Abstract

Multi-robot motion coordination has different applications. This project mainly focuses in Pattern Formation. After user input a pattern, the robot team will move to specific locations and form the pattern. Arduino micro-processor and nRF wireless module will be used. In this report, I will describe the objective, methodology, result and the future work for the project.
Table of content

1. Project Background---------------------------------------------------------------3
2. Project Objectives---------------------------------------------------------------5
   2.1 Centralized Pattern Formation-----------------------------------------------5
   2.2 Robots Building-------------------------------------------------------------6
   2.3 Coordination-----------------------------------------------------------------7
   2.4 Collision Avoidance----------------------------------------------------------7
3. Methodology---------------------------------------------------------------------8
   3.1 Tasks of Robots--------------------------------------------------------------8
   3.2 Robots Type------------------------------------------------------------------9
   3.3 Arduino Board---------------------------------------------------------------10
   3.4 Wireless Communication-----------------------------------------------------11
   3.5 Approach of Pattern Formation----------------------------------------------12
   3.6 Robot Detection-------------------------------------------------------------13
   3.7 Location generation algorithm---------------------------------------------15
   3.8 Matching algorithm----------------------------------------------------------17
4. Result-------------------------------------------------------------------------18
   4.1 Robots Building--------------------------------------------------------------18
   4.2 Controller Building----------------------------------------------------------21
   4.3 Wireless Communication-----------------------------------------------------23
   4.4 Computer Program Algorithm-------------------------------------------------24
      4.4.1 Robot Detecting----------------------------------------------------------25
      4.4.2 Destinations Generation-------------------------------------------------27
      4.4.3 Robot-Destination Matching---------------------------------------------29
      4.4.4 Robot Movement Calculation---------------------------------------------30
      4.4.5 Collision Avoidance-----------------------------------------------------32
      4.4.6 Deadlock Detection------------------------------------------------------33
   4.5 Computer Program UI---------------------------------------------------------35
   4.6 Hardware Setup---------------------------------------------------------------38
5. Future Work---------------------------------------------------------------------41
   5.1. Robot Design Improvement-----------------------------------------------------41
   5.2. Algorithm Optimization-------------------------------------------------------41
   5.3. Considering Decentralized Approach------------------------------------------42
6. List of Figures------------------------------------------------------------------43
7. List of Tables-------------------------------------------------------------------44
8. References----------------------------------------------------------------------45
9. Appendices----------------------------------------------------------------------47
1. Project Background

The development of robot system is progressive. Robot system is now more common in different occasions. We can see robots doing jobs like exploration, rescue and manufacturing. It helps human being in different areas, especially to replace human to perform dangerous, boring or repetitive jobs that people do not prefer.

According to Robot Institute of America, a robot is a reprogrammable, multifunctional manipulator designed to perform a variety of tasks. [1] However, most of the robots we see in daily life are relatively simple which perform only easy and repetitive tasks. They usually are just a still machine that perform one specified task.

And robots that perform complicated tasks often cost a lot that normal people cannot afford. They are usually owned by organizations that are specific for military or research use. They can perform various and complex tasks, but general public can seldom benefit from them.

There seems to be an inversely proportional relationship between robot’s performance and robot’s cost. What if low-cost robots can perform advanced tasks? One robot may not be possible, but a bunch of them could be. This is what this project aims to do – use multiple low cost robots to coordinate together and perform some relatively complex tasks.
In this project, multiple robots are built and they are able to achieve pattern formation. When users provide a pattern/shape to the computer program, then the robots can move to specific locations and construct the pattern.

The remaining part of this document is structured as follow. Section 2 describes the objective of this project. Section 3 states the methodology and how the project is to be implemented. Section 4 lists the result of the project. Section 5 is the future work.
2. **Project Objectives**

The goal of this project is to enable multiple robots to coordinate together to perform specific tasks. The following are the proposed tasks that should be achieved.

2.1 **Centralized Pattern Formation**

This is the main objective of this project. The robots should be able to form different shapes or patterns with their “bodies”. After they receive the order - a shape or pattern that provided by the user, the robots should travel to suitable location and form that shape or pattern altogether. So, if the robots are viewed from the top, the shape/pattern would be merely formed. More robots take part in the formation, more accurate the shape/pattern is.

To achieve this, centralized pattern formation is chosen. A central unit monitors the whole area and all robots [5]. The movement and location of the robots are transmitted from robots to the central unit. And the movement order would be transmitted from the monitoring device to the robots. Then multiple robots would perform tasks accordingly.

Therefore, the main computing place is the central unit which contains a camera and transceiver. It would do most of the computation and send out the commands to the robots [5, 6].
The robots would do relatively less computations and focus on receive and execute the commands from the monitoring device. So, the processing power requirement of the robots is lower and thus lower cost. However, more investment is needed in building the central unit.

2.2 Robots Building

There will be several robots in this project. Approximately 20 robots will be built. All robots will adapt the same design. They are basically combined with 3 components: a microcontroller, a mechanical unit and a communication unit [2].

