RESEARCH PROJECT IN MECHATRONICS ENGINEERING

Autonomous Routing and Connection of Multi-Core Cables into Prefabricated Walls

Russell Feng

Project Report ME003-2021

Co-worker: Zach Macleod

Supervisor: Dr Minas Liarokapis

Department of Mechanical Engineering
The University of Auckland

15 October 2021

ME003-2021

Autonomous Routing and Connection of Multi-Core Cables into Prefabricated Walls

Russell Feng

ABSTRACT

The demand for affordable housing is increasing rapidly. Driven by high labour costs, home prices are climbing ever higher. Therefore, some new methods are introduced to reduce construction costs. A solution is automatic pre-fabrication of homes, improving construction efficiency in terms of time and labour cost. This project focuses on automating handling and routing of electrical calbes in pre-fabricated walls, employing a sophisticated robotic end-effector, a path planning scheme to appropriately control the robot arm, and specially design connector solutions for connecting individual cable cores in 3-way junctions.

The proposed robotic end-effector is equipped with a cable feeding mechanism and it can press-fit the electrical cables into narrow channels created in concrete pre-fabricated walls. The end-effector shows reliable and repeatable operation. The research also presents two types of connector designs that can facilitate connection of the cable cores in the junctions of pre-fabricated walls. The installation of connectors is based on a machine vision framework implemented in MATLAB. The machine vision framework can identify the colours of cables and locate the cables.

Robot programming and path planning relied on kinaesthetic teaching. The results showed the speed of routing using robotic arms is far exceed manual routing.

DECLARATION

Student

I …..Russell Feng………………………………….. hereby declare that:

1. This report is the result of the final year project work carried out by my project partner (see cover page) and I under the guidance of our supervisor (see cover page) in the 2021 academic year at the Department of Mechanical Engineering, Faculty of Engineering, University of Auckland.
2. This report is not the outcome of work done previously.
3. This report is not the outcome of work done in collaboration, except that with a project sponsor as stated in the text.
4. This report is not the same as any report, thesis, conference article or journal paper, or any other publication or unpublished work in any format.

In the case of a continuing project: State clearly what has been developed during the project and what was available from previous year(s):

This project focuses on automating handling and routing of electrical calbes in prefabricated walls, employing a sophisticated robotic end-effector, a path planning scheme to appropriately control the robot arm, and specially design connector solutions for connecting individual cable cores in 3-way junctions.

Signature: Russell Feng

Date: 15/10/2021

Supervisor

I confirm that the project work undertaken by this student in the 2021 academic year **is / is not** (*strikethrough as appropriate*) part of a continuing project, components of which have been completed previously.

Comments, if any:

Signature:

Date: 15/10/2021

Table of Contents

List of Figures

Acknowledgements

I would like to thank the following:

Zach Macleod, my project partner, for his passion for the project, excellent critical thinking and professional engineering skills

Dr Minas Liarokapis, our project supervisor, for the time and guidance throughout the project and many suggestions given to us.

Zuru Tech for their opinions on the project.

Glossary of Terms

Bowden tube	Hollow plastic hose
Channels	The depression in the pre-fabricated wall. The cable will be located in the channel
End-effector	the device at the end of a robotic arm, designed to interact with the environment
Pre-fabricated walls	Processed into reinforced concrete slab components for building assembly in prefabricated factories (fields) or construction sites
Stepper motor	a brushless DC electric motor used for providing torque.
Stripping machine	Automatic machine that can strip the electrical insulation from electric wires.

Abbreviations

IPCs	Insulation Piercing Connectors
IDCs	Insulation Displacement Connectors

1. Introduction

Recently, New Zealand's demand for affordable housing has been increasing. Due to the high labour costs, construction companies urgently need to find a new way to reduce labour costs and improve construction efficiency. Therefore, prefabricated houses can be seen as a potential solution. Suppose a more automated process can be used in the production process of prefabricated houses. In that case, the production speed of prefabricated houses will be further increased, providing more New Zealanders with affordable housing.

Prefabricated buildings are a recent trend and widely used to increase on-site construction quality and efficiency. Prefabricated buildings are even more popular in developing countries because they are affordable and can be assembled quickly [4]. To further improve the efficiency, reliability and affordability of prefabricated buildings, it is critical to simplify the manufacturing and assembly processes . Also, reducing manual processes can minimize costs and better promote prefabricated buildings in developing countries. Furthermore, one of the most critical steps during manufacturing is the manipulation of electrical components. Therefore, it is valuable to fit the electrical building elements such as cables into the prefabricated wall automatically.

Besides, Cables are long and flexible objects and sometimes have small diameters. They are prone to deformation and vibration due to external forces and changes in the environment.

Therefore, this project is to design and develop a system that can place, lay, and connect electrical cables in the pre-fabricated concrete walls. These walls can be used in many affordable building solutions. Since the ideal application scenario of the project involves a "lights off" factory, the system needs to be highly automated. The system contains a robotic gripper for cable routing, a cable feeding mechanism, a stripping machine and a path planning system. The designed robotic gripper will be suitable for manipulating flexible objects such as cables. This project will not consider using the gripper to grasp other rigid objects.

By researching the literature, the initial idea was to use computer vision for path planning. However, in reality, this method has encountered many difficulties. Besides, the path of the channels will be known, so the project can rely on an end-effector that will push-fit a cable into a channel. Therefore, computer vision is not required anymore.

To simplify the project and make the routing more efficient, this project plans to use an automatic wire pre-feeding machine to feed the cables into the gripper. In other words, more considerations shifted to transporting the cables before entering the end effector. Routing will then rely only on the movement of the end-effector instead of using the robotic gripper to operate the cable as a whole, such as bending the cable. The project will also consider the design of junctions for the multi-core cables.

A software program will also be designed to control the sensors and actuators on the robotic arm and gripper. Motion planning is also critical to identify a best-fit solution, helping the system finish manipulating cables.

The project will use the Kuka KR16 robot arm. The project should consider the specific environment of Zuru factories in China because Zuru Tech sponsors it. Therefore, test results and the whole result will be sent to Zuru to assist their further development.

Section 3 describes the selected equipment that supported the development of the system. Section 4 demonstrates the designed end-effector and connectors. Section 5 describes the path planning method and feeding mechanism used in routing and the machine vision system used at junctions.

1.1 Research Objectives

This project designed an end-effector that can handle, manipulate and route the cable. Then use path planning to control the robot arm and finally the connection of individual cable cores at 3 way junctions.

The detailed research objectives are listed below:

- Investigate what design characteristics are needed for the development of robotic grippers that can facilitate the execution of robust grasping and manipulation with flexible cables.
- Design and develop a robotic cable handling and installation mechanism that will be efficient and robust.
- Analyse what perception methods are needed to identify and localize flexible cables.
- Design and develop appropriate robotic perception techniques to facilitate cable handling and installation.
- Investigate what routing and control methods can be employed for the execution of cable routing tasks.

2. Related Work

2.1 Robotic Grippers

A simplified description of grasping is the ability to pick up and hold an object against external disturbances, while manipulation is the ability to exert forces on an object and thus cause its rotation and displacement with respect to the reference frame of the manipulator [5]. Most robotic grippers can provide functions for picking up and manipulating objects. According to requirements, the object can be rotated or displaced. For more complex picking and manipulation requirements, robotic hands need to have the ability to perceive the outside world. Therefore, the end effector of the robotic gripper should be generally able to sense and perceive the characteristics and attributes of the picked-up object [5].

Therefore, the trend of robotic hands is to perceive the posture of the object, and more importantly, adapt to the shape of the object. Therefore, it is possible to control the robot hand not to exert excessive force and cause damage to the object [6, 7]. In unstructured environments, to detect and understand the object and the environment, more complicated grippers are needed. However, heavy grippers are difficult to control the contact force [7]. In [7], it stated a human-like structure gripper. It proves that the kinematic structure of the human hand can improve the grasping power and control ability.

2.1.1 Materials affecting robotic grippers

The material properties of the gripper will affect the performance of it, such as the maximum force and response time that can be provided. Therefore, the use of different compliant materials can make the robot have different performance [5]. If use materials that can change in response to changes in the outside world, the function of the gripper can be greatly expanded, such as shape memory alloys (SMAs) and Ionic materials [5].

2.1.2 Cable grasping

Recent research stated that it is possible to install a touch sensor on the gripper. The idea is to simulate a human finger through a multi-functional sensor. The sensor can sense surface features such as multi-directional force and temperature [6].

