Vision-only Motion Controller for Omni-directional Mobile Robot Navigation

Fairul Azni Jafar[1], Yuki Tateno[1], Toshitaka Tabata[1],
Kazutaka Yokota[1] and Yasunori Suzuki[2]
[1]Graduate School of Engineering, Utsunomiya University,
[2]Toyota Motor Corporation
Japan

1. Introduction

A major challenge to the widespread deployment of mobile robots is the ability to function autonomously, learning useful models of environmental features, recognizing environmental changes, and adapting the learned models in response to such changes.

Many research studies have been conducted on autonomous mobile robots that move by its own judgment. Generally, in many research studies of autonomous mobile robotics, it is necessary for a mobile robot to know environmental information from sensor(s) in order to navigate effectively. Those kinds of robots are expected for automation in order to give helps and reduce humans work load.

In previous research studies of autonomous mobile robots navigation, accurately control on the robot posture with a possibility of less error, was always required. For that, it is necessary to provide accurate and precise map information to the robot which makes the data become enormous and the application become tedious. However, it is believed that for a robot which does not require any special accuracy, it can still move to the destination like human, even without providing any details map information or precise posture control.

Therefore, in this research study, a robot navigation method based on a generated map and vision information without performing any precise position or orientation control has been proposed where the map is being simplified without any distance information being mentioned.

In this work we present a novel motion controller system for autonomous mobile robot navigation which makes use the environmental visual features capture through a single CCD camera mounted on the robot.

The main objective of this research work is to introduce a new learning visual perception navigation system for mobile robot where the robot is able to navigate successfully towards the target destination without obtaining accurate position or orientation estimation. The robot accomplishes navigation tasks based on information from images captured by the robot.

In the proposed approach, the robot identifies its own position and orientation based on the visual features in the images while moving to the desired position. The study focused on developing a navigation system where the robot will be able to recognize its orientation to

the target destination through the localization process without any additional techniques or sensors required. It is believed that by having this kind of navigation system, it will minimize the cost on developing the robot and reduce burdens on the end-user.

For any robot which is developed for elementary missions such as giving a guide in an indoor environment or delivering objects, a simple navigation system will be good enough. The robot for these tasks does not require any precise position or orientation identification. Instead, qualitative information regarding the robot posture during the navigation task that is sufficient to trigger further actions is needed. As it is not necessary for the robot to know the exact and absolute position, a topological navigation will be the most appropriate solution.

This research work developed a visual perception navigation algorithm where the robot is able to recognize its own position and orientation through robust distinguishing operation using a single vision sensor. When talking about robot, precise and accurate techniques always caught our mind first. However, our research group proved that robots are able to work without precise and accurate techniques provided. Instead, robust and simple learning and recognizing methods will be good enough.

The remainder of this chapter is organized as follows. The next section will explain the topological navigation method employed in the research work. Section 3 gives an overview of the related work. In section 4, the omni-directional robot is introduced. Section 5 explains the environmental visual features used in the approach, section 6 describe the evaluation system of neural network, and section 7 details the motion control system. We conclude with an overview of experimental results (section 8) and a conclusion (section 9).

2. Topological navigation

The proposed navigation method presented here uses a topological representation of the environment, where the robot is to travel long distances, without demanding accurate control of the robot position along the path. Information related to the desired position or any positions that the robot has to get through is acquired from the map.

The proposed topological navigation approach does not require a 3D model of the environment. It presents the advantage of working directly in the sensor space. In this case, the environment is described by a topological graph. Each node corresponds to a description of a place in the environment obtained using sensor data, and a link between two nodes defines the possibility for the robot to move autonomously between two associated positions.

The works presented here require a recording run before the robot navigate in an environment, in order to capture representative images which are associated with the corresponding nodes (places).

An advantage of the proposed approach is that the robot does not have to 'remember' every position (image) along the path between two nodes. During the recording run, the robot will only capture images around the corresponding node. This exercise will help to reduce the data storage capacity.