The microcontroller works like a brain for the robot. Arduino board is used because of its extendibility. The microcontroller processes all the input and output data. The mechanical unit contains motors and wheels. They are connected to the microcontroller then receive and execute order from the microcontroller. The communication unit is a wireless module. It is responsible for the receiving and transmitting data. Therefore, the robots able to communicate with others.

The shape of the robots would be circular because their function is pattern formation. A circular shape is more suitable. All robots would be tested and ensured to have the same capability.
2.3 **Coordination**

The multiple robots should work with each other. The coordination is achieved by setting up a central unit. The central unit do most of the computational job.

The positions of the robots are monitored by the central unit. It is expected to contain a camera to identify and locate the robots [3]. By analyzing the current location, the next movement of the robots will be computed. After that, the commands are sent to the robots wirelessly. During the execution, the positions are updated and the central unit keep updating and sending the command. It keeps computing until the goal is achieved.

2.4 **Collision Avoidance**

This is the fundamental requirement for every robot. They should move around the specified area without colliding other robots or the obstacles. Only by doing so, further tasks can be processed smoothly.

To achieve this, the central unit mentioned above is responsible. While it is determining the next movement, it also monitors robots’ position and avoid collision [4]. If the robots are too close to each other and about to collide, the central unit would change the route for the robots.
3. Methodology

3.1 Tasks of Robots

Before the hardware and software details of robots are confirmed, the tasks of the robots are decided because this may affect the design of the robots.

“Multi-robot motion coordination” is searched on Google Scholar. There are total 20700 results and top 5 results are examined. “Localizing”, “mapping”, “exploration”, “path planning” are appeared in one or more articles. However, these projects require much resources and are not very practicable for a one-man-team.

More research has been done and some inspiring projects are found. Including “Kilobot” from Harvard University [7, 8, 9], “Multi-Robot System for Artistic Pattern Formation” from Disney Research [3, 10] and “Multi-Robots Formation Control System” by Keisuke Uto [11].

Considering the workload and the effect of the project, pattern formation is chosen for the main goal of the multi robot system. Users of the system can provide the pattern; the robots then move and form it. This should be a good experience for users to know about this project.

The project also has relatively good extendibility, once the technique of pattern formation is acquired, further development can be done based on this.
3.2 Robot Type

As in the similar projects, some robots used are already on sale and ready-to-use without assembling such as “Kilobot” and “E-puck”. However, the price of these robots is high, around HKD$10,000 for 10 Kilobots [9] and HKD$19,000 for 3 E-pucks [3]. As these robots include many components for different uses which is not necessary in this project, they are not suitable.

By further research, many projects done by others used Arduino as the main component of the robot. It is a microcontroller with input and output pins, different components can be bought separately for specific use [12]. Also, since both the hardware and software is open source, there are replicates that being sold in relatively lower price. After estimation, a robot can be built with the cost under HKD$100. This is more practicable for the project.

The robot is built based on an Arduino board, with few more components added onto it.
3.3 Arduino Board

After deciding the robots would be built based on Arduino, the model of Arduino board need to be confirmed. There are 12 board still on sale and 10 already retired [13]. The main difference of the boards are the size, processor, speed and number of I/O pins.

Some common boards are listed and compared in the Table 1.

<table>
<thead>
<tr>
<th>Name</th>
<th>Processor</th>
<th>CPU Speed</th>
<th>Analog I/O</th>
<th>Digital IO/PWM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mega 2560</td>
<td>ATmega2560</td>
<td>16 MHz</td>
<td>16/0</td>
<td>54/15</td>
</tr>
<tr>
<td>Micro</td>
<td>ATmega32U4</td>
<td>16 MHz</td>
<td>12/0</td>
<td>20/7</td>
</tr>
<tr>
<td>Uno</td>
<td>ATmega328P</td>
<td>16 MHz</td>
<td>6/0</td>
<td>14/6</td>
</tr>
<tr>
<td>Nano</td>
<td>ATmega168 ATmega328P</td>
<td>16 MHz</td>
<td>8/0</td>
<td>14/6</td>
</tr>
</tbody>
</table>

Table 1. Comparison of common Arduino boards

For this project, the optimal robots are form factors that are powerful enough for the tasks and can be built with lowest cost and smallest size. Since the task of the robots is to receive command from central unit and move the motors accordingly which is relatively not computational complex, the processor and CPU speed should not be a limitation. The main concern is the size and the number of I/O pin.

After examining the specifications of every Arduino boards, Arduino UNO and Arduino Nano are shortlisted. They have similar specification, while Arduino Nano has a much smaller size, but it is retired so no official support for it. In the project, Arduino UNO is used because of its easiness of installation and better support.
3.4 Wireless Communication

As the robots are required to communicate wirelessly with the central unit, each robot needs a wireless module to transmit and receive the wireless signal. There are several wireless modules that compatible with Arduino development environment, including Infra-Red, Wi-Fi, Bluetooth and Radio frequency.

