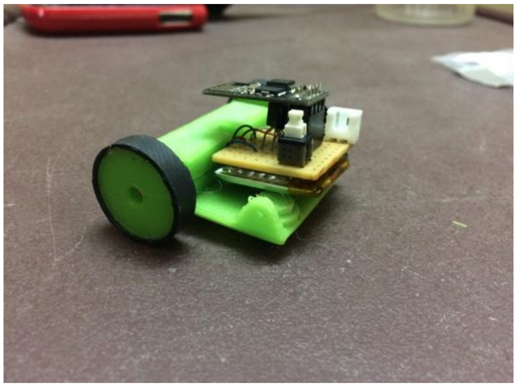


SELF ORGANIZING ROBOTS.

ABSTRACT

This paper introduces a new class of human-computer interfaces comprised of many autonomous robots that handle both display and interaction. The platform consists of a collection of custom-designed wheeled robots each 41x35 mm in dimensions, a Wi-Fi module, and object tracking using image processing techniques. They are controlled by custom written algorithms and python libraries. We illustrate the potential of group interface through a set of algorithms implemented with custom designed robots, and discuss general design considerations unique to collective intelligence.

LITERATURE SURVEY

The concept of collective intelligence was first introduced by Gerardo Beni, Suzanne Hack-wood, and Jing Wang when they were investigating the properties of simulated, self-organizing agents in the framework of cellular robotic systems. In Proceedings of the Seventh Annual Meeting of the Robotics Society of Japan, Tokyo, Japan. Eric Bonabeau, Marco Dorigo and Guy Theraulaz extended the restrictive context of this early work to include any attempt to design algorithms or distributed problem-solving devices inspired by the collective behaviour of social insect colonies, such as ants, termites, bees, wasps, and other animal societies. The aim of our project is to make a group of robots which will construct an input schematic given by the user on a given platform.

EXISTING SYSTEM

Zooids: The project is executed by students of Stanford University. The system introduced swarm user interfaces; It is an open-source open-hardware platform for developing tabletop swarm interfaces. It consists of a collection of custom-designed wheeled micro robots each 2.6 cm in diameter, a radio base-station, a high speed DLP structured light projector for optical tracking, and a software framework for application development and control.

Kilobots: Kilobots are done by students of Harvard University. Kilobots should be operated on a smooth, flat surface to ensure proper robot mobility. To aid communication, the surface should be glossy or reflective. The robots beneath the overhead controller in a about a one meter diameter region will be able to receive messages.

PROPOSED SYSTEM

On basis of the above survey and after referring various technical papers a new and innovative project idea was developed. The aim of the project 'Self organized robots' was to build a group of robots which would construct an input schematic given by the user on a given platform.

The project included study of various fields in Electronics and Computer Science such as Artificial Intelligence, Collective Intelligence, Embedded Systems, Wireless Sensor Networks and Computer Networks. It included Wireless communication between the robots. It consisted of a group or swarm of robots which were guided by a server using various algorithms programmed in its micro-controller. The robot tracking is done by using image processing technique.

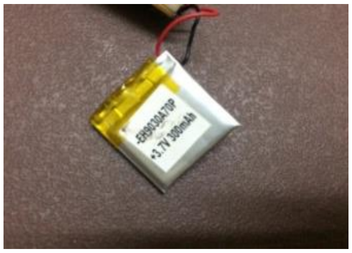
TOOLS

The greatest difficulty faced in the whole system is to make the size of each robot as small as possible. It required building custom designs for the robots and rapid prototyping is employed for the modifications based on practical implementations.

Hardware Requirements

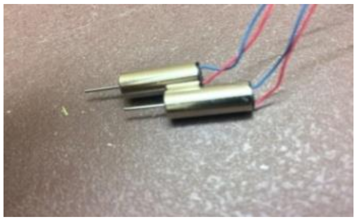
Battery

A Rechargeable Lithium-Polymer battery cell is used to power the entire robot. It has the technical specifications of 3.7v with 300mAh capacity and 20C discharge rate. This optimum design of battery not only helps in keeping the robotic structure in less weight and small size but also supplies required power efficiently. A single charge of the battery can run the robot for around 20 minutes of continues usage and 1 hour of idle standby.