Unlike grasping rigid bodies, grasping a cable is more challenging. Cables are more flexible and deformable, which means their behaviour is more unpredictable. When applying external forces to the cable, it will deform into different shapes [1, 8–10]. Therefore, many techniques and methods for the manipulation of rigid bodies cannot be applied on cables [9]. Furthermore, self-collision is likely to happen between themselves [8].

Besides, cable contour following is also a typical way of positioning and gripping the cable. This can be achieved by attaching a tactile sensor to the gripper. Since the cable is deformable, to get a certain length, the gripper needs to move from one end towards the other end. The gripper will slide along the cable until it reaches the target position [11] (Figure 1).

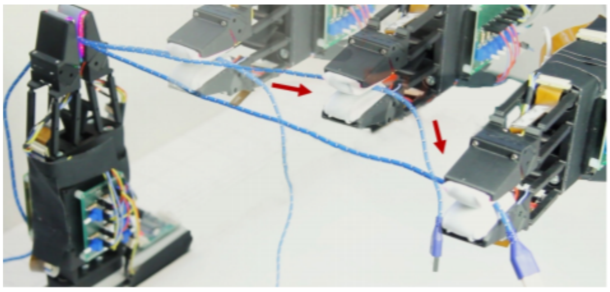


Figure 1 Following a cable using a robotic gripper.

However, the cableâs shape changes dynamically with the sliding motion, so mechanical constraints must be applied to the cable in most cases. This allows the cable to be under a quasi-static condition [11, 12]. If any obstacles near the cable or the working area are relatively small, contour following can get challenging. Furthermore, the required mechanical constraints may be different from time to time.

If the cable needs to be placed in a random shape or not straightly, it can be achieved by considering gripper rotation or multiple grippers grasping together. According to the cable shape, some characteristic parameters are formulated. Then these parameters are used to find the relationship with the robot movement. The model was established to help to derive the control requirement that motivates the cable to become a specific shape [13]. However, this model only considered in a 2D dimension, [13] does not manipulate on a 3D plane.

2.1.3 Cable insertion

Unlike grasping the cable, cable insertion always needs to find the plug (always at the end of the cable). By applying a certain gripping force on the cable, due to friction on the cable surface, the robotic gripper will not lose control of the cable but can slide on it [11].

When multiple cables need to insert, it is critical to arrange the priority of cable insertion, alignment and grasping. Because multiple cables cannot be manipulated simultaneously, the gripper will likely lose control of cables when doing the next step. For example, during the manipulation step, the cable may escape from the gripper. Therefore, [1] presented a two-level structure of robotic manipulation (Figure 2, [1]). Level-I represents the process from grasping to manipulation. It gives grasping a higher priority, which means the gripper will continue grabbing the cable even though the physical contact with the cable is lost during the manipulation step. Level-II is the choice of different cables, determining which one is going to grasp first.

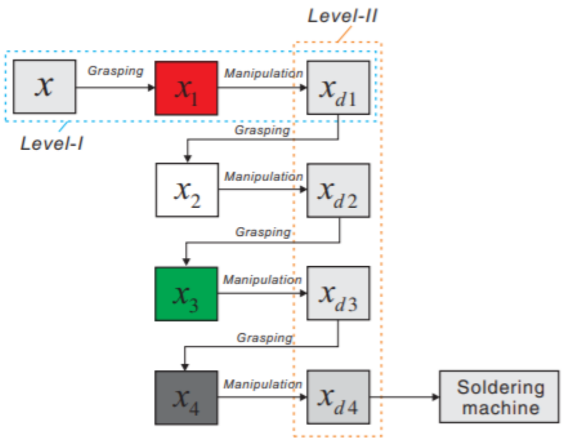


Figure 2 Two-level structure [1].

2.2 Vision System

A vision system is a key to locate and estimate the cable. It is essential to process the information to the gripper to evaluate the forces applying to the object and the aperture of the gripper [6]. Using optical sensors for modelling could be a solution. It can obtain the weight distribution and size of the object. At the same time, it can get the specific location of the object [14]. However, for cable manipulation, it is not necessary to render the entire cable with high resolution. The reason is the mass of cable is evenly distributed.

The vision system usually works together with the tactile system. Before the robotic gripper touches the cable, the vision system is responsible for detecting the status of the cable. When the robotic gripper makes any connection with the cable, it will switch to the tactile mode and then senses the cable using a tactile system [15]. However, when the robotic gripper nearly touches the cable, it may obstruct the vision system, causing the vision system to lose its ability to detect objects. At this time, the tactile system has not yet taken effect, so that it may cause the failure of grasping [6, 9].

Before grasping the cable, the system must first obtain the current shape of the cable and determine the grasping point. [2] predict the gripping width required for each grasp to avoid potential collisions. This gives a much better prediction of the grasping point and corresponding robotic gripper performance (Figure 3).

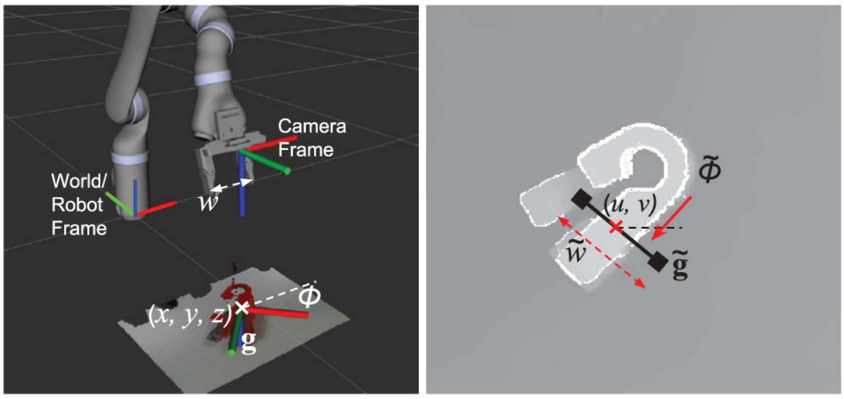


Figure 3 A depth image of the scene [2].

The current shape of the cable can be determined by taking multiple images from different angles to get the depth information of each point. It is possible only to triangulate one point within each image, rather than triangulate every point. Therefore, there is no need of building a whole 3D model of the object [16]. This applies well to translucent or reflective objects. The vision system can focus on the part of the object first then simulate the whole object. The system can also learn from synthetic data that is automatically labelled with the correct grasps [16]. It is also possible to track the position and orientation of a cable by the gravity feedback characterization of the cable [17].

Once people have got the image from the vision system, two distinct vision algorithms are presented to get the position and orientation of the cable. The first approach is binary imaging techniques. By deciding a threshold value, the computer can divide the image into plenty of small elements. Then the computer can further analyze the image to get the shape of the cable [18]. The second approach relies on the analysis of object boundaries generated using a dynamic contour-following Algorithm [18].

Visual servoing is another kind of closed-loop control using vision feedback. It can adapt to the external environment [2]. [1] stated that there is a relationship between the task space of the gripper and the image space of the camera. This model is called a pinhole camera model. It relates the image with the target space, which can help determine the torque of the gripper [2].

2.3 Motion Planning

When the environment is full of obstacles or multiple cables need to cover each other, a path must be planned not to damage the existing cables or cable routing channels [14].

In the current work, there is a discussion on a probabilistic roadmap that can be constructed using computational models by considering the kinematic model of robot arms [8]. This can solve the problem of cable knots by detecting whether there are obstacles in the path. It is also critical to determine the boundary conditions. When there are any obstacles, the differential equation representing the cable will be different [19]. Prediction of the shape itself can be achieved by using higher-order differential equations, which can be calibrated experimentally [12].

[20] further developed the research to real-time online planning. It simulates and obtains global data by analyzing the real-time data of the part of the object. It approximates the relationship between the desired motion of the cable and the real-time velocity of the robot end-effectors.

3. Equipment

3.1 Wall Structure

The prefabricated walls will be built in the "lights off" factory. Since this prefabricated wall will be put into industry, these walls will be mass-produced. Therefore, the channels in the wall will be predetermined. The channels of the wall include straight lines, curves and 3-way junctions. The junction will help the connection of individual cable cores.

Figure 4 indicates the positions of cable channels and the 3-way junction. Cables will be installed into these tight channels.