For centralized pattern formation, the central unit sends out commands of every robots and each robot receives its own command. Therefore, each robot should be able to filter out and execute its own command but not the commands of other robots. To achieve this, the signal sent out from the central unit need to be multi-channel [3]. Therefore, Infra-red is not compatible. Also, the price of Wi-Fi module is relatively high. Bluetooth and Radio frequency module are shortlisted. Both Bluetooth and Radio frequency module are bought and tested.

In the end, radio frequency will be used because no pairing is needed and it allows broadcasting of signal.
3.5 Approach of Pattern Formation

Centralized pattern formation is chosen for this project. There is another approach – decentralized pattern formation, which does not need a central computational unit.

Both methods were considered before. However, decentralized approach means there is no central unit to monitor the position of the robots [4]. Thus, each robot needs the global information of all other robots. This requires bidirectional communication between robots. The requirement of robots would be stricter and rising the cost of each robot. So, the decentralized approach is not chosen.
3.6 Robot Detection

The computer system needs to detect the robots’ locations. Different methods have been tried for this purpose.

At first, colour detection was used. The camera recognises different colours by their HSV values. However, this method very much depends on the environment. The lighting would affect the HSV values. Therefore, the system need to be calibrated every time in a different place. This takes a long time and the result is not accurate.

While using this method, a coloured paper is attached on the top of each robot. Then calibrate the system for each colour. As there are multiple robots, the process has to be done multiple times.

Figure 1. Robot detection by colour recognition
Another problem is that for each robot, a colour is assigned to it. For a small number of robots, it is okay. But for large number of robots, it is difficult to find so many different colours for every robot and the colours are not similar in order to avoid detection mistake. There are multiple problem in using colour detection. Therefore, this method was abandoned.

Figure 2. Robot detection by contour detecting

Then, contour detection is used. This time the system detection the contours under the camera and recognize the shape and location of each pattern.

Please refer to section 4.4.1 for more details on the result.
3.7 Location generation algorithm

To form patterns with the robots, robots need to go to some specific destinations. Then all robots can form a specific pattern as a whole.

In the early stage of the project, it was planned to adapt Voronoi Tessellation [14] to divide a pattern. Then the robots move to specific positions to form the pattern. However, the algorithm requires robots to constantly move to different directions. Due to the limitation of the robots, it will take a large amount of time. Also, this project contains small number of robots. The effect of pattern formation may not be obvious with this method. Therefore, Voronoi Tessellation is abandoned.

A simpler but effective method is adapted. That is to detect the contour of the pattern to be formed. Then the contour is divided into few parts with equal length. Each robot will go to a position that divides the contour. All robots will then form the contour of the pattern and show it more obviously.

This method does not require robots to constantly change their goals. The destinations are calculated at the beginning. All robots just go to their assigned destinations, and the pattern can be formed.

Please refer to section 4.4.2 for more details on the result.
3.8 Matching algorithm

As mentioned in 3.7, there are few destination ordinations are calculated at the beginning. The number of destinations is the same as the number of robots on screen. Therefore, each robot should be paired with one destination. The robot-and-destination pair should be efficient that the robots take shorter distance to go there. Thus, the pattern formation process can be done fast.

In the early stage of this project, the system scan through all robots and assign the nearest destination for them. However, since the system assign the destinations to robots one by one, the last robot sometimes will be assigned to a far destination which take the last robot longer time to get to the destination.

![Image](image.png)

**Figure 3. Last robot in the queue assigned to a far destination**

The system scan the robot from left to right. Therefore, the last robot in the queue is usually the one on the right. As shown on the photo above, the top right robot is assigned to the far position on the left, which is clearly not the best matching result.
Later “Stable marriage problem” [15] was found during research, it is about matching a bunch of men and women, like the case in this project, not men and women, but robots and destinations.

And the solution is Gale–Shapley algorithm [15]. Basically, to solve the problem, all people (men and women) have a preference list about his/her target. At the beginning, all men purpose to their first choice. So, some women received multiple purposes, some receive none. For women that receive multiple purposes, she chooses the one with higher preference and reject others. So, some men maybe accepted, some maybe rejected. The men got rejected purpose to his next choice. So on and so forth. Men keep lowering their standard, women keep finding better one. At the end, all will be paired.

Please refer to section 4.4.3 for more details on the result.
4. Result

4.1 Robots Building

Total 7 robots have been built. All of them have the same design and components. Each robot costs around HKD$70.

