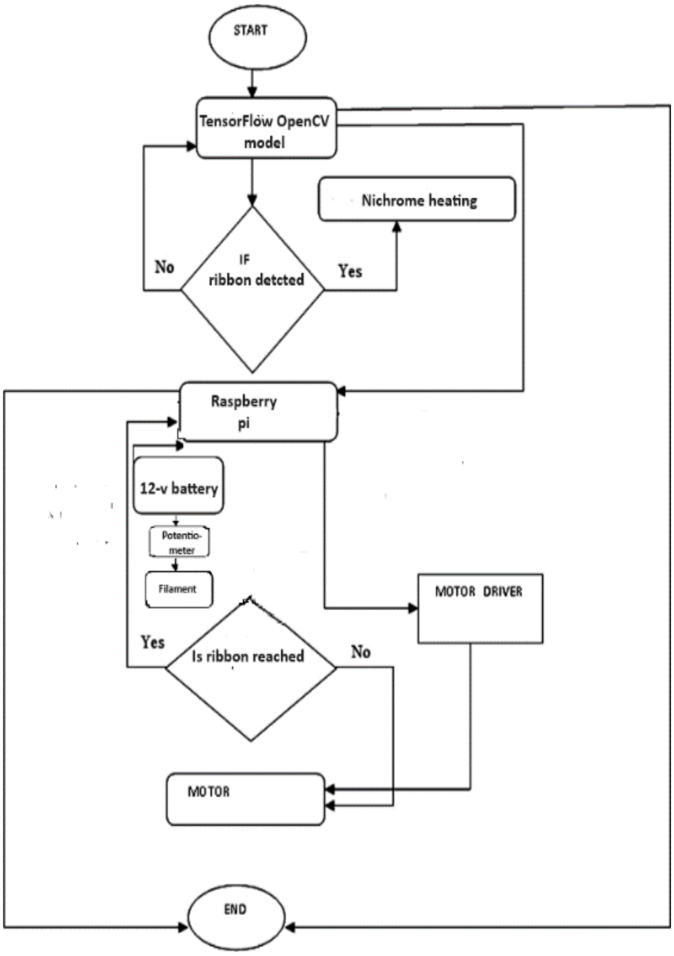

FIGURE 4. **Flowchart of system**

A. ROBOT DESIGN AND HARDWARE CONFIGURATION

The robot is constructed using a robust frame with a motor driver to control its movements. A servo motor is employed for precise ribbon detection, and a nichrome wire is attached to facilitate the cutting process.

1. Computer Vision System:
   A CNN-based model is trained to recognize red ribbon patterns. This model is integrated with the robot's system to enable real-time ribbon detection.
2. Control Logic:
   The motor driver and servo motor are controlled by a microcontroller unit (MCU), programmed with the

necessary logic to navigate the robot and position the nichrome wire accurately.

3. Red Ribbon Detection Algorithm:
   The CNN model employs convolutional neural networks (CNNs) to process images and identify red ribbon patterns. The model is trained on a dataset of annotated ribbon images to ensure accurate detection.

As shown in FIGURE 4, we cut it by using a filament that lies in the wooden frame and touches the ribbon with excess heating. In our study endeavor, the machine vision approach based on OpenCV is the most crucial. For the aim of cutting the ribbon operating at 12v and introducing color detection, OpenCV-based detection is essential. This satisfies the need to automate the robot using computer vision techniques so that it can eventually be used for numerous artificial intelligence applications, including color identification.

B. DATASET COLLECTION AND PREPROCESSING

The user-developed script does not immediately save image datasets. As an alternative, it creates a dataset with RGB values for colors and labels for each color. Using matplotlib, these RGB values are used to produce colored rectangles that are shown on a plot. The dataset is kept in a "data.csv" CSV (Comma Separated Values) file. This CSV file contain rows that each represent a single color sample. The rows have columns that contain the color's label as well as its red, green, and blue values. Therefore, the code stores the numerical representation of colors along with their labels instead of the actual images (FIGURE 5). This can be used for tasks like color classification or any other analysis that requires labeled color data.