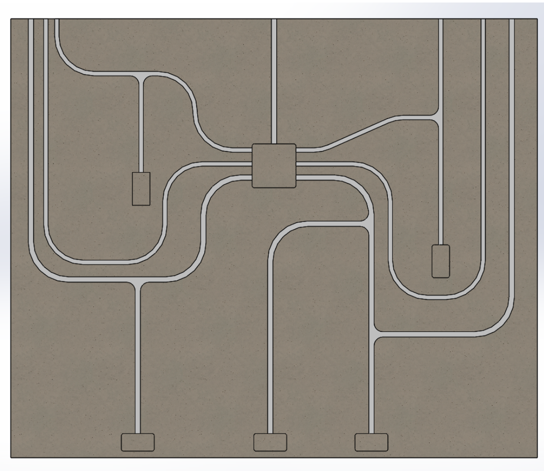


Figure 4 Sample channel geometry of prefabricated wall. Source: ZURU Tech Ltd.

3.2 Cable Stripping

Before the cable is routed to the channel in the wall, the outer plastic casing at both ends of the cable needs to be removed. Many different stripping methods have been considered, such as laser wire cutting, chemical stripping method and existing stripping machine.

The following table demonstrates the features of each stripping method (Table 1).

Stripping Methods	**Advantages**	**Disadvantages**
Laser	Fast	Manual removal is required
Chemical	Can strip multiple cables at the same time (Parallel)	May damage the wire and hard to control.
Ultrasonic	Automatic removal	Only good for cables with small diameter and very uncommon.
Kingsing Stripping Machine (KS-W1022)	Suitable for multi-core cable cutting and Automatic removal	Need to purchase from China

Table 1 Pros and Cons of different stripping methods.

By comparing the features of each stripping methods, Kingsing Stripping Machine **(Multicore Cable Cutting and Stripping Machine KS-W1022)** is selected which can greatly simplify the stripping process (Figure 5). It is purchased from overseas for cutting and removing the plastic insulation of the cable. Furthermore, since the cable used is a multi-core cable, it also satisfied the requirement of cutting and removing the core plastic insulation inside the cable.

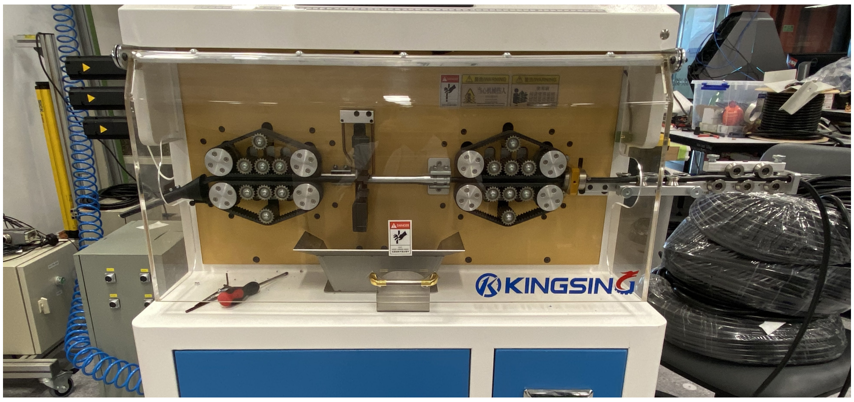
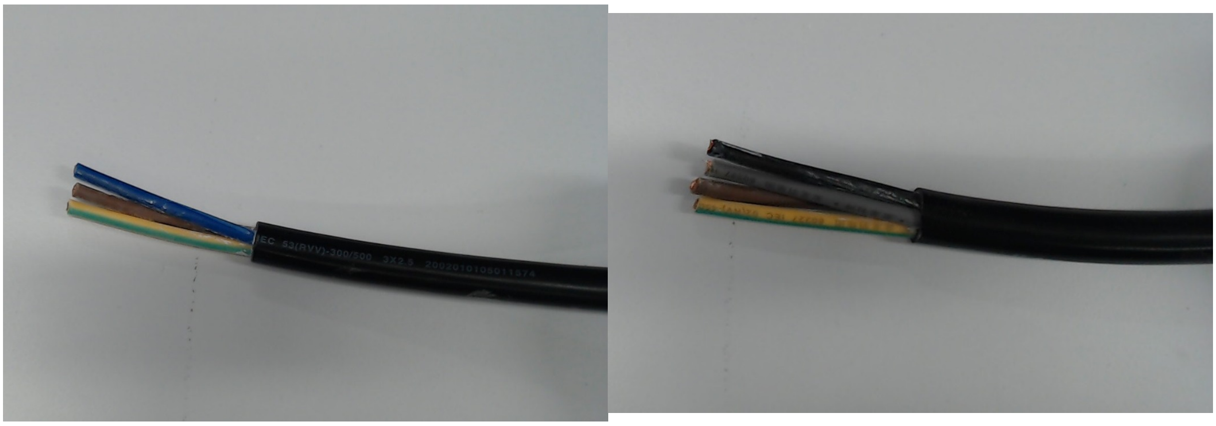


Figure 5 Stripping Machine KS-W1022.

The number of cores of the cables used in this project are 3 and 4 respectively. The result of the cable after stripping process is shown in Figure 6.



(a) 3-core cable

(b) 4-core cable

Figure 6 Cables after stripping process.

4. Design

The cable routing process includes the following steps. The cable spool is to be connected to a wire stripping machine. The cable processed by the wire stripping machine is the cable that is ready to be routed. Simultaneously, they will be transported into the end-effector, which will use a controlled stepper motor to feed the cable. The end-effector on a robotic arm routes the cable into the channels within the prefabricated wall and fully press-fits them. Another mechanism will install the junctions between the pre-routed cables. Figure 7 shows the flowchart of the process. The rest of the section will describe the implementation of each step in detail.

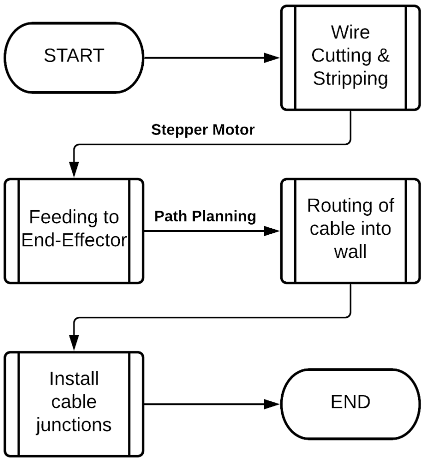


Figure 7 Flowchart of objective complete process.

4.1 End-effector Design

4.1.1 Design of the extruder frame

The end-effector at the top of the robotic arm is an extruder. Its function is to grasp the cable then press and insert it into the channel of the prefabricated wall. The top of the extruder and the robotic arm will be mounted, so there is no relative movement between them. The extruder will wholly follow the direction of the robotic arm. The end-effector is composed of an extruder frame, some wheels and a stepper motor (Figure 8). The cable will be inserted into the gap of the frame of the extruder (there is a specific entrance). The cable will be fed and pressed into the channel through the rotation of wheels in the middle of the frame. The Stepper motor will provide additional torque to assist the forward movement of the cable. All wheels are passed through the middle by threaded axles. The

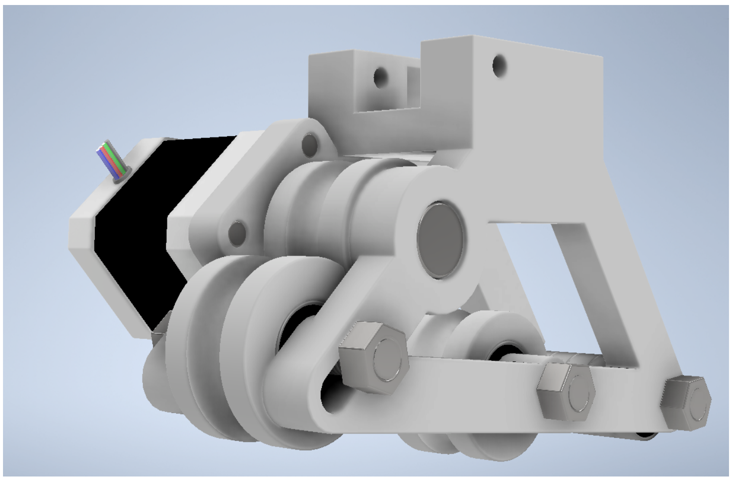
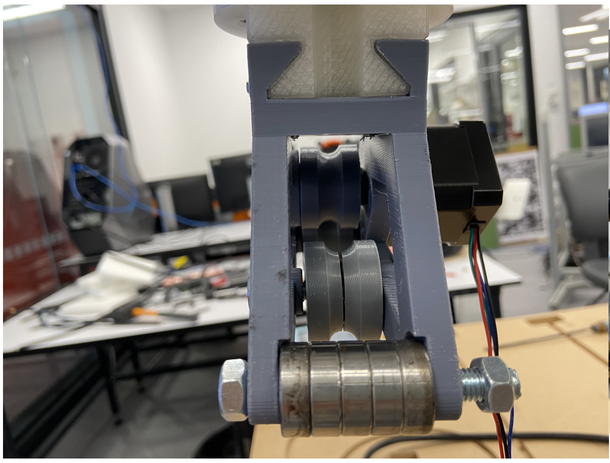