Each robot contains the following components:

<table>
<thead>
<tr>
<th>Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arduino UNO board</td>
</tr>
<tr>
<td>L298N Motor Driver</td>
</tr>
<tr>
<td>NRF24L01 NRF Wireless Module</td>
</tr>
<tr>
<td>Plastic Chassis</td>
</tr>
<tr>
<td>Motors + Wheels</td>
</tr>
<tr>
<td>9V DC batteries</td>
</tr>
</tbody>
</table>

Table 2. Components of a robot

The robots are in a circular shape with diameter around 14cm and 9cm tall. Each robot as two wheels on two side of the robot, each wheel can rotate forward and backward to achieve different movements of the robot. There are four movements of a robot, each requires the cooperation of two wheels.
<table>
<thead>
<tr>
<th>Movement</th>
<th>Left Wheel</th>
<th>Right Wheel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Move Forward</td>
<td>Forward</td>
<td>Forward</td>
</tr>
<tr>
<td>Move Backward</td>
<td>Backward</td>
<td>Backward</td>
</tr>
<tr>
<td>Rotate Left</td>
<td>Backward</td>
<td>Forward</td>
</tr>
<tr>
<td>Rotate Right</td>
<td>Forward</td>
<td>Backward</td>
</tr>
</tbody>
</table>

Table 3. Relationship between robots’ movement and rotation of two wheels

![Figure 4. Final robot design](image)

All robots’ Arduino board are loaded with the same code, except the identity number. Each robot has a unique identity number for system to recognise their identity.
As mentioned in 3.7, on the top of each robot, there is a printed paper attached. It indicates the identity number and the orientation of the robot.
4.2 Controller Building

To control the roots mentioned in 4.1, a controller is required to send command to the robots and control their movement. The controller basically convert the commands from the computer to radio frequency, then broadcast it to the robots.

The basic components of the controller are similar to the robots, but since it only does one job – sending commands, the structure is simpler.

The controller contains the following components:

<table>
<thead>
<tr>
<th>Components</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arduino UNO board</td>
</tr>
<tr>
<td>NRF24L01 NRF Wireless Module</td>
</tr>
<tr>
<td>Plastic Chassis</td>
</tr>
</tbody>
</table>

Table 4. Components of controller

Figure 6. Design of controller
It is connected to the computer to get power and data. The computer will calculate what commands to be sent and transfer the command to the controller via serial communication. Then the controller broadcast it with NRF.
4.3 Wireless Communication

Wireless communication is needed between the controller and the robots. NRF is used in this project. On both controller and robots, there is a NRF24L01 NRF Wireless Module. The module on the controller is for sending commands. The one on robots is for receiving commands.

The commands to control robots are in the following format:

\[ X0Y \]

X is an integer that indicate the robot it is controlling.

Y is an integer that indicate the movement of the robot.

There are 4 types of movement for robots:

1: Move forward

2: Move backward

3: Rotate left

4: Rotate right

For example, the command controlling robot #1 to move forward is \[101\], command controlling robot #5 to rotate left is \[503\]

The controller will send out multiple commands, all robots are able to receive all commands. When a robot received a command, it will check whether the command is for it or not. If it is, the robot executes it, otherwise just ignore it.
4.4 Computer Program Algorithm

As a centralized approach, most of the computation is done in the “central device” which is the laptop in this case. A program is wrote based on C++ to be responsible for most computation and display the UI for users.

The program does the computation about the algorithm implemented, including robot detection, destination calculating, robot-destination matching, collision avoidance etc.

Various algorithms are implemented for different uses. One of the most important part of this project is robot detection. OpenCV library is included. It is a library of function mainly focus in real-time computer vision. Multiple functions are used in the computer program to detect the contours of a shape. Therefore, robots can be detected, the identity number can be recognised and the destinations can be calculated.
4.4.1 Robot Detection

To detect and distinguish different robots, a paper with patterns printed is attached on each robot.

An isosceles triangle pattern represents a robot. The system detects the contours and realize it is a triangle. Then calculate which direction the triangle is pointing to. By doing this, the location and orientation of the robot can be detected by the system.

Other than the location and the orientation of the robots, the identity number of each robot are required too. So I added some pattern on each robot inside the triangle.

![Pattern attached on robot for robot detection](image)

**Figure 7. Pattern attached on robot for robot detection**

Few white circles are added into the triangle. There will be maximum 4 circles. Each circle represents one bit, so 4 circle can at most represent $2^4 = 16$ numbers, from 0000 to 1111 in binary or 0 to 15 in decimal.
Therefore, maximum 16 robots can be supported for 4 circles. If the system is scaled up and more robots are added, just add the maximum number of circles. For example, if the system has 100 robots, there should be maximum 7 circles. Therefore, there are 7 bits and can support $2^7 = 128$ robots for maximum.

Contour detection is better than colour detection because it is based on black and white image. The environment lighting does not affect much. No calibration is needed for every time setup.

Also, the system using contour detection is easily scalable. Adding circle to the robot pattern and a little adjustment on the software part can enable system extension.

![Figure 8. Robot detection result](image)
4.4.2 Destinations Generation

To calculate the destinations of the robots, the length of the pattern contour is considered.

The system detects the contour of the pattern to be formed. Depending on how many robots are under the camera, the contour is divided into few parts with equal length. Each robot will go to a position that divides the contour. All robots will then form the contour of the pattern.