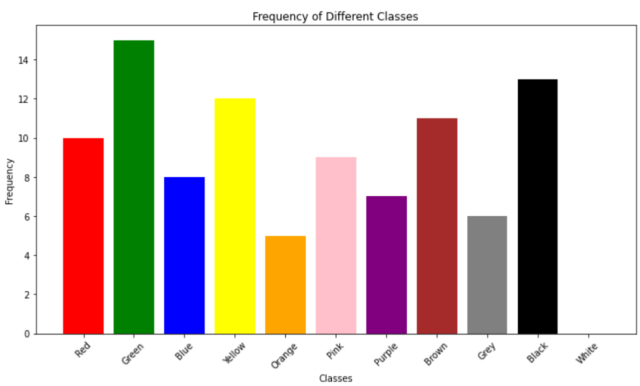

FIGURE 5. Dataset frequency wise division and classes

5014 rows and 14 columns were used in the CSV file dataset.

1. DATASET SPLITTING:

A random sampling technique is used to divide the dataset into training and testing sets. Tests use 20% of the data, while training uses 80% of it. This division guarantees that the effectiveness of the model can be assessed using unobserved data.

2. FEATURE EXTRACTION

The dataset includes labels encoded as binary indicators for different colors (label_Red, label_Green, label_Blue, etc.) and features (red, green, and blue). The model, the number of epochs, the learning rate, the training dataloader (train_loader), and the validation dataloader are all inputs to the fit function, which represents the training loop. The optimizer is initially set up inside the loop with the parameters and the designated optimization function (opt_func). Batch iterations of the training dataset are performed using the model in training mode (model.train()) for every epoch. Every batch has its training step calculated, and the gradients are backpropagated throughout the network. The gradients are then zeroed for the following iteration, and the optimizer is used to update the model's parameters.

C. MODEL ARCHITECTURE

The Keras Sequential API is used to create a neural network model. Multiple dense (fully connected) layers with ReLU activation functions make up the model [23]. Based on requirements for model complexity and experimentation, the number of neurons and layers is selected. To stop overfitting, regularization (L2 regularization with a coefficient of 0.001) is used.

1. MODEL COMPILATION

The model is compiled using a loss function (categorical cross-entropy) appropriate for multi-class classification tasks and an optimizer (Adam optimizer with a learning rate of 0.001). The assessment criterion that is selected is accuracy. Using the compiled configuration, the model is trained on the training dataset. With a batch size of 2048, the training is carried out for a predetermined number of epochs (5001 epochs in this case). Training progress is tracked, and after each epoch, the model's performance is assessed using validation data (20% of the training data).

2. MODEL EVALUATION

Using both the training and validation datasets, the model's performance metrics—accuracy and loss—are tracked throughout training. This makes it possible to evaluate the model's capacity for generalization and identify overfitting.

3. VISUALIZATION

The training progress, including accuracy and loss over epochs, is visualized using Matplotlib and TensorFlow Documentation (tfdocs). This makes it easier to comprehend how the model performs differently during training and whether any modifications are required.

C. DEEP LEARNING ALGORITHM AND TECHNIQUES

Neural Network (Deep Learning): The model architecture is a deep neural network because it consists of several dense layers. Deep learning is favored because it can efficiently extract intricate patterns from data, especially for image classification tasks (which this one appears to be, considering the RGB features). Regularization (L2 Regularization) [23]: By penalizing large weights in the model, L2 regularization is used to stop overfitting. This lowers the possibility that the model will memorize noise in the training set and enhances the model's capacity for generalization. The adaptive learning rate optimization algorithm known as Adam Optimizer is frequently employed in neural network training. Based on the magnitudes of the gradients, it modifies the learning rate during training, which can result in quicker convergence and improved performance.

D. HARDWARE AND SOFWTARES USED

Various hardware are needed like raspberry pi and pi camera for machine vision development for system and relay for triggering, Li-po battery 12v was used for power supply. The potentiometer could vary the voltage per heat requirement discussed in results section (TABLE 1).

TABLE.1
Hardware components for system development

Hardware device	Functions
Raspberry pi	Main controller board
Motors	Controlling robotic car
Relay	Actuating filament
Pi-camera	Detecting color
Motor Driver	Robot precise control
Potentiometer	Voltage control
Li-Po battery 12v	Power supply

Thonny IDE was utilized for programming the system which is inbuilt within raspberry pi based system. The model was deployed and tested then OpenCV program was tested and made to run on boot i.e. when system of raspberry pi opens then program auto opens and all system hardware starts functioning including programmed script (TABLE 2).

