A Metric for Hexagonal Grid-World Cooperative Object Recognition Tasks for a Swarm of Agents

David King[1] and Philip Breedon[1]

Nottingham Trent University, Nottingham, NG1 4BU, UK

Abstract. In the future nano-robots could be used in in-vivo medical applications. By using lattice robots as a unit of measurement tumours could be assessed as malignant or benign by considering the shape of their boundaries through cooperative object recognition. With the objective of moving towards this end goal research has been carried out using simulations on hexagonal lattices that allow simple agents to cooperate to distinguish between two objects' shapes. This paper determines a method which gives a value of difference between any two objects constructed on a hexagonal lattice. This difference value will provide a metric for future cooperative object recognition scenarios. The measure of the difference value was calculated considering varying degrees of knowledge about sections of the objects' boundaries. The results were compared to a simulated task with agents that were homogeneous, anonymous, had a limited sensor range and no common coordinate system. It was found that the difference value metric provides a significant correlation to the time taken to complete a series of cooperative object recognition scenarios.

Key words: multi-agent systems, swarm robotics, cooperative object recognition.

1 Introduction

With continuing advances in miniaturising electronics there is an increased amount of interest in nano-scale robots for medical applications [1, 2]. These nano-robots have suggested applications including isolating harmful cells in the blood stream [3]; diagnosis of endogenous diseases [4] and antibody delivery systems [5]. Due to the robots' individual sizes, a limitation is placed on the robots' individual computational and physical capability when compared to larger robots. One approach that may help overcome this limitation is swarm robotics. Swarm robotics takes inspiration from the behaviours of social insects (ants, bees and termites) to provide multi-agent systems methods to complete tasks. Through the agents local interactions with each other and the environment they inhabit they can coooperate intelligently [6]. Although specific behaviours have been mimicked directly: foraging [7], cooperative transport [8] and self-assembly [9], inspiration alone can be taken from the decentralised approach.

Utilising a swarm robotic approach cooperative object recognition is achievable, where individuals who cannot assess an object alone can cooperate to identify an object through interacting with it and the other members of the swarm. In addition to the miniaturisation of robots, this method will potentially allow for swarms of nano-robots who can identify unwanted or dangerous entities in human bodies through shape analyses. For example tumours that can be assessed by their shape to determine if they are malignant or benign [10, 11]. Currently images of tumours are analysed by a radiologists. The system envisioned here would instead have a swarm of nano-bots injected into a person, who would cooperate to locate and identify tumours, and could potentially be capable of destroying them. To do this requires that the robots be able to measure the boundary of the tumour.

Groups of homogeneous lattice robots, who individually are approximations of regular shapes, can connect together into ordered formations. Two-dimensional lattice structures can be formed and rearranged by triangular, square and hexagonal robots [12, 13, 14, 15, 16, 17]. Three-dimensional lattice structures are formed by cubical and spherical robots. In some cases these cubical and spherical robots are divided into two parts which can rotate relative to each other. Structures formed from these types of robot can perform clustered walks [18, 19] and can replicate themselves [20]. Lattice robots that do not have the capability provided by having two halves rotate relative to each other have also been demonstrated to form different structures where the robots' initial starting structure is a uniform block. The robots detach from each other at specific points allowing them to form the required structures [21].

One advantage of lattice based robots, especially where smaller sized and simpler robots are considered, is their ability to become a unit of measurement. Giplin and Rus [22] show cube agents, Robot Pebbles, which are capable of duplicating inert shapes with identical resolution to the agents themselves. To do this the Robot Pebbles must fully surround the shape they are replicating and pass a signal from agent to agent around the object to determine its shape. This information is then used to form the replica of that shape from the Robot Pebble agents themselves. This type of lattice based measurement would be beneficial where nano-robotic agents neighbour a tumour, or any other object, to determine any unique distinguishing features in its boundary. One difference from Gilpin and Rus's research is that the entire object need not be surrounded as only distinguishing features need to be identified. This has previously been shown to be possible on a hexagonal lattice where a swarm of simple agents could distinguish approximations of hexagons from triangles when cooperating with each other [23, 24]. As more complex and varied object shapes are considered on this lattice a metric of difficulty is required to determine a suitable range of task scenarios. The metric will allow for suitable scenario selection for future research into training methods for the swarm's behaviours. Training the swarm will allow it to adjust to different cooperative object recognition tasks, as well as give guidance on how much memory the agents will require to complete different scenarios.