Robots do not need to constantly change their goals. The destinations are calculated at the beginning. All robots just go to their assigned destinations, and the pattern can be formed.

The photos below show the result of destination generation using contour dividing.

![Figure 9. Position generation result with 2 robots](image)
As shown in the photo above, the algorithm calculates robots’ destinations according to the number of robots. When there are 2 robots, the pattern is divided into two parts; When there are 5 robots, the pattern is divided into five parts. Robots’ destinations depend on the number of robots on screen.
4.4.3 Robot-Destination Matching

To match the robots and their destinations, Gale–Shapley algorithm is used to solve the “Stable Marriage Problem”.

To implement this algorithm, I substitute men and women with robots and destinations. Their preferences are based on the distance. For example, robot A scans through all destinations, and list its preferences with shortest distance first, longest distance last. So as the destinations, they list all robots according to their distance. Then the robots “purpose” to the destinations, and finally everyone got paired.

![Figure 11. Robots-Destinations matching using Gale–Shapley algorithm](image)

As shown in Figure 11, all robots and destinations have a preference list displayed in red text. After running the algorithm, all robots and destinations are paired after the “purpose” process.
4.4.4 Robot Movement Calculation

To calculate the movement of a robot, the orientation of the robot and the direction of the path are compared.

![Figure 12. Orientation and path direction of a robot](image)

The orientation of robot is determined by the direction of the triangle pattern on the robot. On the photo above, a green arrow points from the centre of the triangle to the “head” of the triangle. This arrow indicates the orientation of the robot.

Similarly, to find the direction of the path, a blue arrow points from the centre of the robot to the robot’s destination. The direction is then calculated.
To calculate the robot’s movement, the orientation of two arrows are compared. If the green arrow is on the left of the blue arrow, the robot turns right. If the green arrow is on the right of the blue arrow, the robot turns left. If the orientations of both arrows are the same, the robot move forward.
4.4.5 Collision Avoidance

To avoid collision, robots’ paths were examined. The system checks if there is any robot blocking other robot’s path.

As shown in Figure 14, robot the green lines represent the path of robots. Robot 4 is blocking robot 3’s path. If robot 3 keep moving, collision will occur.

To avoid this, robot 3 will pause its movement, until its path is cleared i.e. robot 4 goes out of its way. Therefore, if robot A is blocking robot B, robot A will continue moving and robot B will pause until robot A goes away.

Figure 14. Robot’s path blocked by another robot
4.4.6 Deadlock Detection

As mentioned in 4.4.5, robot being blocked will pause and stop moving. There may be chance that all robots are blocking each other’s way and deadlock occurs.

![Figure 15. Deadlock on collision avoidance](image)

To solve the deadlock problem, the system first check if there is deadlock after every round of robot move. If deadlock occurs, the system will control the robot to spread out.
Figure 16. Robots spread out after deadlock occurs

It will start from the outermost robot, ensuring it won’t collide with other robots.

After all robots spread out, they aim to their original destinations again. Since, they are spread, they should be able to go to their destinations without deadlock.
4.5 Computer Program UI

As mentioned in 4.4, the C++ based computer program do most of the computation. At the same time, it serves as an input and output interface for users. It outputs information of the system and allows users to input.

![Computer Program UI based on C++](image)

**Figure 17. Computer Program UI based on C++**

The UI of the program mainly consists of 3 parts. The main display, system information and pattern selector.

The main display takes the most area of the UI. It shows the image captured by the camera i.e. the robots. Robots’ information will be shown here too, including the orientation of the robots, the identity of the robots and the path robots will go to.
Figure 18. Robot’s information shown in computer program

As the photo shown above, the red triangle is what the program treat as a robot. The red number is the identity number of the robot. There are 2 arrows pointing form the centre of the robot. The green arrow is the orientation of the robot; the blue one is the path of the robot. This information is shown to allow user to understand what the program is computing and its result.

Figure 19. System information in the computer program UI

At the top left corner, some system information is shown. The information includes the pattern the robots are forming, the number of robots on screen and number of robots that arrived its destination. The information allows users to understand the overall situation of the current moment.
At the bottom left corner, there is an area for user to choose the pattern robots will form. Just click on the pattern and the program will automatically calculate the new destinations for robots.

The computer program is the main input and output interface between user and the system. User can see the process and result of the pattern formation. Also user can manipulate the pattern and see the change of robots’ movement.

Figure 20. Pattern selector in the computer program UI
4.6 Hardware Setup

To demonstrate this project, all components are put together. The moving area of the robots is about 1m x 1 m.

All materials include:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Robots</td>
<td>Controller</td>
<td>Laptop</td>
</tr>
<tr>
<td>Webcam</td>
<td>iPad</td>
<td>Tripod</td>
</tr>
</tbody>
</table>

Table 5. List of materials needed for the whole system

Figure 21. Set up for the demonstrating the project
As mentioned before, the robots are for forming pattern, the controller is for sending commands, the laptop is for doing computation.