TABLE 2
Software components for system function

Software	Function
Thonny IDE	System programming

IV. RESULTS

We could successfully cut the ribbon with our robot after promising results were obtained after training our trained model that shown accuracy approximately 93%. The model was added to the raspberry pi as a compatible for such systems processing and then, tested in the system with the pi-camera based ribbon detection and testing rapidly (FIGURE 6).

A. Model Accuracy

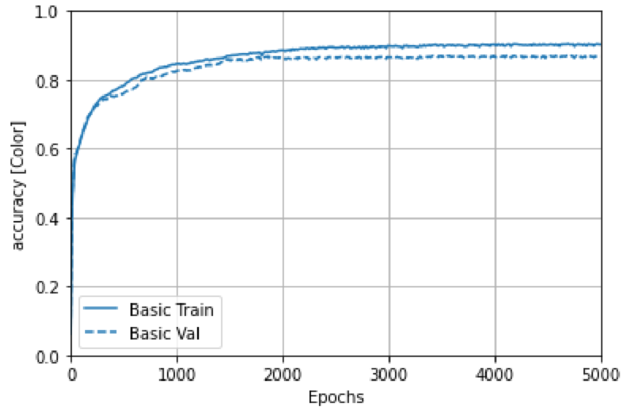

FIGURE 6. Model accuracy over different epochs

The model obtained 0.94 accuracy on training and on valida tion 0.92 it obtained. An accuracy of 0.99 represents a note worthy achievement with important ramifications for sustai nable waste management practices in the context of our rese arch on waste classification and recycling recommendation using an advanced machine learning model. The model's ex ceptional ability to accurately categorize and predict waste t ypes has been demonstrated. It was carefully designed and t rained on a diverse dataset representing various waste item. This high accuracy indicates that our model is capable of ha ndling the complexity of waste classification, which is an im portant part of recycling process optimization.

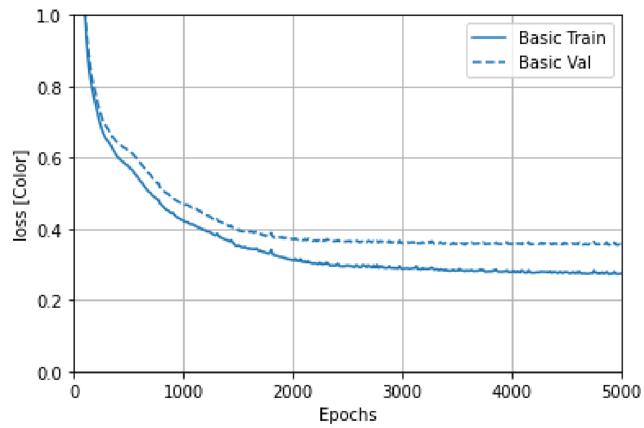

FIGURE 7. Training vs validation loss

A good result was seen during model training and evaluation of performance was successively carried out with better outcomes seen through the model on testing data too (FIGURE 7).

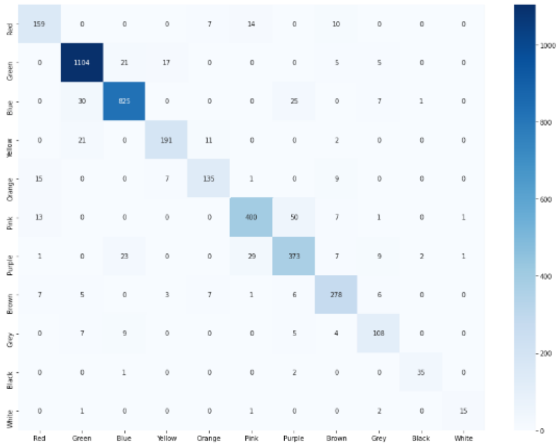

FIGURE 8. Predicted label

As, we can see in FIGURE 8, the confusion matrix shown a great result at predicting various type of colors through our specific motive was to create a red ribbon color detection model

B. MACHINE VISION TEST

Extensive testing revealed promising results from the computer vision model trained for red ribbon detection, using a dataset compiled from images gathered on the internet. The aim of this project was to create a reliable and precise model that could detect and monitor red ribbons in various real-life situations (FIGURE 9).