To determine a metric a method is required for comparing two objects' shapes similarities to each other whilst considering how much information is currently known about them. The object shapes in a hexagonal grid-world could be considered binary images and the arena the background. Shape coding has been used to store binary images in a range of different ways [25, 26]. Most relevant of these techniques is chain-coding which maps the relative positions of the neighbouring pixels at the boundary of a shape on a lattice in a linear order [25]. Usually chain-coding uses a square lattice but it has also been completed on a hexagonal lattice [27]. The paths the chain takes to code the shape can follow the pixels at the inside or the outside of the shape [28, 29]. There are lossy techniques, where estimations are used to decrease the memory required to store the shape information. However, lossy techniques are not suitable for this application as the shapes need to be described accurately. Chain-coding also considers position and orientation when describing shapes. As the objects are considered identical despite their location and rotation these two factors are not required for the cooperative object recognition task metric.

By determining a method of describing the boundary of objects' shapes on a hexagonal lattice, without considering their location and rotation, this paper provides a metric for comparing two object shapes. This metric is then tested using a series of cooperative object recognition scenario simulations involving a swarm individually simplistic agents.

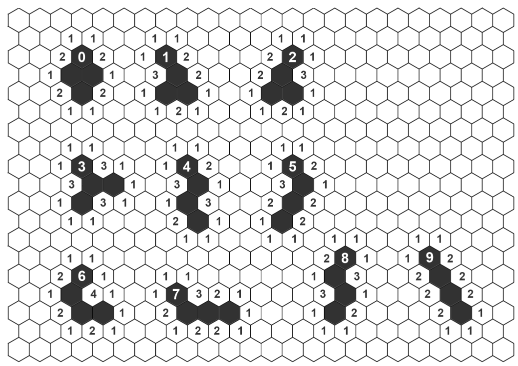
2 Simulation Method

In order to understand what the metric will be measuring the simulation platform first needs describing. This research uses a hexagonal lattice as an arena for testing cooperative object recognition tasks. Each hexagonal cell can be: an object cell, which are grouped to form object shapes; a single agent, termed a hBot; a border, which the hBots cannot pass; or an empty space.

2.1 Object Shapes

Object shapes are constructed by grouping a number of neighbouring hexagonal object cells together, where the simplest object shape is a single cell. An object shape can be described without considering rotation or location by the contours of its boundary region, this is termed the data-chain. If each cell that neighbours an object shape is given a value determined by how many object cells it touches, the data-chain is an array of these values arrange in a clockwise manner represented by the sequence which is first in lexicographical order. All the object shapes with four object cells (ID0 - ID9) are shown, Fig. 1, with the numbers in each of their surrounding cells indicating how many object cells they neighbour, resulting in the data-chains listed.

A scenario is defined by the two object shapes that the hBots must cooperate to distinguish between. In a scenario one of the object shapes is classed as valid, this is the object shape the hBots must identify and remove. The other object shape is invalid; this object shape must not be removed by the hBots. It is the difference between these two object shapes which determines the difficulty of the scenario which the metric will determine (section 3.0).



ID	Data-Chain
0	{1,1,2,1,2,1,1,2,1,2}
1	{1,1,1,2,2,1,1,2,1,1,3}
2	{1,1,1,3,1,1,2,1,1,2,2}
3	{1,1,1,3,1,1,1,3,1,1,1,3}
4	{1,1,1,2,1,3,1,1,1,2,1,3}
5	{1,1,1,2,1,2,2,1,1,1,2,3}
6	{1,1,1,2,1,2,1,2,1,1,1,4}
7	{1,1,1,2,2,1,2,1,1,1,3,2}
8	{1,1,1,3,1,2,1,1,1,3,1,2}
9	{1,1,1,2,2,2,1,1,1,2,2,2}

Fig. 1. All ten object shapes (ID0 - ID9) with four object cells with their surrounding cells indicating the number of object cells they are in contact with which determine their data-chains.