More hardware is needed. The webcam is for capturing image, mainly for robot detection. A tripod is used to hold the webcam in place. The webcam is about 1m above the ground, in order to cover the 1m x 1m robot moving area. The webcam model used is Logitech C920 with resolution of 1080p. It captures the robots from the top and detect every robots’ position.

**Figure 22. Webcam used in the project**

**Figure 23. Image capture by the webcam**
Also, an iPad is connected to laptop as an external monitor. Therefore, the computer program UI can be displayed on iPad. User can then use the touch screen to select the pattern, without a mouse. And the main screen of the laptop can be free and display some debug information.
5 Future Work

5.1 Robot Design Improvement

In order to reduce the time of assembly robots, a premade plastic chassis is used to mount all component. This is the smallest affordable chassis in the market but it is still too big. The robot does not contain many components and does not require such a big chassis.

If time allowed, a new chassis can be designed. The new robot can have a more compact body. Therefore, smaller robots can form more detailed pattern. This is one of the improvement can be done if time and budget are allowed.

5.2 Algorithm Optimization

There are rooms for improvement in the algorithm. More testing can be done on destination generating, matching, collision avoidance and deadlock handling.

The implementation can be more efficient and make the running time faster. The implementation of various algorithms can be reviewed and maybe revised if time allowed.
5.3 Considering Decentralized Approach

Now the project is adapting centralized pattern formation. This can lower the cost of each robot. The computation power of robot is not high too. However, this approach relies on the camera and central unit (the laptop in this case) very much. A camera has to be set up every time running the demo. It takes some time to set up and calibrate the position of the camera.

To achieve decentralized pattern formation, more sensors are required for the robots. And the computation work relies more on the robots themselves. Then no camera is required for the system. This may be a better method since it is not easy to set up a camera everywhere.
6 List of Figures

Figure 1. Robot detection by colour recognition------------------------P.13

Figure 2. Robot detection by contour detecting------------------------P.14

Figure 3. Last robot in the queue assigned to a far destination--------P.16

Figure 4. Final robot design------------------------------------------P.19

Figure 5. Final robot design with paper cover-------------------------P.20

Figure 6. Design of controller----------------------------------------P.21

Figure 7. Pattern attached on robot for robot detection--------------P.25

Figure 8. Robot detection result--------------------------------------P.26

Figure 9. Position generation result with 2 robots-------------------P.27

Figure 10. Position generation result with 5 robots------------------P.28

Figure 11. Robots-Destinations matching using Gale–Shapley algorithm----P.29

Figure 12. Orientation and path direction of a robot------------------P.30

Figure 13. 3 Cases of robot’s movement-------------------------------P.31

Figure 14. Robot’s path blocked by another robot--------------------P.32

Figure 15. Deadlock on collision avoidance---------------------------P.33

Figure 16. Robots spread out after deadlock occurs-------------------P.34

Figure 17. Computer Program UI based on C++------------------------P.35

Figure 18. Robot’s information shown in computer program------------P.36

Figure 19. System information in the computer program UI----------P.36
7 List of Tables

Table 1. Comparison of common Arduino boards-----------------------------P.10

Table 2. Components of a robot-------------------------------------------P.18

Table 3. Relationship between robots’ movement and rotation of two wheels------P.19

Table 4. Components of controller----------------------------------------P.21

Table 5. List of materials needed for the whole system------------------P.38
8 References