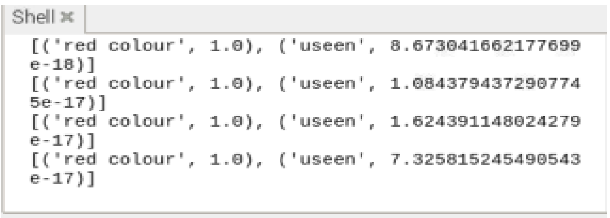

FIGURE 9. Color Detection test in raspberry pi system

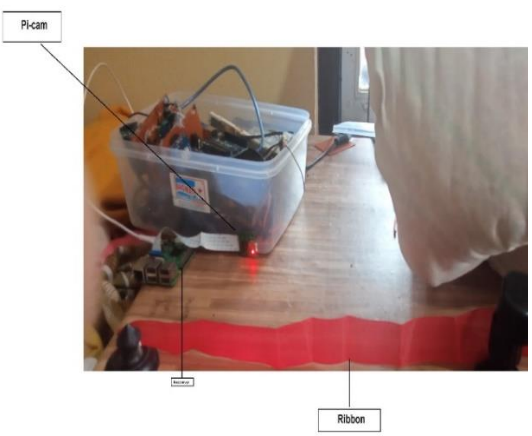

FIGURE 10. Ribbon Detection test

To improve the model's adaptability, a variety of backgrounds, lighting conditions, and ribbon shapes were included in the training dataset, which was assembled from multiple online sources (FIGURE 10). The final model demonstrated a high level of accuracy in red ribbon detection, demonstrating its usefulness in a variety of settings. The model's promising practical applications are highlighted by its strong generalization to previously unseen images. Our TensorFlow model that is compatible to raspberry pi system was trained with accuracy of 93%.

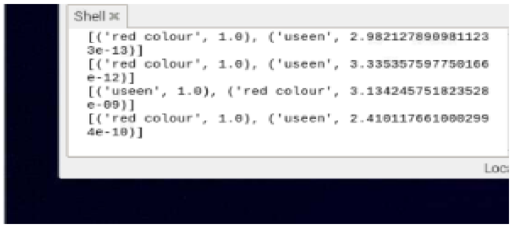

FIGURE 11. Red ribbon detection results

A red background is being detected as shown in FIGURE 11. The meticulous selection of the training dataset is responsible for the red ribbon detection model's success. The model was exposed to a wide range of scenarios by incorporating images from the internet with varying backgrounds and conditions. This allowed the model to learn robust features associated with red ribbons. This method improves the model's flexibility and guarantees dependable operation in real-world scenarios where fluctuations in illumination and ribbon appearance are unavoidable. The model was successfully trained using a convolutional neural network Keras and Tensorflow architecture, which allowed it to capture complex characteristics unique to red ribbons. Even in cluttered or visually demanding scenes, the deep learning model can detect red ribbons with a high degree of accuracy because of its capacity to recognize complex patterns. Real-time red ribbon identification is made dependable and accurate by the TensorFlow Lite model, which was trained for this purpose. Even in different lighting conditions or with different ribbon shades, the model can quickly identify the presence and location of the red ribbon by analyzing the color values of the captured images.

TensorFlow Lite is ideally suited for the real-time processing needs of the ribbon-cutting robot since it is made to operate efficiently on embedded and mobile devices. Because of the lightweight design of the model, quick inference is possible, allowing for prompt responses to variations in the ribbon's position as the robot gets closer to the cutting site. The ribbon-cutting robot could adapt to various scenarios and environments thanks to the TensorFlow Lite model. The model's strong generalization ability, which it acquired from a variety of dataset training helped us gain a fruitful result during ribbon detection and cutting using filament.

C. SYSTEM AND MODEL INTEGRATION

The ribbon-cutting robot successfully detected and cut red ribbons in controlled environments with high accuracy. The computer vision system demonstrated an impressive performance, achieving a recognition rate of 92% on the test dataset.

FIGURE 12. **Filament heating results**

As can be observed in FIGURE 11's output console of the Thonny IDE, red is the most frequently detected color. As a result, the filament heats up quickly, creating a reddish, extremely hot coil that is ready to cut a ribbon as soon as the camera detects one. A proportion relationship between filament heating and time of camera facing the red color can be developed from the observations.