2.2 hBot Agents

The hBots used in the cooperative object recognition task are homogeneous, anonymous, have no common coordinate system and are not aware of their position relative to the arenas coordinate system. The hBots are modelled with local sensor and communication, both with a range of one cell. This capability allows the hBots to determine their current state from their immediate surroundings and to communicate this state to any hBots that they neighbour, providing a feedback loop between neighbouring hBots. The current state-level of the hBot indicates how much knowledge it has about the object's shape, whilst the state of the hBot describes this knowledge specifically.

- A hBot at state-level 0 is not in contact with an object shape and therefore has no knowledge about the object shape, this is state 0.
- A hBot at state-level 1 knows the number of object cells it is neighbouring: one, two, or three and is equivalent to knowing a single value in a data-chain. This knowledge is represented by states 1, 2 and 3 respectively. This value could theoretically be between one and six, however in this research the object shapes were limited as to only allow situations where the values one, two and three occur, to initially reduce the total number of states.
- A hBot at state-level 2 knows as much as three individual agents at state-level 1, as it knows its own state-level 1 state and the states of its neighbours. This is equivalent to knowing three consecutive values in a data-chain and includes states 4 – 21.
- A hBot at state-level 3 knows as much as five individual agents at state-level 1, or three agents at state-level 2. This is equivalent to knowing five consecutive values in a data-chain and includes states 22 – 264.

As certain states are only achievable when the hBots interact with certain object shapes, these states can be used by the hBots to distinguish between object shapes. The total number of states, 265, is determined in part by the number of object cells the hBots can distinguish between, the way the hBots move from one state-level to the next and the possible states the hBots can reach whilst neighbouring each other. A hBot does not distinguish between the position of its neighbours; when hBots determine their next state they do so by sending its own state and the states of its neighbours to the state-rule table with the following format: [own][lowest neighbour][highest neighbour]. This means that the symmetrical features of the object shape, as described by sub-chains from the data-chain, will appear identical. For example: where two sets of three hBots are neighbouring similar parts of object shapes that are mirror images of each other, Fig. 2. The hBot A neighbours an object shape with the data-chain {1,1,1,2,2,1,2,1,1,1,3,2} and the hBot B neighbours an object shape with the data-chain {1,1,1,2,1,2,2,1,1,1,2,3}. The three hBots including hBot A are on the sub-chain {{1,3,2}} and the three hBots including hBot B are on the sub-chain {{2,3,1}}. Although these sub-chains are different the resulting state of hBot A and hBot B at state-level 2 will both be the result of referencing the state-rule table with values [3][1][2], which returns state 17.

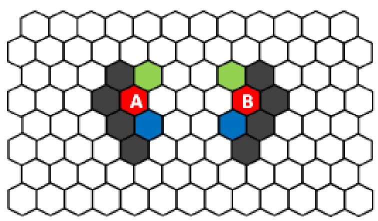


Fig. 2. hBot A and hBot B will both change to the same state even though their neighbours are the opposite way round. Green is state 1, Blue is state 2, Red is state 3.The resulting states for both hBots A and B is state 17.

For each time-step hBots move with equal probability to any of their neighbouring cells, unless the cell is an object cell or a border cell. If an hBot attempts to move into a cell containing another hBot it instead remains stationary. When next to an object cell the hBot generally stays still. However, the hBot has a probability, determined by its state, of moving away from that cell: 0.01 if it is a common state to both object shapes or it is only achievable for the valid object shape and 0.1 if it is only achievable for the invalid object shape. This increases interaction around the object shapes whilst reducing the chance of stagnation. If the hBots simply remained stationary, they could be divided amongst multiple object shapes or parts of object shapes without enough neighboring agents interacting to distinguish between those object shapes.

3 Determining the Difference Values of Object Shape Pairs

Given a data-chain of an existing object shape it is possible to determine which states at which state-levels are achievable for a group of hBots by examining sub-chains, represented by a double curly bracket, of lengths one, three and five.