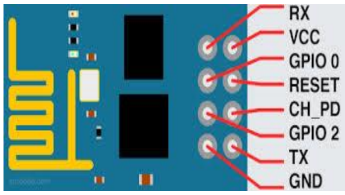
Coreless DC motor

Two coreless DC motors are being employed since it is a two wheeled robot. This two wheeled construction of the robot gives it a 3-degree locomotion i.e. it can move in forward, right and left directions. The motor's specifications are 3.3v and 0.1A. The motor is specifically selected for its high speed acceleration and smaller size. It is of size 16*7mm in dimensions.



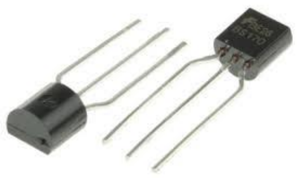
ESP8266 Wi-Fi Controller

The ESP8266 Wi-Fi Module is a self contained Microcontroller with inbuilt Wi-Fi options. It can establish new networks and also able to connect to already available networks. It has a very compact design with two GPIO pins which makes it the most suitable board for this project. It can be easily programmed by C language Arduino IDE.



BS170 N-Channel MOSFET

The BS170 is a logic level MOSFET, which is capable of showing qualities of the higher rating MOSFETs in small signal applications. This property of it used to select it for the switching purpose of the motor using the microcontroller. It has a specification of V_{ds}= 60v and I_d= 500 mA.



Software Requirements

Python

Python is a widely used high-level programming language for general-purpose programming Python features a dynamic type system and automatic memory management and support multiple programming paradigms, including object oriented, imperative, functional programming, and procedural styles.

NumPy

NumPy is an extension to the Python programming language, open-source software adding support for large, multi-dimensional arrays and matrices, along with a large library of high-level mathematical functions to operate on these arrays. NumPy incorporates features of the competing Numarray into Numeric, with extensive modifications.

Matplotlib

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython,

OpenCV

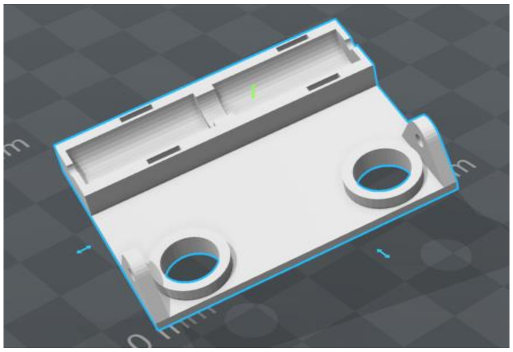
OpenCV (Open Source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision. OpenCV is a set of libraries that help to quickly write code for solving computer vision problems. It handles the nitty-gritty details of the algorithm and allows writing optimized and fast code.

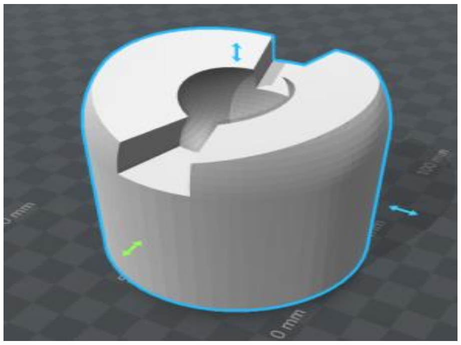
DESIGN

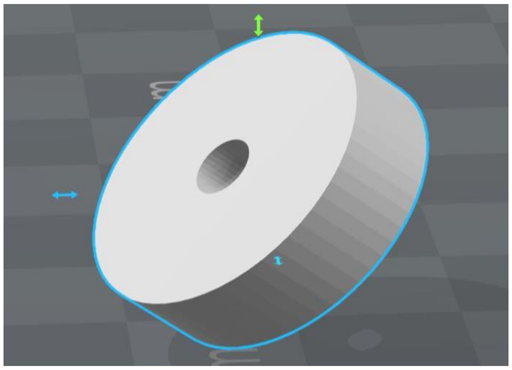
SolidWorks Software

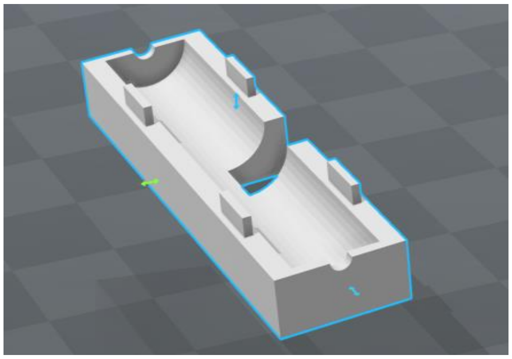
SolidWorks is a solid modeler, and utilizes a parametric feature-based approach to create models and assemblies. Building a model in SolidWorks usually starts with a 2D sketch and 3D model can be derived later. The sketch consists of geometry such as points, lines, arcs, and spines. Dimensions are added to the sketch to derive the required 3D model.