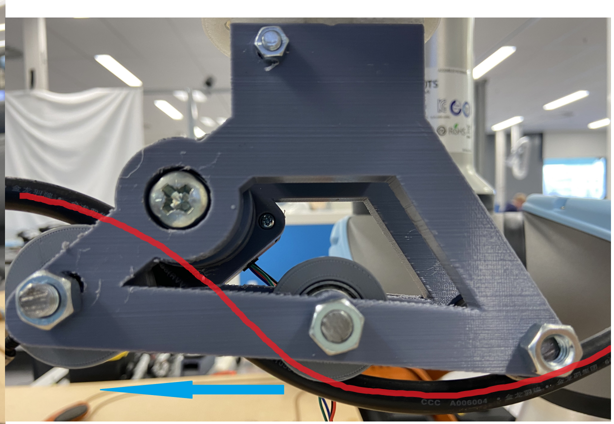
Figure 8 Render of end-effector design.

threaded axle will also pass through the holes in the frame. Nuts fix both ends. Therefore wheels can be fixed in the frame, and the frame can provide support force to the wheels.

Figure 9 below shows the prototype of the end-effector. The red arrow in Figure 9(b) indicates then position of the cable. The blue arrow indicates the movement direction of the end-effector (together with the robot arm).



(a) Front view of the end-effector.



(b) Side view.

Figure 9 Actual photo of end-effector design.

4.1.2 Design of the wheels

The design consists of 4 rollers: an aligning wheel, press roller, driven wheel and loaded wheel. Figure 10 shows an exploded view of the end-effector assembly. The position of the wheels and the stepper motor is shown in the figure as well.

The aligning wheel (Balloon No.1) can push the cable into the channel by 2mm. It will instruct the direction for other wheels behind. Therefore, its groove will be aligned with the channel. It is split into two halves that can rotate individually. Hence when it moves on a curve, there will be a gap and a small displacement between two halves. Therefore, it can eliminate slip when turning.

The press roller (Balloon No.2) will push the cable entirely into the channel. It will come into contact with the wall. At the same time, the cable will be above the channel. As the press roller moves along the channel path, the cable will be completely pressed into the channel. It is a group of 5 bearings that can rotate individually so that slip can be minimum. Two MATLAB functions are created to calculate the required number of bearings and the maximum radius of the roller (Appendix A1 and A2).

The driven wheel (Balloon No.3) is connected to a stepper motor. The gap between the driven wheel and the loaded wheel forms the entrance for the cable. Therefore, the cable enters the end-effector between the driven wheel and the loaded wheel. The effect of the stepper motor will be discussed in Section 4.4. The driven wheel can better locate the cable and assist in feeding the cable.

The loaded wheel is elastically pulled into the driven wheel with elastic bands to create higher friction.

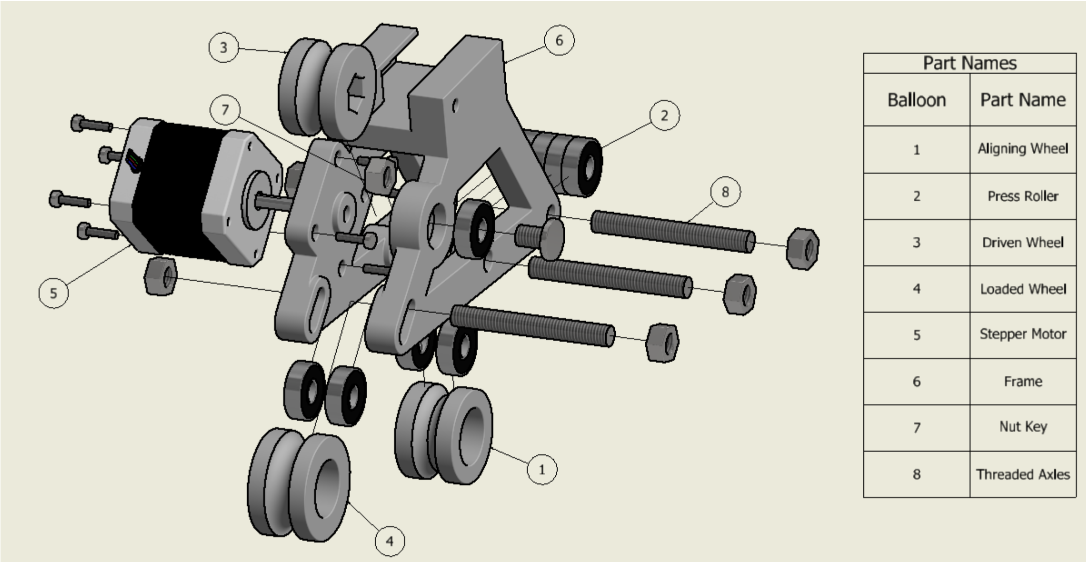


Figure 10 Exploded view of the end-effector assembly.

4.1.3 Connection with the robot arm

The frame is not directly mounted to the robotic arm but through a connector (Figure 11). The four small holes on the top of the connector will match the positions of the holes at the end of the robot arm so that it can be fixed to the end of the robot arm with screws. However, the frame and the connector are connected by "insertion", and only one screw is fixed horizontally. Through experiments, it can be determined that as long as the accuracy of 3D printing is sufficient, this fixing method is very stable. However, this also makes it difficult to remove it from the frame.

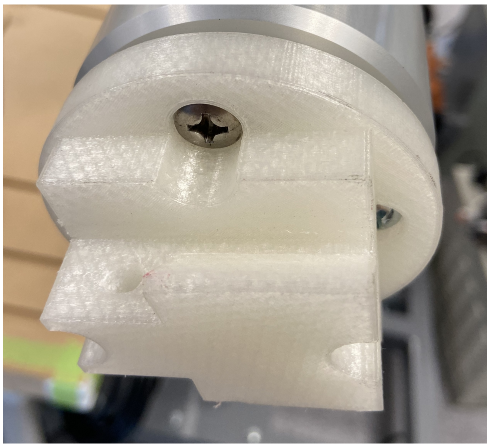


Figure 11 The actual photo of the 3-D printed connector.

4.2 Connector Design

The connecting process has to be done in a minimal and tight area in the wall (Figure 12), which is the position of the junction. Since the maximum number of cores of one single cable is 4 (Figure 6b), there are up to 8 cores terminating in the junction area, and each core has 30A of current.

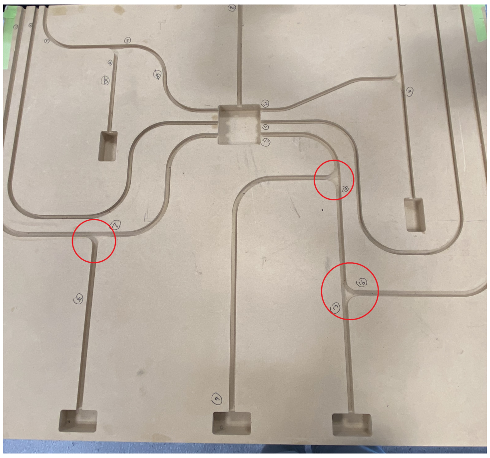


Figure 12 The actual photo of the wall. Red circles indicate the positions of the junction.