At the end of the recording run, a topological map of the environment will be build by the robot. Data of visual features obtained from the images of each node are used as instructor data and a neural network (NN) data is produced for each node after trained by NN, before the actual navigation is conducted in the environment.

The topological navigation approach here allows the distance between nodes to be far from each other. The topological map in this approach may have only a few nodes as distance between each node can be quite far. Each node corresponds to a description of a place in the environment obtained using sensor data.

When the robot is moving to the target destination during autonomous run, it will first identify its own position. By knowing its own starting node, the robot will be able to plan the path to move to the target position and obtain the information about the next node from the map. Based on the information, the robot will start moving until it is able to localize that it has arrived at the next node. When the robot find that it is already at the next node, it will then obtain information for the next moving node from the map and start moving toward it. This action will be repeatedly carried out until the robot arrives at the desired destination. An overview of the navigation method is presented in Fig. 1.
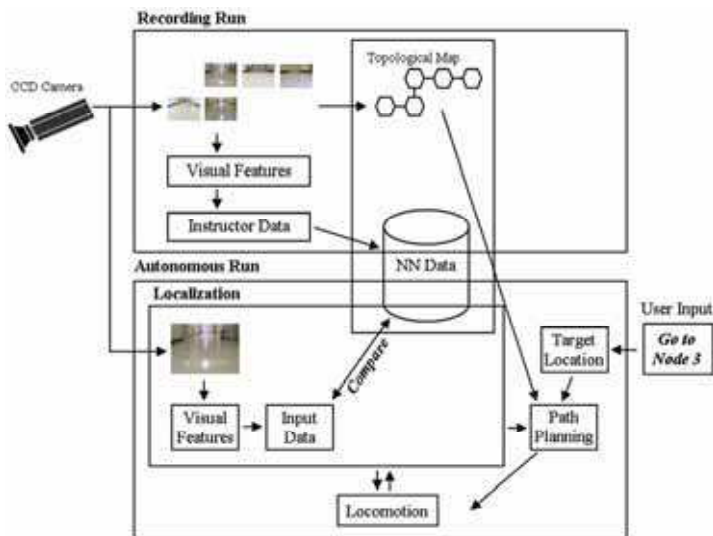

Fig. 1. Overview of the navigation process

In the localization process performed during navigating towards the target destination, the robot compares the extracted visual features from the image captured during the robot movement, with the stored visual features data of the target destination using NN. The NN output result will lead to the knowledge of whether the robot is already arriving near the respective node or not.

In this research study, a robust navigation method where the robot will take advantage of the localization process not only to identify its own position but also the orientation is proposed. By applying this method on the robot, it is believed that the robot is able to correct its own pose and navigate towards the target destination without loosing the direction while moving to the target destination.

2.1 Map information

The representations of large-scale spaces that are used by humans seem to have a topological flavour rather than a geometric one (Park & Kender, 1995). For example, when

providing directions to someone in a building, directions are usually of the form "go straight to the hall, turn left at the first corner, use the staircase on your right," rather than in geometric form.

When using a topological map, the robot's environment is represented as an adjacency graph in which nodes represent places (that the robot needs to identify in the environment) and arcs stand for the connections between places. A link (show by the arc) on the topological map means that the robot can successfully travel between the two places (landmarks). A link can only be added to the map if the robot has made the corresponding journey. In some instances the links are established at the same time as the places are identified. Evidence has been provided that localization based on topological map, which recognizes certain spots in the environment, is sufficient to navigate a mobile robot through an environment.