The 3D model based on the spatial requirements which has been done using the SolidWorks software is displayed below,







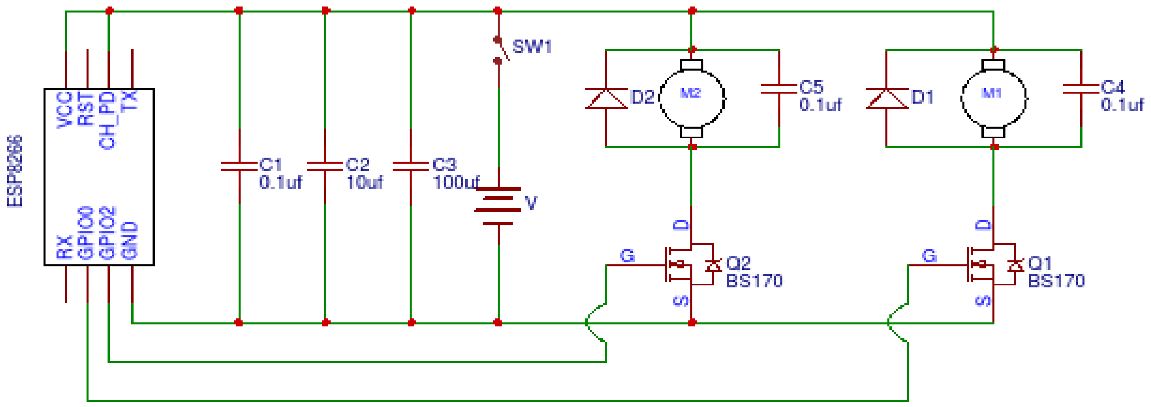


3D Printing Technology

3D Printing is done using 3D Printer. The 3D printer used for prototyping is WANHAO Duplicator. The WANHAO Duplicator makes solid, three-dimensional objects out of melted WANHAO PLA (PolyLactic Acid) Filament. 3D model derived from SolidWorks is translated into instructions for the WANHAO Duplicator and sent to the machine via USB cable or SD Card. Then the WANHAO Duplicator heats the WANHAO PLA Filament and squeezes it out through a nozzle to make a solid object layer by layer. This technique is called fused deposition modeling (FDM).

Circuit Design

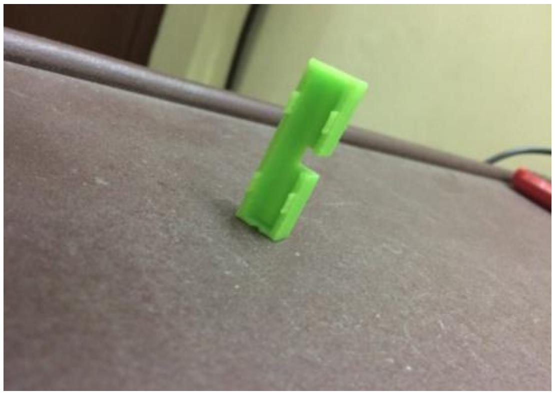
The circuit is designed mainly on the consideration of simplicity and smallest size possible. So, very basic components are only used in the electrical circuit.