Definition 1. *At sub-chain length one the single number in the sub-chain is equal to that of the state of the hBot in contact with the object shape at that same location.*

– $\{\{A\}\}$ *becomes state [A].*

At sub-chain length three a conversion is required complicated. The sub-chain must first be re-ordered and then the relative state of the hBot must be found.

– $\{\{A,B,C\}\}$ *becomes [B][lowest of A and C][highest of A and C] which returns the new current state for the hBot at position B.*

At sub-chain length five two stages of conversion are required. First three, three length sub-chains must be found and then the relationship between these three needs to be found to give the final state.

– $\{\{A,B,C,D,E\}\}$ *is broken into three* $\{\{A,B,C\}\}$, $\{\{B,C,D\}\}$ *and* $\{\{C,D,E\}\}$.
– $\{\{A,B,C\}\}$ *becomes [B][lowest of A or C][highest of A or C] giving state P.*
– $\{\{B,C,D\}\}$ *becomes [C][lowest of B or D][highest of B or D] giving state Q.*
– $\{\{C,D,E\}\}$ *becomes [D][lowest of C or E][highest of C or E] giving state R.*

Now there is a sub-chain $\{\{A,P,Q,R,E\}\}$ *and Q can be resolved.*

– $\{\{P,Q,R\}\}$ *becomes [Q][lowest of P or R][highest of P or R] which gives the final state for the hBot at position C.*

This method is used as it considers the way in which the hBots interact with both each other and the object shapes' boundaries. Sub-chains of lengths, one, three and five, relate to the information that one, three and five hBots could gather. Considering each sub-chain in the data-chain, is equivalent to considering each position that a group of hBots could interact with the boundary of the object shape.

For example the data-chain of object shape ID0, {1,1,2,1,2,1,1,2,1,2}, is examined to find the hBots' achievable states at state-level 1, 2 and 3 when interacting with the object shape, Table 1. These achievable states are compared to those of object shape ID1, {1,1,1,2,2,1,1,2,1,1,3}, in Table 2 where the difference value is found by dividing the number of achievable states at each state-level that are not present in the other object shape by the length of the data-chain. The achievable states which are not possible in the alternate object shape are highlighted bold. This represents the number of positions that a hBot could determine that it is next to one object shape and not the other and the number of cooperating hBots required to do so.

Table 1. Determining the Achievable States for Object Shape ID0

Data-Chain	State-Level 1	State-Level 2	State-Level 3
1	1	[1][1][2] = 5	[5][5][10] = 36
1	1	[1][1][2] = 5	[5][5][10] = 36
2	2	[2][1][1] = 10	[10][5][7] = 104
1	1	[1][2][2] = 7	[7][10][10] = 70
2	2	[2][1][1] = 10	[10][5][7] = 104
1	1	[1][1][2] = 5	[5][5][10] = 36
1	1	[1][1][2] = 5	[5][5][10] = 36
2	2	[2][1][1] = 10	[10][5][7] = 104
1	1	[1][2][2] = 7	[7][10][10] = 70
2	2	[2][1][1] = 10	[10][5][7] = 104

Table 2. Determining the Difference Values of Object Shape ID0 and Object Shape ID1

State-Level 1: Achievable States		State-Level 2: Achievable States		State-Level 3: Achievable States	
ID0	ID1	ID0	ID1	ID0	ID1
1	1	5	**6**	36	**52**
1	1	5	**4**	36	**26**
2	1	10	5	**104**	**32**
1	2	**7**	**11**	**70**	**112**
2	2	10	**11**	**104**	**112**
1	1	5	5	36	**37**
1	1	5	5	36	36
2	2	10	10	**104**	**103**
1	1	**7**	5	**70**	**40**
2	1	10	**6**	**104**	**57**
	3		16		**184**
Difference Values (relative to alternate object shape)					
0/10	1/11	2/10	5/11	6/10	10/11