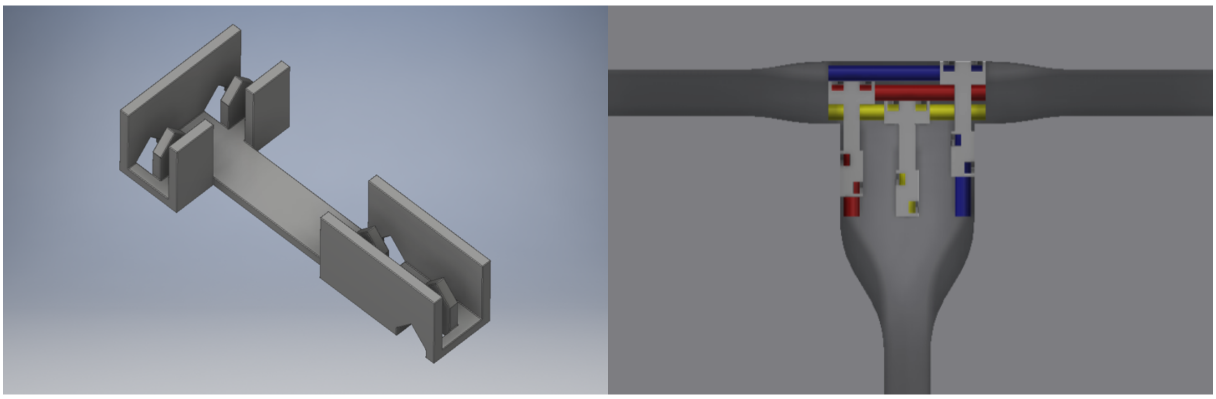
Also, there are two types of cable that are used: stranded-core and solid-core. Therefore, the junction connector must have the capability of minimizing thin copper strands fraying and shorting with other connections.

As a result, two different types of connectors were chosen: Insulation Piercing Connectors (IPCs) and Insulation Displacement Connectors (IDCs). These connectors allow cables to be connected without removing the insulation layer of the core wire (but still need to remove the outermost insulation layer of the cable). Therefore, the wires inside the core wire will not become messy and frayed. This reduces the risk of short and open circuits.

4.2.1 Insulation Piercing Connectors

IPC connection is suitable for the connection of stranded-core data cables. A conductive needle penetrates the core wire insulation and enters the cable. Its material is metal. IPC is wider than the insulated core of the cable.

Another robotic arm will be used. It will handle the IPC and press it into the cable from directly above, at the position of the junction. This process occurs after the routing is completed. After the IPC is pressed in, the epoxy resin will be poured on the connection to avoid unexpected connections between different cores (Figure 13a). This design is too wide, so the carved channel would have to be modified to a different reconfiguration (Figure 13b).



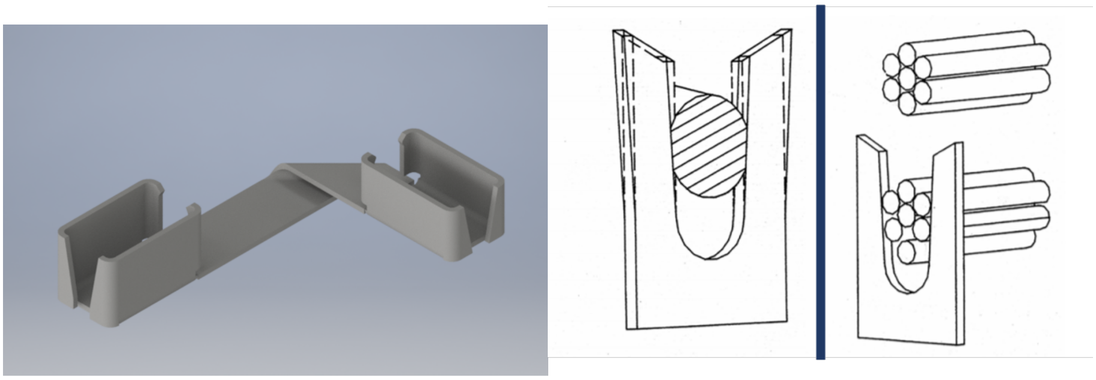
(a) Proposed IDC Design **(b)** Proposed Channel Reconfiguration

Figure 13 Design of IPC

4.2.2 Insulation Displacement Connectors

IDC connection is suitable for both stranded-core and solid-core cables. It is also made of metal, and it has similar width as the IPC connection. The installation method is also similar to that of the IPC connection. The robotic arm will press the connector down around the cable. Epoxy will insulate the connection point (Figure 14a).

Even though it is suitable for both types of cables, solid-core cables have better performance than stranded-core cables because solid-core cables are not prone to deformation and disorder (Figure 14b [21]).



(a) Proposed IDC Design.

(b) Comparison of solid and stranded-core IDCs (Left: Solid, Right: Stranded.) [21]

Figure 14 Sample image processing picture.

5. Methods

5.1 Stepper Motor Control

The stepper motor is to push the cable into the end-effector. If a stepper motor is not used, the transmission of the cable will rely entirely on the friction between the wheels. However, since the end-effector has to adapt to the pressing when making a curved motion, the distance between the wheels will change, which makes the friction force unstable. Therefore, the stepper motor can reduce the slip between the cable and the wheel.

A DRV8825 stepper motor and a 12V power source (Figure 15). The output speed of the stepper motor will match the moving speed of the robotic arm, which can be achieved through Arduino programming. A capacitor is placed over the rails of the 12V power source to protect the driver from power surges.

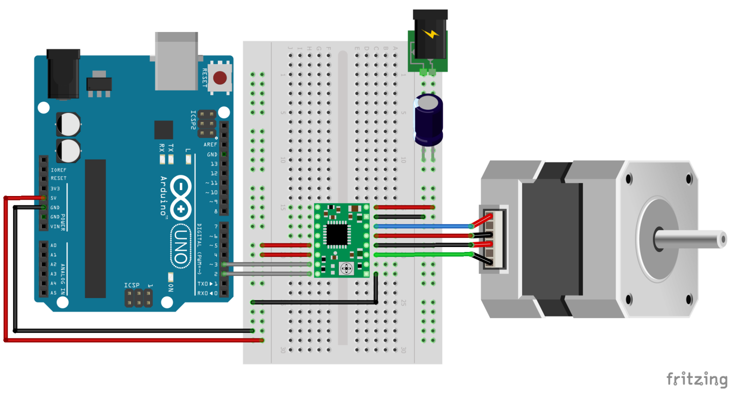


Figure 15 Stepper motor circuit diagram.

5.2 Path Planning

Path planning has been completed on a UR5e robot arm using kinaesthetic teaching. Kinaesthetic teaching uses waypoints to specify a path. The waypoints are calibrated by manually move the robot arm to the desired position and stored in the robot.

Linear and circular movements define the path. Linear movements require the start and end waypoints, whereas circular movements also require the curve's midpoint. Table 2 summarizes the type of segments as well as different waypoints. It takes the blue path in Figure 16 as an example.

Segments	Types of Segments	Waypoints
From point 1 to point 2	Straight line	Point 1, 2 (Start and end point)
From point 2 to point 4	Arc with midpoint of point 3	Point 2, 3, 4
From point 4 to point 5	Straight line	Point 5 is the junction

Table 2 Summary of segment properties.

The following figure displays the path of the channel (Figure 16). The blue and yellow lines are two examples of the path.

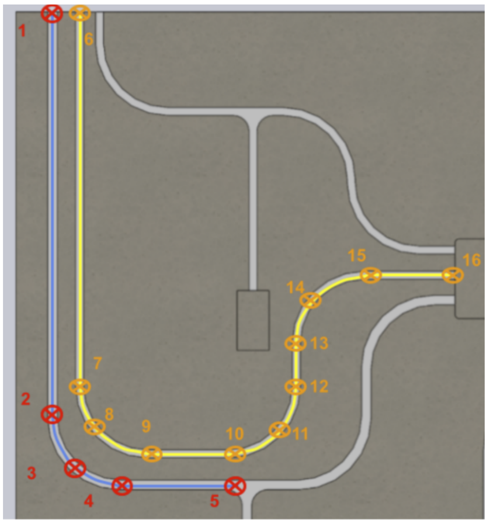


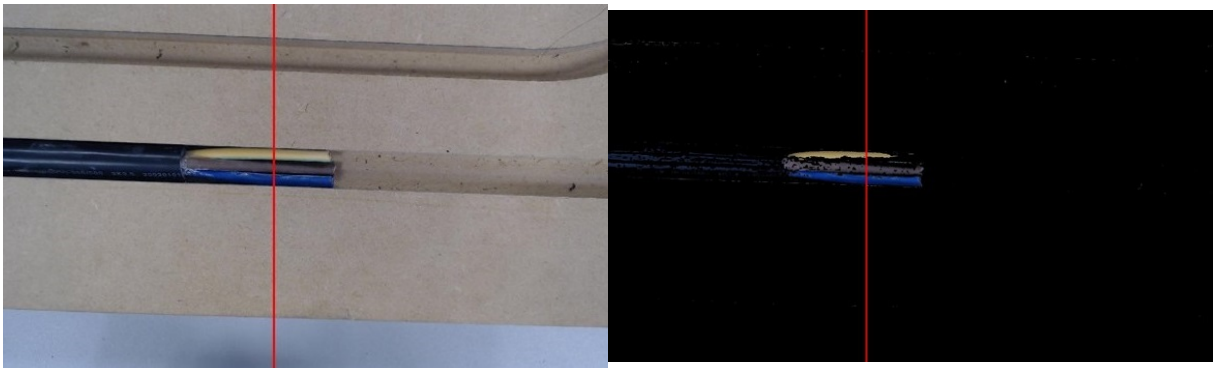
Figure 16 Paths of the channel (Waypoints 1 to 16) [3].