The use of topological maps (graphs) has been exploited by many robotic systems to represent the environment. A compact topological map with fewer locations requires less memory and allow for faster localization and path planning. Topological representations avoid the potentially massive storage costs associated with metric representations. In order to reach the final goal, the navigation problem is decomposed into a succession of sub goals that can be identified by recognizable landmarks. An example of topological map is shown in Fig. 2.
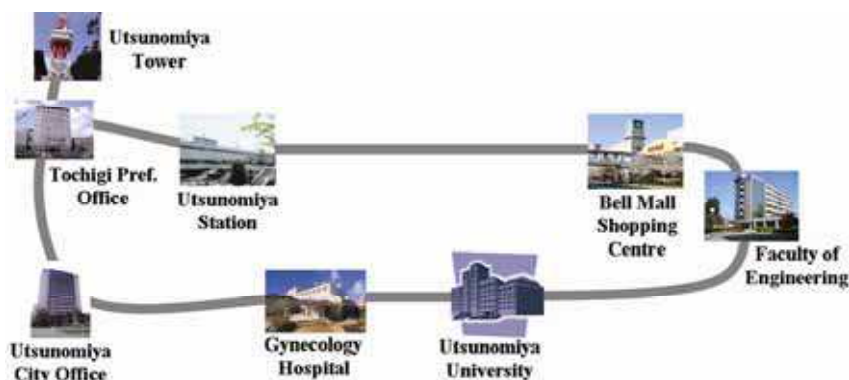

Fig. 2. A topological map of landmarks in the city of Utsunomiya, Japan

3. Related work

An intelligent robot navigation system requires not only a good localization system but also a dexterous technique to navigate in environment. The mobile robot has to be able to determine its own position and orientation while moving in an environment in order for the robot to follow the correct path towards the goal. Most current techniques are based on complex mathematical equations and models of the working environment, however following a predetermined path may not require a complicated solution.

Applications of additional sensors to control robot posture have been widely introduced (Morales et al., 2009). Conversely, in the approach introduced in this research work, there is no application of additional sensor to control the orientation of the robot. The only sensor that is used in the robot system is vision sensor. Many researchers presented methods of

controlling robot pose by combining the problem with navigation towards the goal using only vision sensor, where visual servoing is most popular.

In a research work introduced by Vasallo et al. (Vasallo et al., 1998), the robot is able to correct its own orientation and position through a vanishing point calculated from the corridor guidelines where the robot is navigating. The method allows the robot to navigate along the centre of the corridor, where the progress along the path (links) is monitored using a set of reference images. These images are captured at selected positions along the path. The robot will be able to determine its own position through a comparison between the acquired image and the reference image set. They used SSD (sum of squared differences metric) method to perform the comparison. In these research works, they separated the task on detecting position and orientation. In addition, the methods require tedious techniques for both localization and navigation tasks in order to control the robot precisely from losing the actual direction.

Mariottini et al. (Mariottini et al., 2004) employed a visual servoing strategy for holonomic mobile robot based on epipolar geometry retrieved from current and desired image grabbed with the on-board omnidirectional camera to control the pose of the robot during navigation. Their servoing law is divided in two independent steps, dealing with the compensation, respectively, of rotation and translation occurring between the actual and the desired views. In the first step a set of bi-osculating conics, or bi-conics for short, is controlled in order to gain the same orientation between the two views. In the second step epipoles and corresponding feature points are employed to execute the translational step necessary to reach the target position.

There is a method introduced by Goedeme et al. (Geodome et al., 2005) which is comparable to the proposed approach. They developed an algorithm for sparse visual path following where visual homing operations are used. In this research work, they focus on letting the robot to re-execute a path that is defined as a sequence of connected images. An epipolar geometry estimation method is used to extract the position and orientation of the target during navigating towards the target point. The epipolar geometry calculation is done based on visual features found by wide baseline matching. Information obtained from the epipolar geometry calculation enables the robot to construct a local map containing the feature world positions, and to compute the initial homing vector. However, the method of using epipolar geometry requires difficult computation tasks.