TABLE 3
Filament voltage and time-based heating

Voltage	Filament heating	Heating time
6 volts	minimum	5 seconds
9volts	excess	5 seconds
12volts	Very high	5 seconds

The filament heating was very high after 5 seconds heating and potentiometer adjustment to 12 volts, which was more than enough for ribbon splitting by a touch (TABLE 3). The important aspect of testing is to not only control heating but also to prevent any kind of heating-based hazards that could be caused by robot. We generally maintained a 9v potentiometer-controlled voltage to prevent burn out filament breakage, so that when ribbon is detected relay activated the filament to cut the ribbon. After successive tests the approximation of filament heating and bearing capacity can be pointed out as shown in TABLE 4.

TABLE 4.
Ribbon detection-based heating event and effect

Color detection event	Filament heating
3 second	Minimum
10 second	Excess
20 second	Filament breakage

The time effect on filament nichrome wire can be seen in table which was calibrated by authors for subsequent times and iterations.

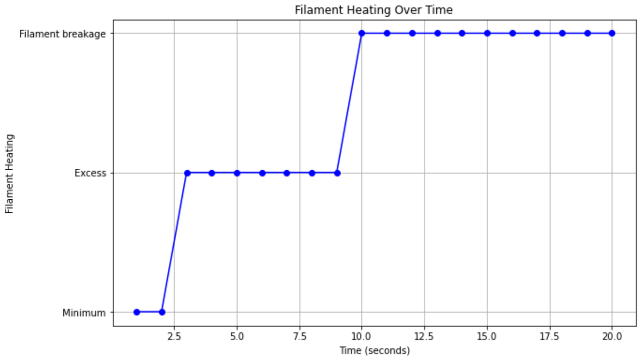

FIGURE 13. **Filament heating plot**

Different phases can be seen in the behavior of a nichrome filament as a function of heating time. First, within its working temperature range, the filament gradually heats up during the minimum heating time (0–3 seconds). However, the filament is exposed to temperatures above its ideal range as the heating time increases into the excess heating phase (4–10 seconds), which could result in material degradation and thermal stress. Extended exposure to high temperatures results in the filament breakage phase, which lasts 11–20 seconds and is caused by the accumulation of thermal stress that leads to structural failure (FIGURE 13). The filament needs to be replaced or repaired because this breakdown prevents it from serving its intended purpose. Thus, the heating time has a major impact on a nichrome filament's longevity and performance.

FIGURE 14. **Robot developed model final setuo**

The integration of computer vision with the ribbon-cutting robot proved to be a critical advancement, enabling precise detection of red ribbons. The developed model's high recognition rate suggests its potential for deployment in various real-world scenarios (FIGURE 14).