## 9 Appendices

### 9.1 Materials purchased in October 2016

<table>
<thead>
<tr>
<th>序号</th>
<th>图片</th>
<th>宝贝名称</th>
<th>数量</th>
<th>规格属性</th>
<th>商家编码</th>
<th>单价</th>
<th>金额</th>
<th>优惠</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>智能小车底盘 机器人 寻迹 小车 遥控小车 带磁盘 强磁 电机 H4</td>
<td>2</td>
<td>双层小车底盘</td>
<td></td>
<td>22.95</td>
<td>44.90</td>
<td>1.00</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>2014最新款UNO R3 开发板 送线 行家版本 GCC</td>
<td>3</td>
<td>H42</td>
<td></td>
<td>18.00</td>
<td>53.00</td>
<td>1.00</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>ProtoShield机器人小型扩展板 含mini面包板学习开发板</td>
<td>1</td>
<td></td>
<td></td>
<td>4.60</td>
<td>4.10</td>
<td>0.50</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>L293D motor control shield 电机驱动扩展板</td>
<td>2</td>
<td>H44</td>
<td></td>
<td>7.60</td>
<td>15.00</td>
<td>0.20</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>HC-05主从机一体蓝牙模块 Arduino无线蓝牙串口通讯模块串口通讯</td>
<td>1</td>
<td></td>
<td></td>
<td>16.29</td>
<td>15.99</td>
<td>0.30</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>SYB-170微型面包板 mini面包板 实验平台</td>
<td>3</td>
<td>白</td>
<td></td>
<td>0.83</td>
<td>2.37</td>
<td>0.12</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>黑色 NRF24L01无线模块 24L01+升级版 黑金刚</td>
<td>3</td>
<td>i55</td>
<td></td>
<td>3.50</td>
<td>10.20</td>
<td>0.30</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>显示模块 MAX7219点阵模块 控制模块 单片机模块 送线</td>
<td>2</td>
<td>B505</td>
<td></td>
<td>6.50</td>
<td>12.80</td>
<td>0.20</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td>40P 杜邦公头转母头 杜邦线 公对母 面包线 20CM 长 40根一排 特价</td>
<td>2</td>
<td>H1</td>
<td></td>
<td>2.00</td>
<td>3.70</td>
<td>0.30</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td>杜邦线/彩排线/40P-40p双头/公对公1P-1P/长10CM/10厘米</td>
<td>2</td>
<td>C32</td>
<td></td>
<td>2.10</td>
<td>4.00</td>
<td>0.20</td>
</tr>
<tr>
<td>11</td>
<td></td>
<td>2.54间距单排针-每排40P</td>
<td>5</td>
<td>B801</td>
<td></td>
<td>0.20</td>
<td>1.00</td>
<td>0.00</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td>单排螺丝孔排母 排针母座 间距2.54MM 1*40P 针座排针插座 内金外壳</td>
<td>2</td>
<td>B705</td>
<td></td>
<td>0.80</td>
<td>1.40</td>
<td>0.20</td>
</tr>
<tr>
<td>13</td>
<td></td>
<td>10CM对母杜邦线 40pin 10cm米 特价促销 量大价格优惠</td>
<td>2</td>
<td>C31</td>
<td></td>
<td>1.60</td>
<td>3.00</td>
<td>0.20</td>
</tr>
</tbody>
</table>
### 9.2 Materials in November 2016

<table>
<thead>
<tr>
<th>序号</th>
<th>图片</th>
<th>宝贝名称</th>
<th>数量</th>
<th>销售属性</th>
<th>商家编码</th>
<th>单价</th>
<th>金额</th>
<th>优惠</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><img src="image1" alt="图片" /></td>
<td>全新芯片 L298N电机驱动板模块 直流步进电机电机器人智能车</td>
<td>2</td>
<td></td>
<td>B83</td>
<td>7.00</td>
<td>12.55</td>
<td>1.45</td>
</tr>
<tr>
<td>2</td>
<td><img src="image2" alt="图片" /></td>
<td>双轴按键拨杆 PS2游戏拨杆控制杆传感器 电子积木</td>
<td>2</td>
<td></td>
<td>B67</td>
<td>1.90</td>
<td>3.70</td>
<td>0.10</td>
</tr>
<tr>
<td>3</td>
<td><img src="image3" alt="图片" /></td>
<td>高亮6<em>6</em>4.3MM 轻触按键 微动开关 微动开关 轻触开关 按键开关</td>
<td>1</td>
<td>6<em>6</em>5个 6<em>6</em>5</td>
<td></td>
<td>3.00</td>
<td>3.00</td>
<td>0.00</td>
</tr>
<tr>
<td>4</td>
<td><img src="image4" alt="图片" /></td>
<td>按键开关 8*8MM 自锁 双排 6脚八脚 黑白黑按键开关 50只</td>
<td>1</td>
<td>R1702</td>
<td></td>
<td>7.00</td>
<td>6.75</td>
<td>0.25</td>
</tr>
<tr>
<td>5</td>
<td><img src="image5" alt="图片" /></td>
<td>【新品】实验平台 UNO R3 主板固定通用型 升级版 亚克力</td>
<td>1</td>
<td></td>
<td>B612</td>
<td>2.90</td>
<td>1.80</td>
<td>1.10</td>
</tr>
<tr>
<td>6</td>
<td><img src="image6" alt="图片" /></td>
<td>电池盒 不带盖子 4节5号电池盒 四位5号电池座</td>
<td>2</td>
<td></td>
<td>O116</td>
<td>0.88</td>
<td>1.76</td>
<td>0.00</td>
</tr>
<tr>
<td>7</td>
<td><img src="image7" alt="图片" /></td>
<td>Arduino 专业9V 电池插 DC插头 DC插头</td>
<td>3</td>
<td></td>
<td>E82</td>
<td>0.48</td>
<td>1.44</td>
<td>0.00</td>
</tr>
<tr>
<td>8</td>
<td><img src="image8" alt="图片" /></td>
<td>面包板黑色电源模块+830孔 高级大面包板+65条彩色面包线 套</td>
<td>1</td>
<td>黑色</td>
<td>H32+H21+H31</td>
<td>9.30</td>
<td>8.90</td>
<td>0.40</td>
</tr>
<tr>
<td>9</td>
<td><img src="image9" alt="图片" /></td>
<td>MB-102 玫瑰金面包板 830 孔面包板 实验连接板</td>
<td>1</td>
<td></td>
<td>H21</td>
<td>3.20</td>
<td>3.10</td>
<td>0.10</td>
</tr>
<tr>
<td>10</td>
<td><img src="image10" alt="图片" /></td>
<td>UNO R3 开发板 外壳 学习板 盒子开发板 螺丝</td>
<td>1</td>
<td></td>
<td>O68</td>
<td>2.50</td>
<td>1.95</td>
<td>0.55</td>
</tr>
</tbody>
</table>
### 9.3 Materials in January 2017