The aligning wheel is designed to be directly under the centre of the robot arm. The waypoint refers to the midpoint of the mounting flange of the robotic arm, which is also the starting point of the end-effector. Therefore, there will be no offset for the system.

5.3 Machine Vision at Junctions

The insulating layer of each core has a different colour. To place the connectors accurately, the system needs to distinguish the order of the insulating layers. This can be achieved by identifying the colour of the insulating layer. Since the whole process will be carried out in a factory, it is reasonable to assume that the lighting is similar every time it is installed. If there it is necessary to install the connectors in different environments, an LED can be added to the end effector. The camera will also be installed on the end effector to collect image information. If the camera position and zoom are fixed and accurate, each image can be divided into corresponding rows and columns. The positions of the rows and columns on each image are the same.

The first step is to identify the colours in the work identification line. The working identification line will be a vertical straight line. It will pass through all rows in a column (Figure 17a). The system needs to identify the line on the red line, that is, the colour of the core line (Figure 17b).



(a) Sample image: position of working identification line.
(b) Sample image: Identify colours with working identification line.

Figure 17 Sample of processing image.

The second step is to use MATLAB to process image information (Appendix A3). MATLAB function can divide the image into corresponding pixels. Each core will be arranged row by row. The row numbers are sorted from top to bottom. After that, the program will calculate the average index of rows where the colour is located. The average row index of each core insulation colour is shown in Figure 18. Then the function will output the corresponding colour string. For example, y represents yellow, n represents brown, and b represents blue. Therefore, the row of each colour can be determined. The process can work 100% of the time.

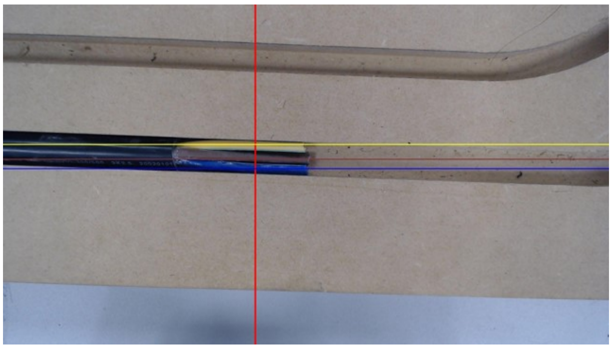
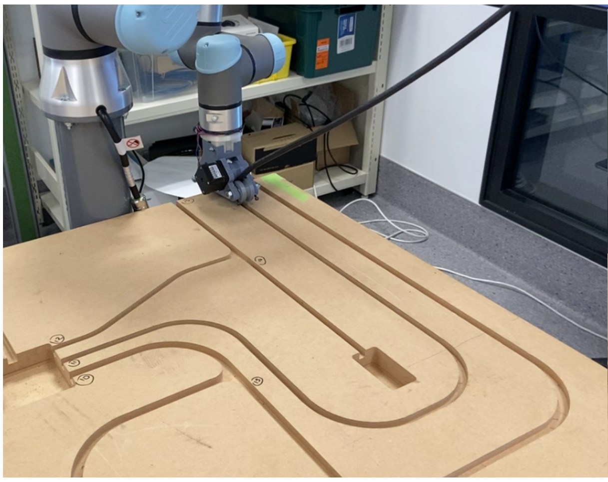


Figure 18 Sample Image used for Processing - determine the order of core colours.

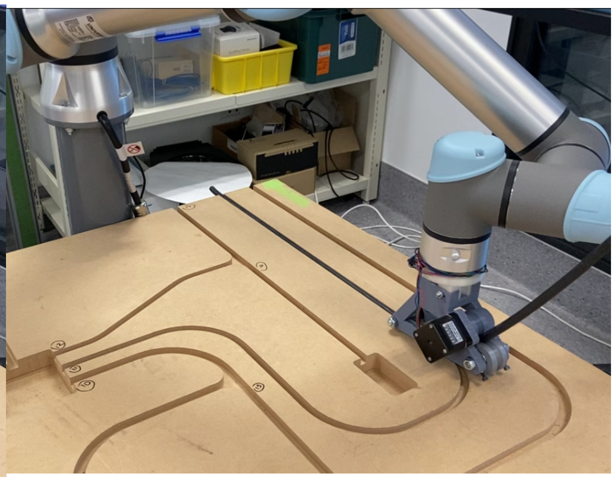
6. Results

6.1 Cable Routing

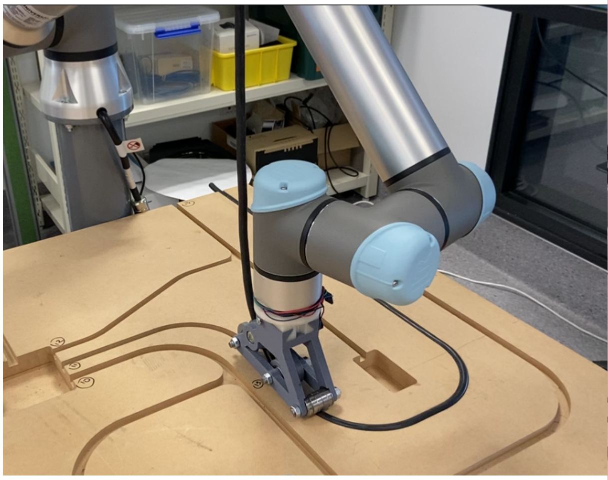
By experiments, the end-effector can successfully pass these waypoints and simultaneously push the cable into the channel (Figure 19). Figure 19a to d shows the process of routing, including the start point, the arc and the end of the path. Through kinesthetic teaching programming, its success rate reached 92% in the 12 initial test runs recorded before the laboratory was closed. The entire system can be used for 10mm stranded wires and 12mm solid core cables. However, the performance of twisted core cables is significantly better. Through experiments, it is also obvious that the speed of using a robotic arm for wiring will far exceed that of manual wiring. The path in Figure 19 will only take 16 seconds.



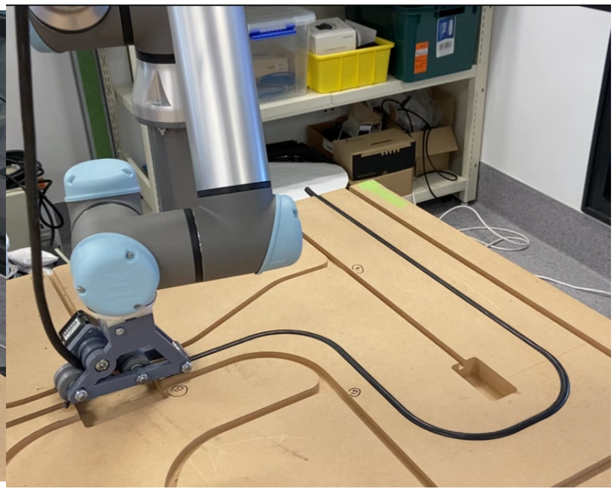
(a) Start point



(b) Start point of the arc.



(c) End point of the arc.



(d) End of the process.

Figure 19 Process of Cable Routing.

6.2 Overview of the System

The whole system consists of four parts: Cable pre-processing equipment, cable transportation, cable routing using end-effector and connector installation at junctions (Figure 20).