In contrast to the approach introduced by Geodome the autonomous run operation in the proposed navigation method in this research work is straightforward and does not comprise any complicated tasks. In the work done by Geodome, the robot needs to perform an initialization phase in order to calculate the epipolar geometry between starting position and the target position. The robot will have to extract a trackable feature to be use as a reference during driving in the direction of the homing vector. In the tracking phase performed after the initialization phase, the feature positions of a new image are tracked, and thus the robot position in the general coordinate system is identified through the epipolar geometry measurements. The tasks perform by the robot in this approach are rather complicated. Dissimilar to this method, the robot in the proposed approach identifies its own starting position and immediately moves towards the target destination. All the identification works are done directly based on the environmental visual features explained in section 5. Moreover, the robot in the proposed approach does not have to measure the distance to move to the following place.

In spite of that, all the studies introduced in the literature have been a very successful achievement. However, they require complicated techniques to let the robot identify its own position and to control the robot from losing the actual direction during navigating towards the target destination. In the proposed approach here, a simple yet robust technique for both robot localization and navigation is developed with the main objective is to let the robot arrive safely at the proximity of the target destination. An emphasis is put on situations where robot does not require accurate localization or precise path planning but is expected to navigate successfully towards the destination. The approach also does not aim for accurate route tracing.

With this consideration, a robust navigation method for autonomous mobile robot where the robot will take advantage of the localization process not only to identify its own position but also the orientation is proposed. It is believed that both robot position and orientation can be identified through a single technique, and it is not necessary to have separate technique to control the robot orientation. A method with less technique required for identifying both position and orientation, will reduce the robot navigation time.

The proposed method is based on environmental visual features evaluated by neural network. By applying this method on the robot, it is believed that the robot is able to correct its own pose and navigate towards the target destination without losing the direction while moving to the target destination.

In a simple navigation method which does not require for any precise or accurate techniques as introduced in this research study, the robot programming will become less tedious. The method that is introduced in this research work will able to help minimizing cost in robot development.

An earlier contribution in the area of mobile robot navigation using image features and NN is the work of Pomerleau (Pomerleau, 1989)}. It was then followed by many other research works where they have successively let the robot navigate in the human working environments (Burrascano et al., 2002; Meng & Kak, 1993; Na & Oh, 2004; Rizzi et al., 2002).

One very similar research work with the proposed method is the one introduced by Rizzi et al. (Rizzi et al., 2002)}. The robot in this approach uses an omnidirectional image sensor to grab visual information from the environment and applies an artificial NN to learn the information along the path. The visual information, which composed of RGB color values, is preprocessed and compacted in monodimensional sequences called *Horizon Vectors* (HV), and the path is coded and learned as a sequence of HVs. The system in this approach guides the robot back along the path using the NN position estimation and orientation correction. The orientation identification is performed by a circular correlation on the HVs.

4. The omni-directional mobile robot

The autonomous mobile robot system in this research study is based on the Zen360 omni-directional mobile robot which was developed by RIKEN Research Centre (Asama et al., 1996). The driven mechanism part of the robot was developed by the previous members of the Robotics and Measurement Engineering Laboratory of Utsunomiya University. The robot consists of a computer and a CCD colour camera. Each image acquired by the system has a resolution of 320 x 240. The entire system is shown in Fig. 3.
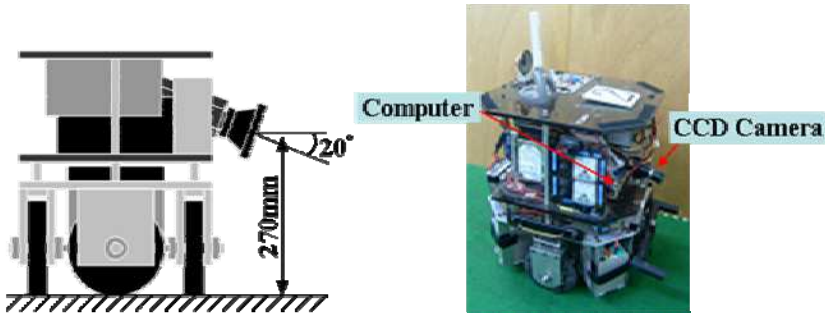
Fig. 3. The Omni-directional mobile robot

Model	Sony XC999
Heel Angle	106.30°
Tilt Angle	79.18°
Resolution	320 X 240 [pixel]

Table 1. Camera specification

The robot has 4 wheels where the wheel diameter is 360mm on the circumference of circle with the rotating shaft is pointing towards the centre. The robot mechanism possesses 2 translatory DOF and 1 rotating DOF, which in total of 3 DOF (Degree of Freedom).