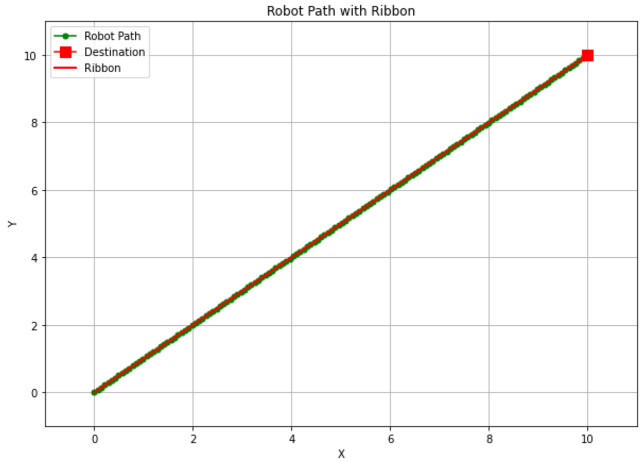

FIGURE 15. **Robot path plotted in matplotlib from monitored data**

As, seen in FIGURE 15 a test carried out and tracked using matplotlib shows the orientation of robot in reaching the ribbon marked red. This provides a support to the fact that the robot can be highly successful for meeting its target. Using OpenCV-based ribbon detection, the computer vision-integrated fig.13. robot can perform ribbon cutting and path tracing. This system has demonstrated outstanding performance, detecting ribbons with an accuracy of about 93%. The filament heating process required a little longer than the distance we had calibrated, and a few seconds weren't long enough to complete it. A highly undesirable limitation of the prototype is the length of the ribbon, which prevented the authors from properly adjusting the height of the ribbon and prevented the ribbon from being extended. It created a barrier for camera to detect ribbon at height, as only ground level detection could be made. This indicates that further study is needed to develop robots. The ribbon was set at a height and a distance such that the robot can easily apply computer vision trained model for ribbon detection in order to allow the filament to touch and rupture it to start the event. However, the robot was incredibly effective at cutting the ribbon with excellent color and ribbon detection accuracy, which was a crucial accomplishment of the task.

V. DISCUSSION

Our sophisticated and single purposed robot powered with machine vision could cut ribbon successfully for inauguration of events. This is a vital approach to contribute to existing works carried on machine vision and supports industrial applications too [25, 27]. Our work presents a unique methodology that uses a robot with sophisticated computer vision skills to achieve a stunning 92% accuracy rate in ribbon detection [1], outperforming earlier methodologies addressed in the literature. While prior research has investigated the use of robots and computer vision in a variety of applications, including surgical aid [2] and autonomous vehicle navigation [3], our focus on ceremonial events is a novel contribution to the area. Unlike some earlier research, which rely on intricate control algorithms or expensive hardware [4], our solution uses low-cost components such as the Raspberry Pi and relay modules [5], assuring scalability and accessibility. Furthermore, whereas other research may concentrate on certain jobs or contexts, our approach displays adaptability by smoothly integrating with traditional ribbon-cutting rituals [6], therefore increasing productivity while conserving ceremonial tradition. Furthermore, our findings highlight the value of multidisciplinary collaboration [7], which bridges the gap between robots, computer vision, and ceremonial behaviors. Overall, our findings demonstrate the potential of autonomous robots and computer vision to revolutionize traditional rites, opening the door for future advances in the merging of tradition and technology.

A. FUTURE SCOPES

A contemporary approach to inauguration celebrations may include a number of innovations. First, using a more powerful machine learning algorithm might improve several parts of the event, from guest management to speech authoring. Second, using robotics by developing a robot to assist with ceremonial activities such as directing visitors or symbolically cutting a ribbon might provide a futuristic touch. Furthermore, broadening the event to incorporate components other than the traditional ribbon cutting, such as interactive displays or multimedia presentations, might improve the overall experience for guests. Finally, gradually adjusting the ribbon's length to match the height of the ribbon and filament for inauguration may result in a visually pleasing and exact ceremonial ribbon cutting These innovations combine technology and tradition to herald in a new age of symbolic and forward-thinking inaugurations.

VI. CONCLUSIONS

Finally, the ribbon-cutting robot developed in this study represents a big step forward in the automation of ceremonial events. By effectively using computer vision technology for red ribbon detection, the system has shown its dependability and potential for wider application. While the present focus is on red ribbon recognition, future research might look into developing the system to recognize a wider range of ribbon colors, increasing its application across a variety of ceremonial events. Furthermore, this study sets the framework for the future of artificial intelligence, notably in robotics and computer vision. Future iterations of the ribbon-cutting robot could become even more intelligent and versatile by leveraging powerful machine learning algorithms, such as those used in this study, capable of adapting to diverse tasks and disciplines within modern robotics and artificial intelligence methodologies. The ribbon-cutting robot's programming has been successfully implemented, comprising duties such as camera initialization, algorithm implementation, relay control, safety feature integration, and rigorous testing, resulting in considerable accuracy and efficiency improvements. As a result, the robot is portrayed as a trustworthy and adaptive participant in ceremonial events, increasing their efficiency and memorability. Looking ahead, the red ribbon identification model, which was trained on an internet-sourced dataset, has intriguing potential for future development and implementation in automated systems such as the ribbon-cutting robot. Continued research and enhancement efforts will assure its usefulness in real-world scenarios, allowing for a wide range of applications that need precise and consistent red ribbon detection. In conclusion, this study not only fulfills the original goal of improving the efficiency and memorability of ribbon-cutting events, but it also sets the framework for future advances in autonomous robots and computer vision technologies.