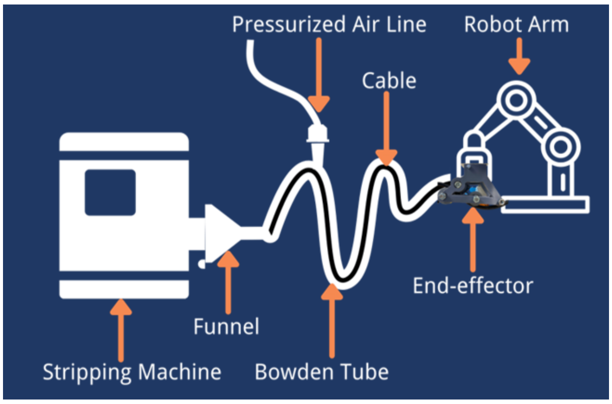


Figure 20 Conceptual diagram of the system.

The cable spool is to be connected to a wire stripping machine. The stripping machine will cut off the insulation of the outermost layer of the cable, leaving the core insulation. The stripping process is automatic, and the stripping machine can push the cable out of the device. After that, there will be a funnel to ensure the cable can be fed into the Bowden Tube.

Then a pressurized air line is inserted at the beginning of the tube. It will output pressurized gas to the tube. The pressurized air line can provide kinetic energy for the cables in the tube. It allows the cable to move quickly in the tube until the other end.

The cable is therefore fed into the end-effector. The Bowden tube will connect between the funnel and end-effector. Then the first robot arm is able to route the cable into prefabricated walls through kinesthetic teaching programming of the robot arm. The second robot arm is used to press the connector to the junction. Therefore, different cables can be connected to each other at corresponding connections.

7. Discussions

The results indicate that the cable can be routed into the pre-fabricated wall using a designed end-effector and robotic arm. The cable is pre-processed using a stripping machine and cable transportation equipment – a Bowden tube and pressurized air. Routing can be implemented in various kinds of paths within the wall.

The designed end-effector can solve the problem of gripping and manipulating flexible cables. The colour of the cable core can be identified by machine vision to locate the cable. In the actual industry, this research can reduce the labour costs required for routing. Since this study designed the process from a spool of cables to the final routing to the pre-fabricated wall, the entire process is automated.

7.1 Efficiency of Autonomous Cable Routing

The efficiency of autonomous routing is much higher than that of manual routing. Because it takes a lot of force to manually press the cable into the channel, the manual routing speed is very slow. In addition, the connector design proposed in this research also makes installation more efficient. The installation of the connector only requires another robot arm to press down, so the installation speed is very fast.

The end-effector designed in this study is suitable for 10mm and 12mm diameter cables.

This means that there is no need to replace the end-effector when routing different cables. This also simplifies the execution of routing and improves a certain degree of efficiency.

However, the path planning method proposed in this research requires kinesthetic teaching of the robotic arm in advance. This may slow down the initial setup of routing in the factory. Moreover, if the structure of the pre-fabricated wall changes, engineers need to re-kinesthetic teach the robotic arm.

Due to the lack of available data, the results cannot confirm that the designed end-effector is suitable for all types of robotic arms. If the robot arm used moves too fast or runs under too high power, although this can improve the efficiency of wiring, it may cause the end-effector to get stuck when turning. Therefore, this study does not recommend setting the power of the robotic arm too high.

7.2 Effect of Friction During Cable Transportation

In the process of cable transportation, the cable may have greater friction with the Bowden tube, which may cause the cable to block the movement in the tube. Especially if the shape of the cable is curved, the cable may generate a force opposite to the moving direction with the tube. This can be solved by increasing the pressure of the input air. However, the shape of the Bowden tube also needs to be as close to a straight line as possible.

7.3 Effect of the Environment of the Factory

This research assumed that the environment of the factory, especially lighting conditions, is identical and predictable. However, in reality, the lighting conditions of the factory are affected by the structure, location of the factory and weather. Therefore, the vision-based cable identification method used in connector installation at junctions proposed in this study may also be affected.

8. Conclusions

The designed end-effector demonstrated the realization of the autonomous routing system on pre-fabricated walls. It can handle, manipulate and route flexible cables. The results have shown that kinesthetic teaching of the robotic arm is suitable for routing of known pre-fabricated walls, providing a robotic based cable handling framework. IDC and IPC connectors are suitable for two types of cables: solid-core and stranded-core cables. The installation of connectors is based on machine vision which can locate the cable and understand the sequence of the inner cores of the cable. The connector can connect different cables at the corresponding junctions. The results also show that autonomous routing is more efficient than manual routing.

9. Suggestions For Future Works

One area for future research in this project is investigating the application of machine vision in automatic routing. The current path planning method is kinesthetic teaching. Therefore, every time a new wall structure is replaced, the robotic arm must be retrained. If machine vision can be used to identify the design of the wall, especially the route of the location of the channel, the system can be made more intelligent and suitable for more wall structures.

Future development may also improve the performance of the end-effector. The currently designed end-effector is not suitable for high-power robot arms, and it is prone to slipping when facing extremely sharp turns (U-turns greater than 90 degrees).

The cable transportation method is also an interesting point worth considering. The current transportation method is more suitable for short-distance transportation. If the stripping machine and robot arm are too far apart, Bowden tube and pressurized air transportation are not ideal. This is also linked to another proposal for future research: the cable selection system. The current cable requires manual selection. The system cannot identify whether the cable is 10mm or 12mm in diameter and cannot make a selection. This is especially obvious when multiple prefabricated walls are routed at the same time. Suppose the system can automatically distinguish and select different types of cables and identify the corresponding cable channel through the vision system. In that case, it can further save labour costs and reduce errors. This can also help build a system in which the stripping machine connects multiple end-effectors.

If the connector needs to be installed in a different factory environment, the current machine vision algorithm also needs to be improved. Therefore, the machine vision system can identify the cable core's colour and position under various lighting conditions.

References

[1] X. Li, X. Su, Y. Gao, and Y.-H. Liu, "Vision-based robotic grasping and manipulation of usb wires," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 3482–3487.

[2] D. Morrison, P. Corke, and J. Leitner, "Learning robust, real-time, reactive robotic grasping," *The International Journal of Robotics Research*, vol. 39, no. 2-3, pp. 183–201, 2020.

[3] "Zuru tech."

[4] Z. Yuan, C. Sun, and Y. Wang, "Design for manufacture and assembly-oriented parametric design of prefabricated buildings," *Automation in Construction*, vol. 88, pp. 13–22, 2018.

[5] J. Shintake, V. Cacucciolo, D. Floreano, and H. Shea, "Soft robotic grippers," *Advanced Materials*, vol. 30, no. 29, p. 1707035, 2018.

[6] A. Billard and D. Kragic, "Trends and challenges in robot manipulation," *Science*, vol. 364, no. 6446, 2019.

[7] H. Heidari, M. J. Pouria, S. Sharifi, and M. Karami, "Design and fabrication of robotic gripper for grasping in minimizing contact force," *Advances in Space Research*, vol. 61, no. 5, pp. 1359–1370, 2018.

[8] M. Saha and P. Isto, "Motion planning for robotic manipulation of deformable linear objects," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.* IEEE, 2006, pp. 2478–2484.

[9] J. Sanchez, J.-A. Corrales, B.-C. Bouzgarrou, and Y. Mezouar, "Robotic manipulation and sensing of deformable objects in domestic and industrial applications: a survey," *The International Journal of Robotics Research*, vol. 37, no. 7, pp. 688–716, 2018.

[10] K. Tabata, H. Seki, T. Tsuji, T. Hiramitsu, and M. Hikizu, "Dynamic manipulation of unknown string by robot arm: realizing momentary string shapes," *ROBOMECH Journal*, vol. 7, no. 1, pp. 1–17, 2020.

[11] Y. She, S. Wang, S. Dong, N. Sunil, A. Rodriguez, and E. Adelson, "Cable manipulation with a tactile-reactive gripper," *arXiv preprint arXiv:1910.02860*, 2019.

[12] A. Papacharalampopoulos, S. Makris, A. Bitzios, and G. Chryssolouris, "Prediction of cabling shape during robotic manipulation," *The International Journal of Advanced Manufacturing Technology*, vol. 82, no. 1-4, pp. 123–132, 2016.

[13] J. Zhu, B. Navarro, P. Fraisse, A. Crosnier, and A. Cherubini, "Dual-arm robotic manipulation of flexible cables," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 479–484.