3.1 Results

The difference value for each combination of object shape pair was determined, excluding object shape ID6. This object shape was not included as it was the only object shape to include a 4 in its data-chain. The results are shown for achievable states at state-level 1, 2 and 3 in Fig. 3, left, centre and right, respectively. From these heat maps it is shown that the higher the state-level the more different the object shapes appear. It also shows that object shape A's difference value from object shape B is not the same as B's from A. This lack of symmetry is due to the indiviual features of the boundary of the object shape that sub-chains describe. If a feature is common to two object shapes that are compared then it cannot be used to distinguish them. If a feature is unique to one object shape that feature can be used to distinguish it from the other but not vice-versa. Therefore, it is possible to have two object shapes where one is easier to distinguish from the other due to the unique features it has. Finally the heat maps show that object shape pairs: ID1 and ID2, ID5 and ID7, ID4 and ID8 are indistinguishable, this is due to them being mirror images of each other.

	ID0	ID1	ID2	ID3	ID4	ID5	ID7	ID8	ID9
ID0	0.00	0.00	0.00	0.40	0.00	0.00	0.00	0.00	0.00
ID1	0.09	0.00	0.00	0.27	0.00	0.00	0.00	0.00	0.09
ID2	0.09	0.00	0.00	0.27	0.00	0.00	0.00	0.00	0.09
ID3	0.25	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.25
ID4	0.17	0.00	0.00	0.17	0.00	0.00	0.00	0.00	0.17
ID5	0.08	0.00	0.00	0.33	0.00	0.00	0.00	0.00	0.08
ID7	0.08	0.00	0.00	0.33	0.00	0.00	0.00	0.00	0.08
ID8	0.17	0.00	0.00	0.17	0.00	0.00	0.00	0.00	0.17
ID9	0.00	0.00	0.00	0.50	0.00	0.00	0.00	0.00	0.00

	ID0	ID1	ID2	ID3	ID4	ID5	ID7	ID8	ID9
ID0	0.00	0.20	0.20	1.00	0.20	0.00	0.00	0.20	0.60
ID1	0.55	0.00	0.00	0.64	0.18	0.09	0.09	0.18	0.36
ID2	0.55	0.00	0.00	0.64	0.18	0.09	0.09	0.18	0.36
ID3	1.00	0.00	0.00	0.00	0.00	0.25	0.25	0.00	0.75
ID4	0.67	0.17	0.17	0.50	0.00	0.33	0.33	0.00	0.67
ID5	0.58	0.25	0.25	0.75	0.42	0.00	0.00	0.42	0.42
ID7	0.58	0.25	0.25	0.75	0.42	0.00	0.00	0.42	0.42
ID8	0.67	0.17	0.17	0.50	0.00	0.33	0.33	0.00	0.67
ID9	0.67	0.17	0.17	0.83	0.50	0.17	0.17	0.50	0.0

	ID0	ID1	ID2	ID3	ID4	ID5	ID7	ID8	ID9
ID0	0.00	0.60	0.60	1.00	1.00	0.60	0.60	1.00	1.00
ID1	0.91	0.00	0.00	0.82	0.82	0.64	0.64	0.82	0.91
ID2	0.91	0.00	0.00	0.82	0.82	0.64	0.64	0.82	0.91
ID3	1.00	0.25	0.25	0.00	0.50	1.00	1.00	0.50	1.00
ID4	1.00	0.67	0.67	0.83	0.00	0.67	0.67	0.00	1.00
ID5	0.92	0.75	0.75	1.00	0.83	0.00	0.00	0.83	0.83
ID7	0.92	0.75	0.75	1.00	0.83	0.00	0.00	0.83	0.83
ID8	1.00	0.67	0.67	0.83	0.00	0.67	0.67	0.00	1.00
ID9	1.00	0.67	0.67	1.00	1.00	0.50	0.50	1.00	0.00

Fig. 3. Heat-maps of difference values of row object shape relative to column object shape calculated for state-level 1 (left), state-level 2 (centre), and state-level 3 (right).

4 Cooperative Object Recognition Task

The arena in which the tests were performed was hexagonal with each of the six sides measuring 21 cells. Within the arena six of each of two types of object shapes were placed as shown in Fig. 4. One of these object shapes was classed valid for removal and the other object shape was classed as invalid. Thirty-seven hBots were placed at the centre of the arena in a tight cluster at time-step zero. Whenever a hBot reached a state that was achievable for only the valid object shape, the object shape it was neighbouring would be removed instantly, as if dissolved. The number of time-steps it took the swarm to remove all of the six valid object shapes was recorded.