During the actual locomotion, the opposing 2 wheels are rotating on the same translation direction derived through the activation of the 2 wheels bearing, therefore a stabilize translation motion towards x axis or y axis is feasible. Furthermore, a turning motion is also feasible due to the same direction driven of the 4 wheels.
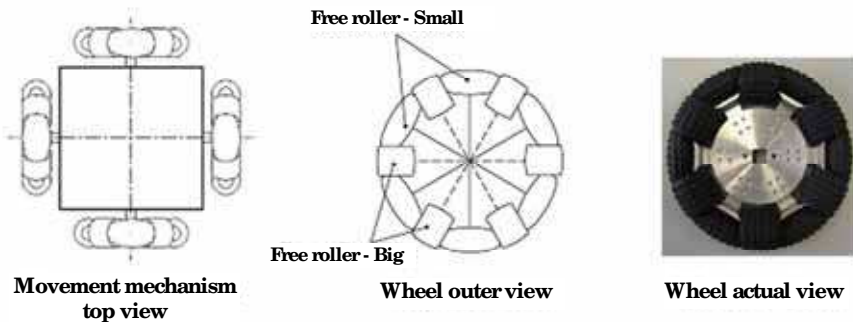
Fig. 4. Robot wheel with free rollers

Under ordinary circumstances, the arranged wheels become resistance to the travelling direction and the vertical direction. However, there are free rollers fixed on the wheel periphery of all the wheels as shown in Fig. 4, where it reduced the resistance on the wheel rotating direction and allowed for a smooth movement. Each wheel is built up from 6 big free rollers and 6 small free rollers. The two type of free rollers are arranged 30[deg] in angle alternately, and the free roller outer envelope form the outer shape of the wheel. With this

condition, the robot is able to perform feasible omni-directional locomotion where it can move in all directions without affecting its posture.

5. Features extraction

Feature extraction is a process that begins with feature selection. The selected features will be the major factor that determines the complexity and success of the analysis and pattern classification process. Initially, the features are selected based on the application requirements and the developer's experience. After the features have been analyzed, with attention to the application, the developer may gain insight into the application's needs which will lead to another iteration of feature selection, extraction, and analysis. When selecting features, an important factor is the robustness of the features. A feature is robust if it will provide consistent results across the entire application domain.

5.1 Colour features

The concept of using colour histograms as a method of matching two images was pioneered by Swain and Ballard (Swain & Ballard, 1991). A number of research works also use colour histogram in their method for robot localization (Kawabe et al., 2006; Rizzi & Cassinis, 2001; Ulrich & Nourbakhsh, 2000). These research works previously verified that colour can be use as the features in mobile robot localization. However, unlike those studies, we examine all the colours in an image, and use their dispositions, rather than histograms, as features.

In examining the colour, CIE XYZ colour scheme is use as it can be indicated in x and y numerical value model. CIE XYZ considered the tristimulus values for red, green, and blue to be undesirable for creating a standardized colour model. They used a mathematical formula to convert the RGB data to a system that uses only positive integers as values.