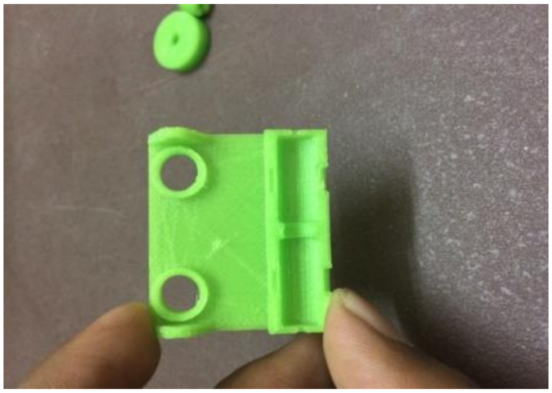


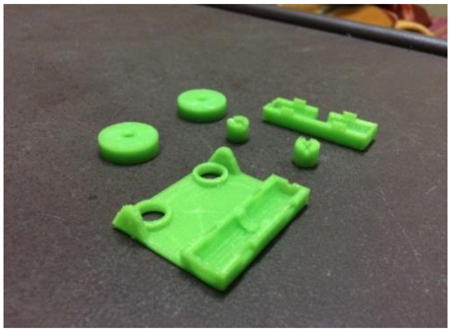
IMPLEMENTATION

Hardware Implementation

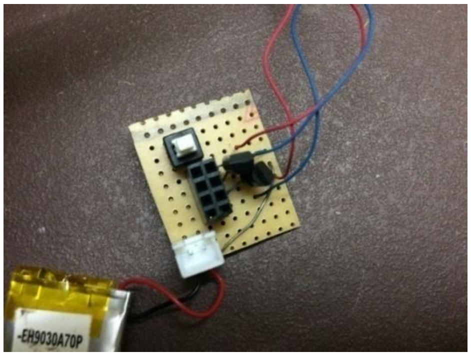
The above 3D models has been printed with the help of a 3D printer, the printed parts are shown below,