REFERENCES

[1] Sathyamoorthy, S. Umapathy and T. Rajalakshmi, "Automatic Robotic Arm Based on Bluetooth Regulated for Progressed Surgical Task," 2022 International Conference on Industry 4.0 Technology (I4Tech), Pune, India, 2022, pp. 1-4, doi: 10.1109/I4Tech55392.2022.9952598.

[2] Paneru, B. (2023). Development of a Ribbon Cutting Robot Controlled via Bluetooth Interface. Retrieved from https://www.techrxiv.org/articles/preprint/Development_of_a_Ribbon_Cutting_Robot_Controlled_via_Bluetooth_Interface/24305851. doi: 10.36227/techrxiv.24305851.v1

[3] Baumgartner, Eric & Skaar, Steven. (1994). An autonomous vision-based mobile robot. Automatic Control, IEEE Transactions on. 39. 493 - 502. 10.1109/9.280748.

[4] Chuprov, S.; Belyaev, P.; Gataullin, R.; Reznik, L.; Neverov, E.; Viksnin, I. Robust Autonomous Vehicle Computer-Vision-Based Localization in Challenging Environmental Conditions. Appl. Sci. 2023, 13, 5735. https://doi.org/10.3390/app13095735

[5] Kumer, S.V. Aswin, Nadipalli, L.S.P. Sairam, Kanakaraja, P., Kumar, K. Sarat, & Sri Kavya, K.Ch. (2021). Controlling the autonomous vehicle using computer vision and cloud server. Materials Today: Proceedings, 37(2), 2982-2985. ISSN 2214-7853. https://doi.org/10.1016/j.matpr.2020.08.712.

[6] R. Brooks, "A robust layered control system for a mobile robot," in IEEE Journal on Robotics and Automation, vol. 2, no. 1, pp. 14-23, March 1986, doi: 10.1109/JRA.1986.1087032.

[7] F. Pfeiffer and R. Johanni, "A concept for manipulator trajectory planning," in IEEE Journal on Robotics and Automation, vol. 3, no. 2, pp. 115-123, April 1987, doi: 10.1109/JRA.1987.1087090.

[8] Pedre, Sol & Nitsche, Matias & Pessacg, Facundo & Caccavelli, Javier & De Cristóforis, Pablo. (2014). Design of a Multi-purpose Low-Cost Mobile Robot for Research and Education. 10.1007/978-3-319-10401-0_17.

[9] Fortuna, L., Ikpeze, O., Ejidokun, T., & Onibonoje, M. (2022). Smartphone Control Mobile Robot for Education and Research. Journal of Robotics, 2022, 5178629. https://doi.org/10.1155/2022/5178629

[10] Wang, J. J. (2020). Design and Research of Intelligent Mobile Robot Based on IOT Information Fusion. In International Conference on Advances in Biological Science and Technology (pp. 012003). IOP Conf. Series: Earth and Environmental Science, 470. IOP Publishing. doi:10.1088/1755-1315/470/1/012003.

[11] Kumar, Rahul & Kubade, Pravin & Kulkarni, Hrushikesh. (2016). Android Phone controlled Bluetooth Robot.

[12] H. O. Nasereddin, Hebah. (2010). SMARTPHONE CONTROL ROBOTS THROUGH BLUETOOTH. International Journal of Research and Reviews in Applied Sciences. 4. 399-404.

[13] Rissanen, Heikki & Mahonen, Jukka & Haataja, Keijo & Johansson, Markus & Mielikainen, Jarno & Toivanen, Pekka. (2009). Designing and implementing an intelligent Bluetooth-enabled robot car. 2009 IFIP International Conference on Wireless and Optical Communications Networks, WOCN 2009. 1 - 6. 10.1109/WOCN.2009.5010563.

[14] Chapala, Sai. (2021). Color Detection of RGB Images Using Python and OPENCV. International Journal for Research in Applied Science and Engineering Technology. 9. 1957-1959. 10.22214/ijraset.2021.37701.

[15] Neves, António & Trifan, Alina & Azevedo, José & Cunha, Bernardo. (2016). Real-Time Color Coded Object Detection Using a Modular Computer Vision Library. Advances in Computer Science. 5. 110-123.