All possible pairs of object shapes with four object cells were tested excluding object shapes that mirrored each other and object shape ID6. These object shapes were not included because mirrored object shapes are perceived as identical in this system and object shape ID6 is the only object shape with a number 4 in its data-chain. Currently the system is only capable of distinguishing object shapes with numbers 1, 2 and 3 in their data-chains. Each experiment was run fifty times with both object shapes classed as valid and invalid in turn. Algorithm 1 describes the simulation, hBot and object shape interaction used.

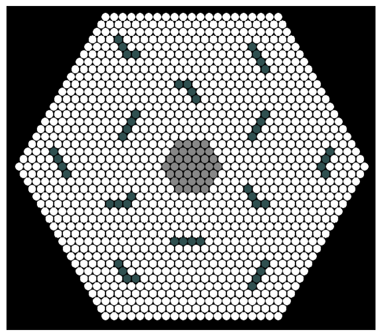


Fig. 4. The simulated arena with six valid object shapes (ID5), six invalid object shapes (ID9) and thirty-seven hBots in the centre.

4.1 Comparison of Swarm Results with the Difference Values

The average number of time-steps it took to complete each of the 30 cooperative object recognition tasks was compared to the mean of the three difference values found for the same two objects, as shown in Fig. 5. These results have a correlation coefficient of -0.78.

By considering only the difference values at each state-level individually an indication of how the number of cooperating hBots affect their ability to distinguish the object shapes. At state-level 1 there was a correlation coefficient of -0.48. However, this is less than the difference values at state-level 2 and state-level 3 which have correlation coefficients of -0.73 and -0.84 respectively. This suggests that the hBots that reach only state-level 1 have a lower effect on whether or not the object shapes will be distinguished from one another.

5 Discussion

A method was proposed for determining the difference value between two object shapes. This was done by comparing sub-chains at lengths one, three and five of the data-chain that represents the object shape. Unlike chain-coding a data-chain

Algorithm 1 The Simulation, hBot and object interaction

```
repeat
    for each hBot i do
        if hBot[i] state == a unique valid shape state then
            object neighboring hBot[i] is deleted
            hBot[i] state = 0
        else if hBot[i] state == a unique invalid state then
            10% chance of hBot[i] state = 0
        else then
            1% chance of hBot[i] state = 0
    end for
    Update contents of all cells
    for each hBot i do
        hBot[i] senses surroundings
        if hBot[i] state == 0 then
            hBot[i] moves to random neighboring cell if empty
    end for
    for each hBot i do
        hBot[i] senses surroundings
    end for
    for each hBot i do
        hBot[i] updates current state
    end for
    Update contents of all cells
    Cells displayed
    Time-steps++
until all six valid object shapes deleted
```

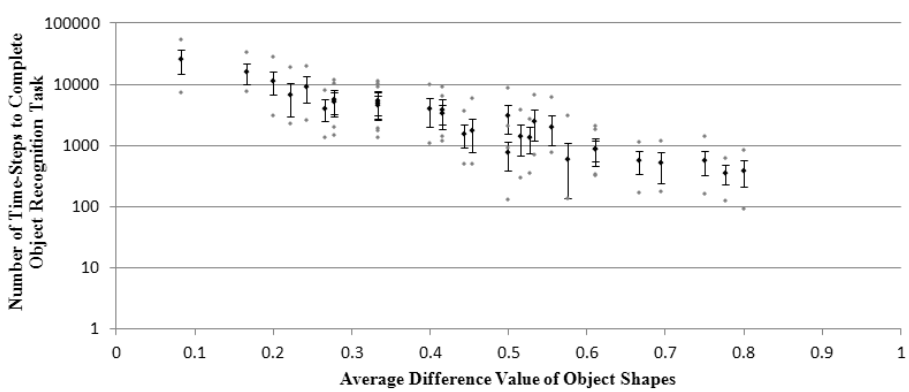


Fig. 5. The average difference value of a pair of object shapes compared to the number of time-steps it took a swarm of hBots to remove the six valid object shapes.

does not have information on its location or orientation which makes it possible to compare the shapes to each other without considering these two factors. To measure the accuracy of the found difference values a series of experiments were run with a range of four cell object shapes. These experiments determined how long it would take a swarm of hBots to remove six valid object shapes whilst ignoring six invalid object shapes.