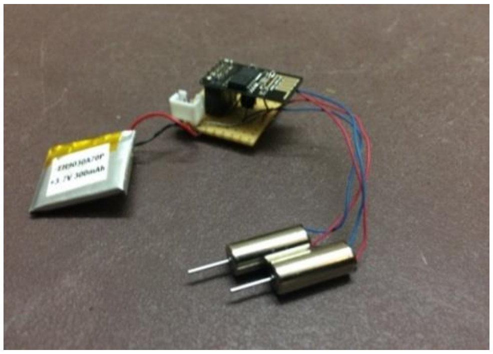




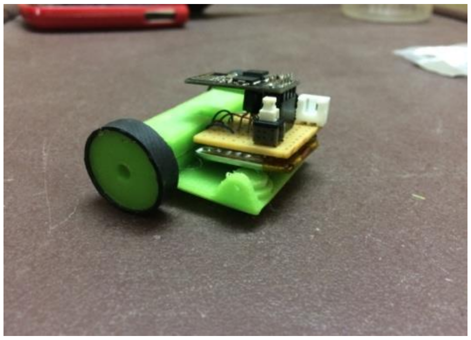


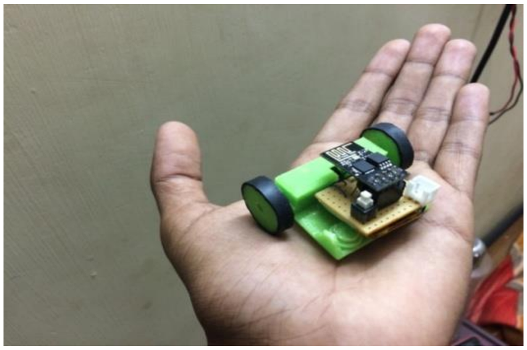
Electrical Circuit



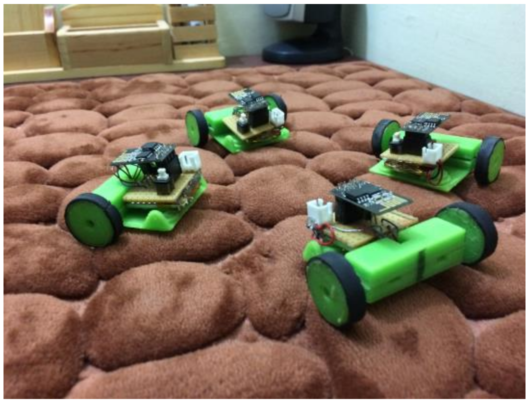


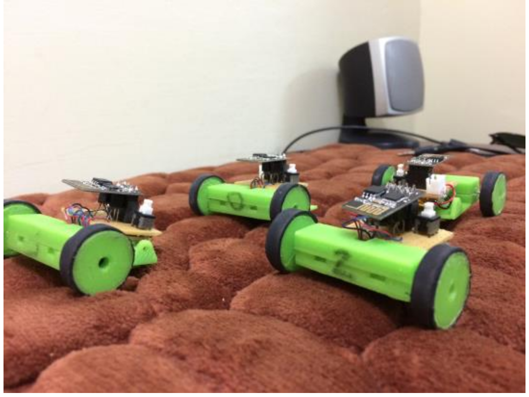
Assembly and Final Product





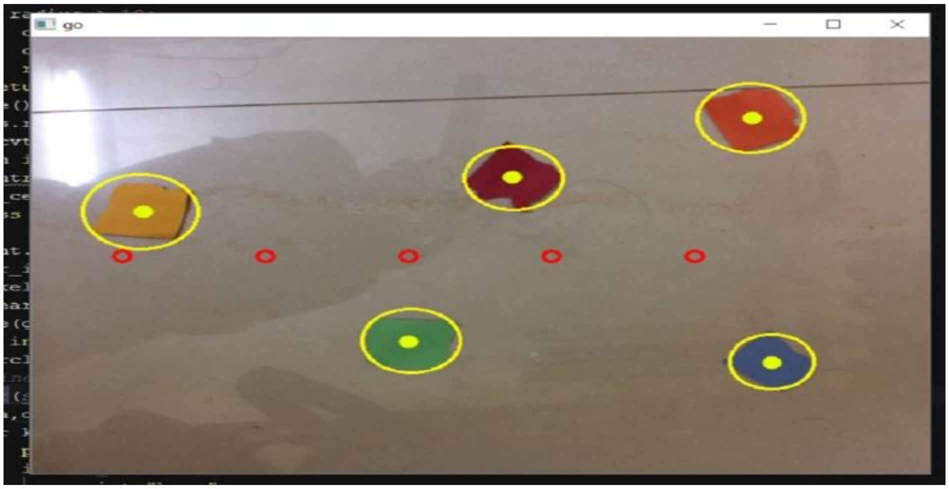
Swarm Group:





WORKING

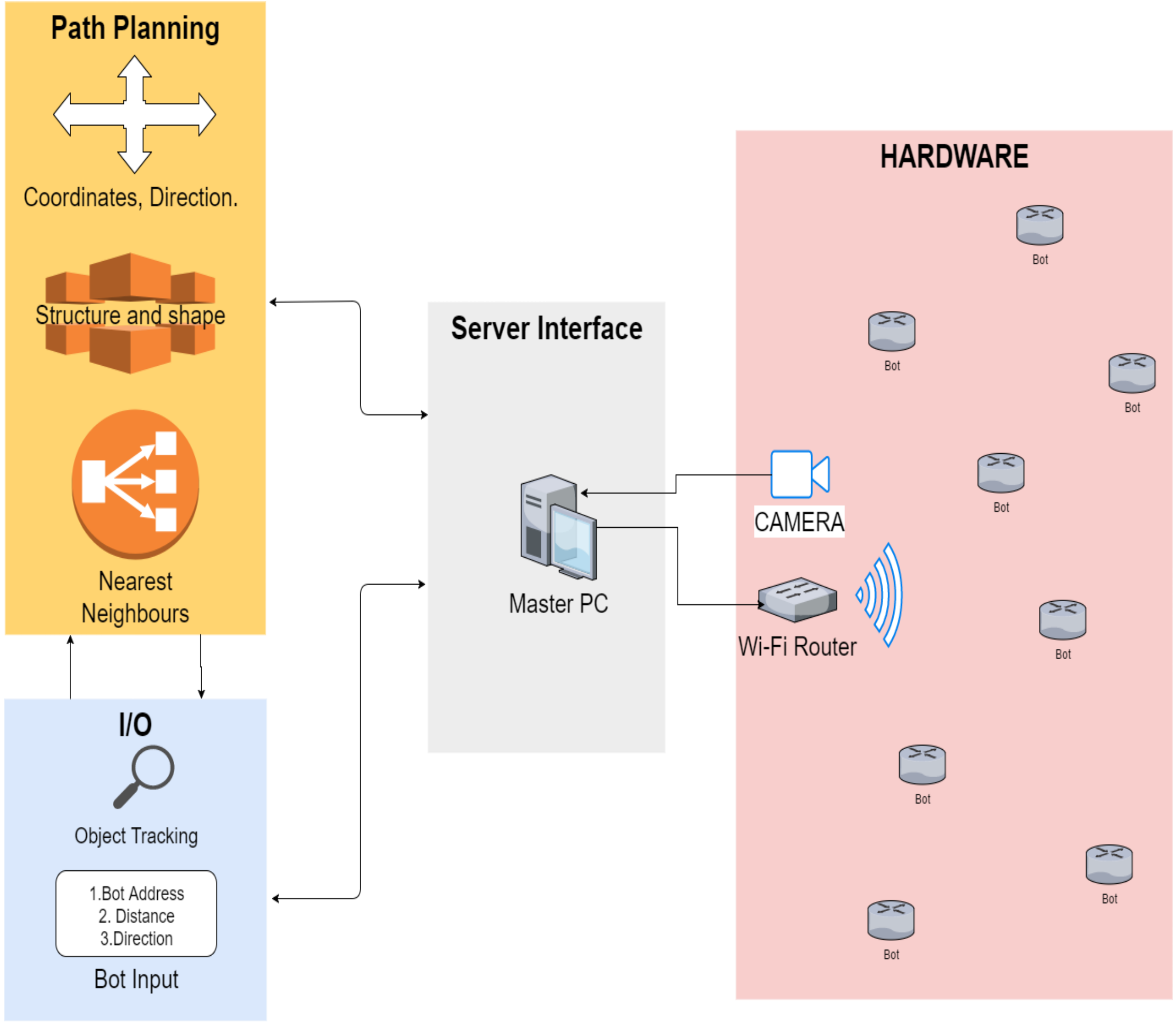
Robot Position Tracking:



Robots are tracked with the help of a python library named openCV. It is highly used in the fields of image processing. Each robot is given a separate color on top of it so, a simple color identification technique is used to find the color tag of the robot. Each color tag represents a unique bot ID from which the Wi-Fi address of the robot can be found.

The image contains a color detection sample for 5 different colors simultaneously and also using image processing the centers of the each color contours are also found. This gives a great way of robot tracking even in very ruff conditions.

Architecture:



The communication between the robots and the master is via a secure Wi-Fi local network. This local network paves way for two way communication between the master and the slaves. The master is usually a PC with a custom built program running on it. This master program governs the position, movement and arrangement of the robots.

A video camera is used to keep track the position of the robots using color detection methods and it's a continuously running loop once the program has started until the goal is achieved.

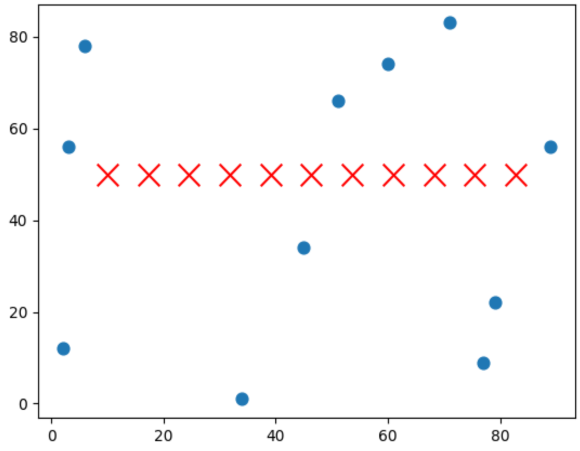
Skeleton Structures

The algorithms are designed in such a way that it is adaptive to any given area and form the same structure. When the user gives any surface as input the program calculates the skeletal structures and minimal number of robots for that structure. We have implemented two skeletal structures as pre defined default structure, if needed in future it is very trivial to add more and more skeletal structures to the program.

Line structure

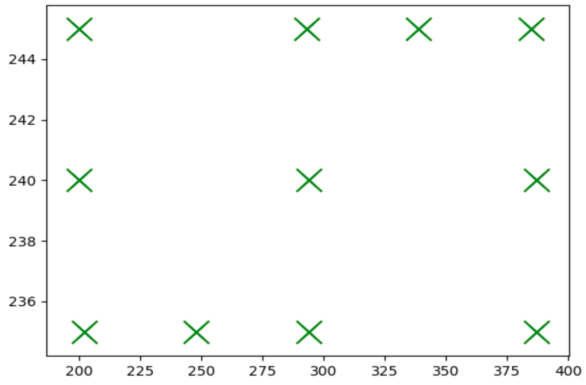
The blue points in the picture indicate the robots which are placed in a random manner.

The Red Cross indicates the target skeleton points which are generated from the custom built algorithm for line formation for any given plane.



S-structure

The green cross indicates the target skeleton points which are generated from the custom built algorithm for 'S' formation for any given plane.



Path Planning:

After acquiring both the current position and destination points, the algorithm plans the path for the robots to move in its degree of freedom. This process of path planning happens in two stages,

Initially, the bot points are allocated to random points in the skeleton and the distance is calculated. This follows no specific format or computation so it's a quick way but this causes the total distance travelled by the robots to be larger. This practice makes the whole process very time consuming and it wastes a lot of battery power. So, an algorithm is employed to rectify this disadvantage.