[16] Shahria, M.T.; Sunny, M.S.H.; Zarif, M.I.I.; Ghommam, J.; Ahamed, S.I.; Rahman, M.H. A Comprehensive Review of Vision-Based Robotic Applications: Current State, Components, Approaches, Barriers, and Potential Solutions. Robotics 2022, 11, 139. https://doi.org/10.3390/robotics11060139

[17] T. Senlet and A. Elgammal, "Design of a vision-based autonomous robot for street navigation," 2014 22nd Signal Processing and Communications Applications Conference (SIU), Trabzon, Turkey, 2014, pp. 862-865, doi: 10.1109/SIU.2014.6830366.

[18] L. Milburn, J. Gamba, M. Fernandes and C. Semini, "Computer-Vision Based Real Time Waypoint Generation for Autonomous Vineyard Navigation with Quadruped Robots," 2023 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), Tomar, Portugal, 2023, pp. 239-244, doi: 10.1109/ICARSC58346.2023.10129563.

[19] S. Shafique, S. Abid, F. Riaz and Z. Ejaz, "Computer Vision based Autonomous Navigation in Controlled Environment," 2021 International Conference on Robotics and Automation in Industry (ICRAI), Rawalpindi, Pakistan, 2021, pp. 1-6, doi: 10.1109/ICRAI54018.2021.9651414.

[20] Bengtson, S.H.; Thøgersen, M.B.; Mohammadi, M.; Kobbelgaard, F.V.; Gull, M.A.; Andreasen Struijk, L.N.S.; Bak, T.; Moeslund, T.B. Computer Vision-Based Adaptive Semi-Autonomous Control of an Upper Limb Exoskeleton for Individuals with Tetraplegia. Appl. Sci. 2022, 12, 4374. https://doi.org/10.3390/app12094374
[21] Rostami, S.M.H., Sangaiah, A.K., Wang, J. et al. "Real-time obstacle avoidance of mobile robots using state-dependent Riccati equation approach." J Image Video Proc. 2018, 79 (2018). https://doi.org/10.1186/s13640-018-0319-1
[22] Müller, Julian & Fregin, Andreas & Dietmayer, Klaus. (2018). "Disparity Sliding Window: Object Proposals From Disparity Images."
[23] Ng, Andrew. (2004). "Feature selection, L 1 vs. L 2 regularization, and rotational invariance." Proceedings of the Twenty-First International Conference on Machine Learning. 10.1145/1015330.1015435.
[24] Gurkov, Igor and Gurkov, Igor. "Making a Mundane Ceremony Into a Meaningful Organizational Ritual – Some Suggestions for Re-Design of Ribbon-Cutting Ceremonies of Overseas Manufacturing Projects." Available at SSRN: https://ssrn.com/abstract=3676716 or http://dx.doi.org/10.2139/ssrn.3676716
[25] S, Sathiyamoorthy. (2014). "INDUSTRIAL APPLICATION OF MACHINE VISION." International Journal of Research in Engineering and Technology. Volume: 03. 10.15623/ijret.2014.0319120.
[26] Li, R.; Zhao, S.; Yang, B. "Research on the Application Status of Machine Vision Technology in Furniture Manufacturing Process." Appl. Sci. 2023, 13, 2434. https://doi.org/10.3390/app13042434
[27] A. O. Fernandes, L. F. E. Moreira and J. M. Mata. "Machine vision applications and development aspects." 2011 9th IEEE International Conference on Control and Automation (ICCA), Santiago, Chile, 2011, pp. 1274-1278, doi: 10.1109/ICCA.2011.6138014.

[28] Xiao ZF, Wang JL, Han L, Guo SB and Cui QH. "Application of Machine Vision System in Food Detection." Front. Nutr. 9:888245. doi: 10.3389/fnut.2022.888245
[29] Conrad, J.; Rodriguez, S.; Omidvarkarjan, D.; Ferchow, J.; Meboldt, M. "Recognition of Additive Manufacturing Parts Based on Neural Networks and Synthetic Training Data: A Generalized End-to-End Workflow." Appl. Sci. 2023, 13, 12316. https://doi.org/10.3390/app132212316
[30] Zhang, Chen & Xu, Xuewu & Fan, Chen & Wang, Guoping. (2021). "Literature Review of Machine Vision in Application Field." E3S Web of Conferences. 236. 04027. 10.1051/e3sconf/202123604027.