<table>
<thead>
<tr>
<th>#</th>
<th>图片</th>
<th>宝贝名称</th>
<th>数量</th>
<th>归类属性</th>
<th>商家编码</th>
<th>单价</th>
<th>金额</th>
<th>税</th>
<th>优惠</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>算法小车底座 金属制 带脚小车 金属制 透明小车</td>
<td>5</td>
<td>金属光泽</td>
<td>H04</td>
<td>22.95</td>
<td>113.75</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>全新芯片 L298N驱动板盖板 高压驱动电机小车</td>
<td>5</td>
<td>红色</td>
<td>83</td>
<td>6.77</td>
<td>33.85</td>
<td>1.08</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>Arduino UNO R3 开发板 UNO R3 开发板</td>
<td>5</td>
<td>黑色</td>
<td>H42</td>
<td>17.30</td>
<td>86.50</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>24L01/24L01转换器 24L01转换器 高压共模</td>
<td>5</td>
<td>灰色</td>
<td>955</td>
<td>3.60</td>
<td>18.00</td>
<td>0.60</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>智能小车 金属铝板 飞机 金属铝板 金属铝板</td>
<td>5</td>
<td>透明</td>
<td>M1805</td>
<td>16.98</td>
<td>84.95</td>
<td>0.35</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>杜邦线 2芯 2芯 2芯 2芯 2芯 2芯 2芯 2芯</td>
<td>5</td>
<td>黄色</td>
<td>H3</td>
<td>2.18</td>
<td>10.90</td>
<td>0.65</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>杜邦线 2芯 2芯 2芯 2芯 2芯 2芯 2芯 2芯</td>
<td>5</td>
<td>绿色</td>
<td>C32</td>
<td>2.00</td>
<td>9.95</td>
<td>0.65</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>杜邦线 2芯 2芯 2芯 2芯 2芯 2芯 2芯 2芯</td>
<td>5</td>
<td>白色</td>
<td>C21</td>
<td>1.85</td>
<td>9.25</td>
<td>0.65</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td>NANO V3 0.328mm 连接器 金属器件 金属器件 金属器件 金属器件 金属器件 金属器件</td>
<td>1</td>
<td>透明</td>
<td>A54</td>
<td>23.40</td>
<td>21.80</td>
<td>1.60</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td>ZY26 无刷驱动器mini型套板 无刷驱动器mini型套板</td>
<td>1</td>
<td>金属</td>
<td>H1207</td>
<td>1.30</td>
<td>1.30</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td>ZY50 无刷驱动器mini型套板 无刷驱动器mini型套板</td>
<td>2</td>
<td>透明</td>
<td>V93</td>
<td>9.10</td>
<td>18.20</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td>10V电池充电 10V电池充电 10V电池充电 10V电池充电</td>
<td>10</td>
<td>透明</td>
<td>D32</td>
<td>0.28</td>
<td>2.65</td>
<td>0.15</td>
<td></td>
</tr>
</tbody>
</table>
9.4 Materials in March 2017

<table>
<thead>
<tr>
<th>序号</th>
<th>图片</th>
<th>产品名称</th>
<th>数量</th>
<th>销售属性</th>
<th>图号</th>
<th>价格</th>
<th>金额</th>
<th>折扣</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>![图片]</td>
<td>2014新款UNO R3 开发板</td>
<td>1</td>
<td>一级</td>
<td>H42</td>
<td>17.00</td>
<td>16.50</td>
<td>0.96</td>
</tr>
<tr>
<td>2</td>
<td>![图片]</td>
<td>12V电位器（9V电压）电位器 (15CM长) Δ型电位器</td>
<td>5</td>
<td>二级</td>
<td>D32</td>
<td>6.288</td>
<td>1.256</td>
<td>0.19</td>
</tr>
</tbody>
</table>

9.5 Testing video – Control robot with Android phone via Bluetooth
https://goo.gl/txHtQ1

9.6 Testing video – Wireless communication via nRF
https://goo.gl/bwUlzz

9.7 Testing video – Control robot with self-made controller via nRF
https://goo.gl/mm5g53

9.8 Testing video – Object tracking by color
https://goo.gl/ySY4W0

9.9 Testing video – Object tracking by color – robot
https://goo.gl/7HW9ul

9.10 Testing video – Robot move to destination automatically
https://goo.gl/OxTp9

9.11 Testing video – System flow
https://goo.gl/vvlRuA