Nearest Neighbours Algorithm:

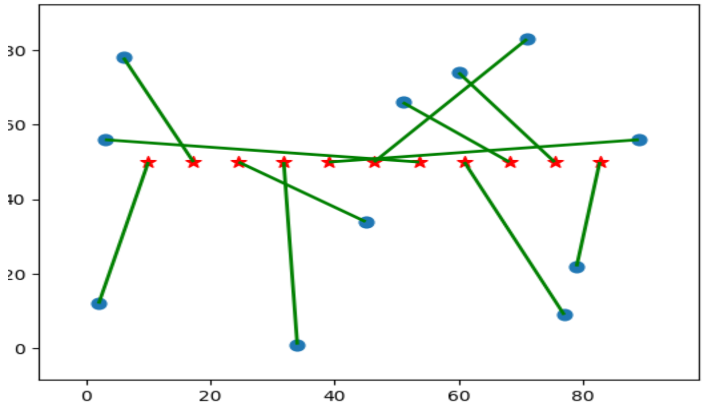
The path derived without using nearest neighbour algorithm. By observation, it can be seen that the distance travelled by the robots is much longer. So, applying nearest neighbor algorithm the following paths are obtained which are more optimum to the given set of paths.

The distance is found for all the robots given and the target point is selected for a particular robot by selecting the smallest available Euclidean distance that had been found before. The Euclidean distance can be found by the following formula,

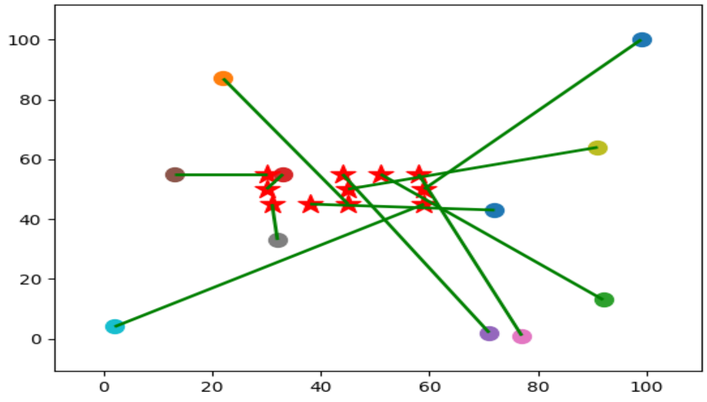
$$\mathrm{dist}(x, y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

Euclidean distance = sq root (bot point – target point).

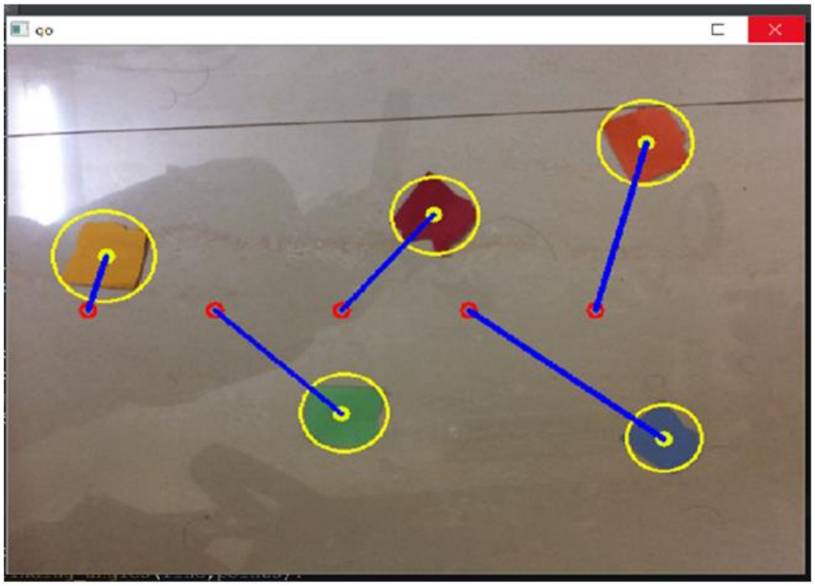
In this picture, the green lines indicate the path traced by the robots for the line formation after applying the nearest neighbouring algorithm.