The results showed a significant correlation between the difference values representative of each of the state-levels with the amount of time-steps it took the hBots to complete the cooperative object recognition task. As expected, where only the states achievable by hBots at state-level 1 are considered in the difference value measurement there was a lower strength correlation. This is likely due to the hBots being generally unable to distinguish between the object shapes on their own, with the exception of object shapes ID0 and ID9 which are the only object shapes without a 3 in their data-chains. Therefore, any other object shapes can be distinguished from them by a state-level 1 hBot in state 3.

This shows that the difference values give a clear indication of how difficult it is for a swarm of hBots to complete a cooperative object recognition task based on the difference between the boundaries of the object shapes. Using this information it will be possible to derive future task scenarios with a suitable range of difficulties for comparing different scenarios and techniques for cooperative object recognition. An area of interest is training the hBots to determine the correct behaviours to distinguish between the object shapes with limited feedback.

Due to the methods chosen for completing this investigation there were some limitations to the simulation platform and limitations in the way in which the agents were modelled, which should be addressed in future research:

- The use of a synchronized grid-world reduced the similarity with physical systems.
- Using a less common hexagonal lattice, comparisons with existing multi-agent systems are difficult to make.
- All the agents were modelled with perfect sensors and communication, it would be interesting to see the effect of miscommunication on the systems capability to identify objects correctly.
- The system only deals with object shapes that have the values 1, 2 and 3 in their data-chain, ignoring possible shapes that have the values 4, 5 and 6, to reduce the state relationships possible.
- The hBots cannot distinguish between hollow shapes and solid shapes, as they only currently interact with the outer boundary of the object shape and have no means of communicating to other hBots inside the hollow if they were there.
- Object shapes that are mirror images of each other appear identical due to the way the hBots sense their surroundings.
- Currently the number of state-levels would only allow a single hBot to have the equivalent knowledge of five state-level 1 hBots. This means that certain shapes currently would not be possible to distinguish from each other.

Overall the difference value metric will be a valuable tool for future studies in co-operative object recognition research with hexagonal grid-world simulations.

References

[1] Haberzettl, C.: Nanomedicine: destination or journey? Nanotechnology **13**(4) (2002) R9

[2] Saha, M.: Nanomedicine: promising tiny machine for the healthcare in future-a review. Oman medical journal **24**(4) (2009) 242

[3] Winfield, A.F., Harper, C.J., Nembrini, J.: Towards the application of swarm intelligence in safety critical systems. In: System Safety, 2006. The 1st Institution of Engineering and Technology International Conference on, IET (2006) 89–95

[4] Cerofolini, G., Amato, P., Masserini, M., Mauri, G.: A surveillance system for early-stage diagnosis of endogenous diseases by swarms of nanobots. Advanced Science Letters **3**(4) (2010) 345–352

[5] Douglas, S.M., Bachelet, I., Church, G.M.: A logic-gated nanorobot for targeted transport of molecular payloads. Science **335**(6070) (2012) 831–834

[6] Şahin, E.: Swarm robotics: From sources of inspiration to domains of application. In: Swarm robotics. Springer (2005) 10–20

[7] Winfield, A.F.: Towards an engineering science of robot foraging. In: Distributed Autonomous Robotic Systems 8. Springer (2009) 185–192

[8] Campo, A., Nouyan, S., Birattari, M., Groß, R., Dorigo, M.: Negotiation of goal direction for cooperative transport. In: Ant Colony Optimization and Swarm Intelligence. Springer (2006) 191–202

[9] Groß, R., Bonani, M., Mondada, F., Dorigo, M.: Autonomous self-assembly in swarm-bots. Robotics, IEEE Transactions on **22**(6) (2006) 1115–1130