[14] W. Yuan, S. Dong, and E. H. Adelson, "Gelsight: High-resolution robot tactile sensors for estimating geometry and force," *Sensors*, vol. 17, no. 12, p. 2762, 2017.

[15] S. D. RAVINDER, G. METTA, G. CANNATA, and M. VALLE, "Special issue on robotic sense of touch," *IEEE transactions on robotics*, vol. 27, no. 3, 2011.

[16] A. Saxena, J. Driemeyer, and A. Y. Ng, “Robotic grasping of novel objects using vision,” *The International Journal of Robotics Research*, vol. 27, no. 2, pp. 157–173, 2008.

[17] P. Chang and T. Padır, “Model-based manipulation of linear flexible objects with visual curvature feedback,” in *2020 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE, 2020, pp. 1406–1412.

[18] D. Vernon, “A system for robot manipulation of electrical wires using vision.” *Robotica*, vol. 8, no. 1, pp. 47–60, 1990.

[19] A. Papacharalampopoulos, P. Aivaliotis, and S. Makris, “Simulating robotic manipulation of cabling and interaction with surroundings,” *The International Journal of Advanced Manufacturing Technology*, vol. 96, no. 5, pp. 2183–2193, 2018.

[20] S. Jin, C. Wang, and M. Tomizuka, “Robust deformation model approximation for robotic cable manipulation,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 6586–6593.

[21] [Online]. Available: https://www.zierick.com/papers/IDC.pdf

Appendix A MATLAB codes

Program A1 Calculation of the required number of bearings

```
function [n] = PushRollerBearingCount(cable_r, axle_d, bearing_w)
% Finds the number of bearings needed for the push roller considering the
% radius of the cable to be layed, the horizontal distance between the
% two axles, and the width of the bearings. Use consistent units.

% Find the min radius of the channel curve
curve_r = 16 * cable_r;

% Find the number of bearings reuired
n = 2 / bearing_w * (curve_r - sqrt(curve_r ^ 2 - axle_d ^ 2));

% Round number up
n = ceil(n);

end
```

Program A2 Calculation of the maximum radius of the driven wheel

```
function [R_max] = MaxDriveWheelRadius(cable_r, curve_r, press_depth, press_ratio)
% Finds the max radius of the drive wheel considering the radius of the
% cable to be layed, the min radius of the channel curve, the min depth that
% the cable is to be pushed in (from middle of cable) and the min ratio of the
% total cable press force which is pushing the cable down. Use consistent
% units.

% Find the max angle for press ratio
press_angle = acos(press_ratio);

% Find the max deivation from press angle
centre_deviation = tan(press_angle) * (cable_r - press_depth);

% Find the max half-length of cable pushed in by wheel at once which can
% fit within deviation
l = sqrt(curve_r ^ 2 - (curve_r - centre_deviation) ^ 2);

% Find the max radius from half_length
R_max = ((press_depth - cable_r) ^ 2 + l ^ 2) / (2 * (cable_r - press_depth));

end
```

Program A3 Determine the colour of each row

```
function [pattern] = GetColourCode(image, column, cores)
% Description: Receives an image of an electrical cable and working axis
% to determine the pattern of the coloured core insulations. This version
% assumes that the image has been rotated until the cable if horizontal and
% a column of coloured cores has been specified. This is to imitate the
```

```
% type of accuracy that would been observed in an industrial process
% Inputs:  image - RGB image matrix
%          column - the column number to get the colour pattern from
%          cores - number of cores in cable
% Outputs: pattern - array of colour characters, top-to-bottom (b = blue /
%                    g = grey / k = black / n = brown / y = yellow-green )
% NOTE: 3-core and 4-core cables have "bny" and "gkny" colours respectively
% Author: Zach Macleod

%% Initialisation
% Set colour boundaries (in HSV colour model)
% b(x) = (lower limit for hue, upper limit for hue;
%                lower limit for saturation, upper limit for saturation ;
%                lower limit for value, upper limit for value)
b = zeros(5, 3, 2);
b(1, :, :) = [0.56, 0.7; 0.5, 1; 0.2, 0.7]; % Blue
b(2, :, :) = [0, 1; 0, 0.03; 0.2, 0.4]; % Grey
b(3, :, :) = [0, 1; 0, 1; 0, 0.2]; % Black
b(4, :, :) = [0.6501, 0.8; 0.1, 0.5; 0.15, 0.6]; % Brown - Dependent on
% light conditions
%b(4, :, :) = [0.6501, 1.08; 0.1, 0.5; 0.15, 0.6]; % Brown - Dependent on
% light conditions
b(5, :, :) = [0.08, 0.55; 0.3, 1; 0.3, 0.8]; % Yellow-Green

% Initialise the coloured pixel and cumulative row count
pixels = zeros(1, 5);
n = zeros(1, 5);

% Initialising size of image
[rows, ~, ~] = size(image);

%% Processing
% Convert image to HSV colour model
image = rgb2hsv(image);

% For each pixel, check if it fits inside the boundaries of each colour. If
% so, add to the pixel count and the cumulative row count of that
% colour. If the upper limit of hue is above 1, add 1 to the pixel hue
% (HSV colour model uses a cylcic hue so 0 and 1 are the same - red).
for r = 1:rows
    for c = column-1:column+1
        for a = 1:5
            if ((b(a, 1, 2) >= 1) + image(r, c, 1) >= b(a, 1, 1) && (b(a, 1, 2) >= 1) + image(r, c, 1) <= b(a, 1, 2)) && image(r, c, 2) >= b(a, 2, 1) && image(r, c, 2) <= b(a, 2, 2) && image(r, c, 3) >= b(a, 3, 1) && image(r, c, 3) <= b(a, 3, 2)
                n(a) = n(a) + r;
                pixels(a) = pixels(a) + 1;
            end
        end
    end
end

```

```
%% Output
% Take the relevant colours for 3/4 cores
if cores == 3
    n = n([1, 4, 5]);
    pixels = pixels([1, 4, 5]);
    pattern = 'bny';
elseif cores == 4
    n = n([2, 3, 4, 5]);
    pixels = pixels([2, 3, 4, 5]);
    pattern = 'gkny';
end

% Divide each colour's cumulative row by it's pixel count to find
% the mean row
n = n ./ pixels;


% Sort in ascending order
[n, order] = sort(n);

% Rearrange pattern for output
pattern = pattern(order);

%% Show Image
% Convert the image back to the RGB colour model
image = hsv2rgb(image);

% Add the working column as a line of red
image(:, column - 2: column + 2, 1) = 1;
image(:, column - 2: column + 2, 2) = 0;
image(:, column - 2: column + 2, 3) = 0;

disp(n);

% Add the mean rows to the image as relevant colours
for k = 1:length(pattern)
    try
        if pattern(k) == 'b'
            image(ceil(n(k))-1:ceil(n(k))+1, :, 1) = 0;
            image(ceil(n(k))-1:ceil(n(k))+1, :, 2) = 0;
            image(ceil(n(k))-1:ceil(n(k))+1, :, 3) = 1;
        elseif pattern(k) == 'g'
            image(ceil(n(k))-1:ceil(n(k))+1, :, 1) = 0.3;
            image(ceil(n(k))-1:ceil(n(k))+1, :, 2) = 0.3;
            image(ceil(n(k))-1:ceil(n(k))+1, :, 3) = 0.3;
        elseif pattern(k) == 'k'
            image(ceil(n(k))-1:ceil(n(k))+1, :, 1) = 0;
            image(ceil(n(k))-1:ceil(n(k))+1, :, 2) = 0;
            image(ceil(n(k))-1:ceil(n(k))+1, :, 3) = 0;
        elseif pattern(k) == 'n'
            image(ceil(n(k))-1:ceil(n(k))+1, :, 1) = 0.5;
            image(ceil(n(k))-1:ceil(n(k))+1, :, 2) = 0.2;
            image(ceil(n(k))-1:ceil(n(k))+1, :, 3) = 0.2;
        elseif pattern(k) == 'y'
            image(ceil(n(k))-1:ceil(n(k))+1, :, 1) = 1;
            image(ceil(n(k))-1:ceil(n(k))+1, :, 2) = 1;
            image(ceil(n(k))-1:ceil(n(k))+1, :, 3) = 0;
        end
    catch
        disp(['No pixels of colour ''', pattern(k), ''' were found']);
```

```
    end
end

% Show image
imshow(image);

imwrite(image, 'processed.jpg');
```