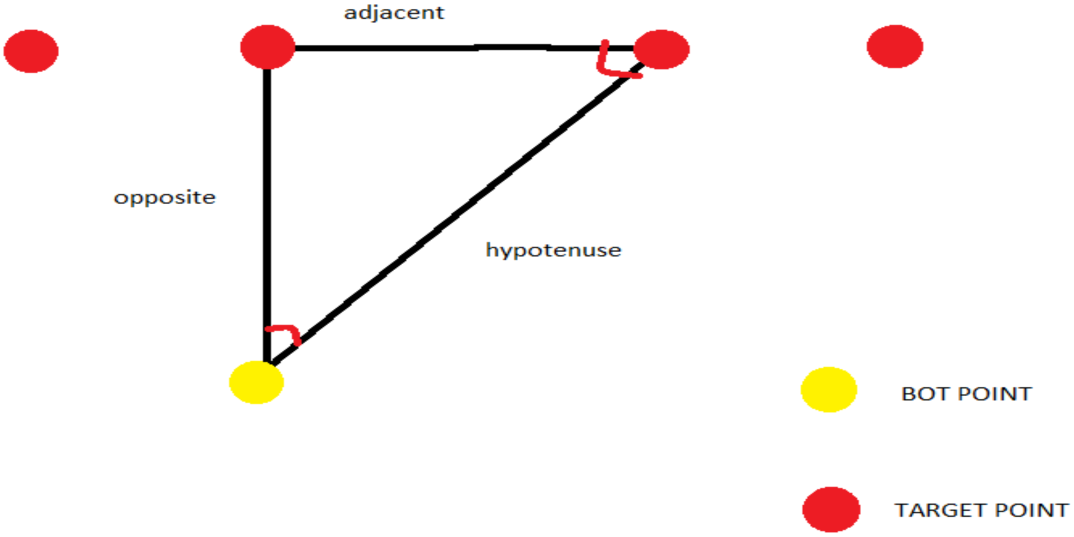
The same algorithm is applied for the S-structure formation. The distance is greatly reduced in both the cases.



Real time tracking and application of the nearest neighbor algorithm has been shown this image. The blue lines represent the paths that should be traced down by the robot to attain the optimum point for forming the line structure.



Feeding data to the Robots:



The robot should be fed with the data of distance and angle for each specific for the structure formation. So, to calculate distance and angle the following method is used

To find the distance, hypotenuse is found to the triangle formed while adjacent is the median for the whole plane. The angle that to be tilted is found by using the formula,

$$\alpha = \sin^{-1}(\text{opposite/hypotenuse})$$

(Distance, α) is appended to a list which contains the same for all other robots. So, the ESP Arduino obtains this distance and angle and facilitates the robot to move in this α direction to that distance.

CONCLUSION

After studying different research papers and scientific journals we found that our project “Self Organizing Robots” had many advantages over the other conventional robotics techniques.

Firstly, the small size of the client robot helps us to carry the robots conveniently to any given location to perform work. Secondly, the server which has been developed makes the task of the user to assign work to the robots rudimentary.

Using the concept of ’Collective intelligence’ reduces the cost and increases the expandability of the robots. A particular robot can be replaced in case of damages instead of replacing the whole.

In future, size can be extremely reduced and the same algorithm can be implemented irrespective of their platform. It has vast scopes in various fields such as Medicine, Defence etc.

REFERENCES

• BASE PAPER DETAILS

1.Kilobot: A Low Cost Scalable Robot System for Collective Behaviors Michael Rubenstein, Christian Ahler, Radhika Nagpal IEEE Intl. Conf on Robotics and Automation (ICRA), 2012.

2.Mathieu Le Goc, Lawrence Kim, Ali Parsaei, Jean-Daniel Fekete, Pierre Dragicevic, Sean Follmer. Proceedings of the 29th Annual ACM Symposium on User Interface Software & Technology (UIST 2016), June 2016, Tokyo, Japan. (In press).

• OTHER REFERENCES

1.Collective Transport of Complex Objects by Simple Robots: Theory and Experiments Rubenstein, Cabrera, Werfel, Habibi, McLurkin, Nagpal Intl. Conf. on Autonomous Agents and Multiagent Systems (AAMAS), May 2013.

2.Programmable Self-Assembly in a Thousand-Robot Swarm Michael Rubenstein, Alejandro Cornejo, Radhika Nagpal Science, Vol 345, no 6198, 15 Aug 2014.

3.Rise of the Swarm, Communications of the ACM, 2013. SEAS Article (Kilobots are leaving the nest), 2011 Slashdot Article (Nov 2011).