[10] El-Faramawy, N., Rangayyan, R., Desautels, J., Alim, O.: Shape factors for analysis of breast tumors in mammograms. In: Electrical and Computer Engineering, 1996. Canadian Conference on. Volume 1., IEEE (1996) 355–358

[11] Rangayyan, R.M., El-Faramawy, N., Desautels, J.L., Alim, O.A.: Measures of acutance and shape for classification of breast tumors. Medical Imaging, IEEE Transactions on **16**(6) (1997) 799–810

[12] Fukuda, T., Ueyama, T., Kawauchi, Y.: Self-organization in cellular robotic system(cebot) for space application with knowledge allocation method. i-SAIRAS'90 (1990) 101–104

[13] Hosokawa, K., Shimoyama, I., Miura, H.: Dynamics of self-assembling systems: Analogy with chemical kinetics. Artificial Life **1**(4) (1994) 413–427

[14] Pamecha, A., Chiang, C.J., Stein, D., Chirikjian, G.: Design and implementation of metamorphic robots. In: Proceedings of the 1996 ASME Design Engineering Technical Conference and Computers in Engineering Conference. Volume 10., Irvine, California, USA: ASME (1996)

[15] White, P., Kopanski, K., Lipson, H.: Stochastic self-reconfigurable cellular robotics. In: Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on. Volume 3., IEEE (2004) 2888–2893

[16] Goldstein, S.C., Campbell, J.D., Mowry, T.C.: Programmable matter. Computer **38**(6) (2005) 99–101

[17] Bishop, J., Burden, S., Klavins, E., Kreisberg, R., Malone, W., Napp, N., Nguyen, T.: Programmable parts: A demonstration of the grammatical approach to self-organization. In: Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on, IEEE (2005) 3684–3691

[18] Ostergaard, E.H., Lund, H.H.: Distributed cluster walk for the atron self-reconfigurable robot. In: The 8th Conference on Intelligent Autonomous Systems (IAS-8)
[19] Østergaard, E.H., Kassow, K., Beck, R., Lund, H.H.: Design of the atron lattice-based self-reconfigurable robot. Autonomous Robots **21**(2) (2006) 165–183
[20] Zykov, V., Mytilinaios, E., Desnoyer, M., Lipson, H.: Evolved and designed self-reproducing modular robotics. Robotics, IEEE Transactions on **23**(2) (2007) 308–319
[21] Gilpin, K., Kotay, K., Rus, D., Vasilescu, I.: Miche: Modular shape formation by self-disassembly. The International Journal of Robotics Research **27**(3-4) (2008) 345–372
[22] Gilpin, K., Rus, D.: A distributed algorithm for 2d shape duplication with smart pebble robots. In: Robotics and Automation (ICRA), 2012 IEEE International Conference on, IEEE (2012) 3285–3292
[23] King, D., Breedon, P.: Towards cooperative robotic swarm recognition: Object classification and validity. Key Engineering Materials **450** (2011) 320–324
[24] King, D., Breedon, P.: Robustness and stagnation of a swarm in a cooperative object recognition task. In: Advances in Swarm Intelligence. Springer (2011) 19–27
[25] Katsaggelos, A.K., Kondi, L.P., Meier, F.W., Ostermann, J., Schuster, G.M.: Mpeg-4 and rate-distortion-based shape-coding techniques. Proceedings of the IEEE **86**(6) (1998) 1126–1154
[26] Zhang, D., Lu, G.: Review of shape representation and description techniques. Pattern recognition **37**(1) (2004) 1–19
[27] Scholten, D.K., Wilson, S.G.: Chain coding with a hexagonal lattice. Pattern Analysis and Machine Intelligence, IEEE Transactions on (5) (1983) 526–533
[28] Nunes, P., Marqués, F., Pereira, F., Gasull, A.: A contour-based approach to binary shape coding using a multiple grid chain code. Signal Processing: Image Communication **15**(7) (2000) 585–599
[29] Park, H., Martin, G.R., Yu, A.C.: Compact representation of contours using directional grid chain code. Signal Processing: Image Communication **23**(2) (2008) 87–100