In the proposed method, the CIE chromaticity diagram is separate into 8 colours with a non-colouring space at the centre. It is regarded that those colours which are located in the same partition as the identical colour. The separation of chromaticity diagram is shown in Fig. 5.
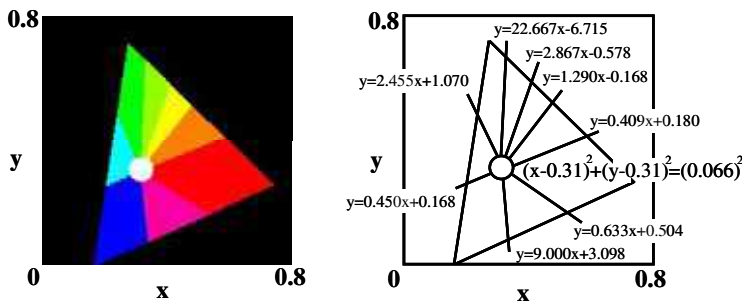

Fig. 5. Separating the chromaticity diagram into 8 colour partitions

Colours in an image are considered black when the lightness of the colour is low. It is necessary to take these circumstances into consideration when applying colour features. Therefore, luminosity L is applied to classify between black and the primary colour of each partition in the separated chromaticity diagram. L is presented as follow;

$$L = 0.299R + 0.597G + 0.114B \qquad (1)$$

We also use the luminosity to classify the non-coloring space into white, grey and black color as we learned that the degree in lightness in the non-coloring space is changing gradually from white to black through grey color.

$L\geq50$ is classify as white and black for $L<35$ at the non-coloring space of $x=0.31$ and $y=0.31$. Grey color is considered when $50>L>35$. For other partitions, black color is considered when $L\leq10$. The value for L in this approach is determined empirically.

Formulating a simple way using the details of colour information, by roughly separating the colours into 11 classifications as explained above is considered.

Pixels whose colours fall into one colour domain are considered to have the same colour. Through this, the colour disposition in a captured image is evaluated based on area ratio in the entire image and coordinates of the centre of area (x and y coordinates) of each 10 colours in the entire image, which means total of 30 data of visual features can be acquire from colour features.

5.2 Shape features

A shape is made when the two ends of a line meet. Some shapes have *curved* lines while other shapes may have *straight* lines. For example, a wheel has a shape made from a curved line. One kind of shape is called *geometric*. Geometric shapes are simple shapes that can be drawn with straight lines and curves, for example a desk. The work in this research study is dealing with lines and points which are connected through the lines, in term of shape feature.

Edge detection methods are used as a first step in the line detection process. Edge detection is also used to find complex object boundaries by marking potential edge points corresponding to places in an image where rapid changes in brightness occur. After these edge points have been marked, they can be merged to form lines and object outlines.

In the proposed approach, the edge extraction is carried out based on a simple gradient operation using Robert operator. The original colour image is converted to gray level and then normalized so that the range of gray value is within [0,255]. From this, lines in an image can be extracted through the edge extraction process.
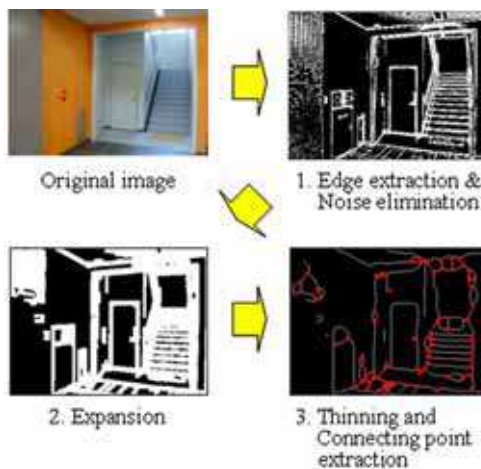

Fig. 6. Extracting lines and connecting points

When further image processing done on the extracted edges, points which are connected through the lines can be extracted and together with the extracted lines, they can be use as information to recognize a place. The processes on extracting the points include noise elimination, expansion and finish up with a thinning process. These processes can be seen in Fig. 6. From the extracted connecting points and lines, it is able to acquire 2 data of visual features, which consist;

- Ratio of the lines in the entire image
- Ratio of the connecting points in the entire image

5.3 Visual features data

An example of colour separation work and shape (lines and connecting points) extraction work which is done on an image is shown in Fig. 7.

Original image   Color separation   Shape extraction
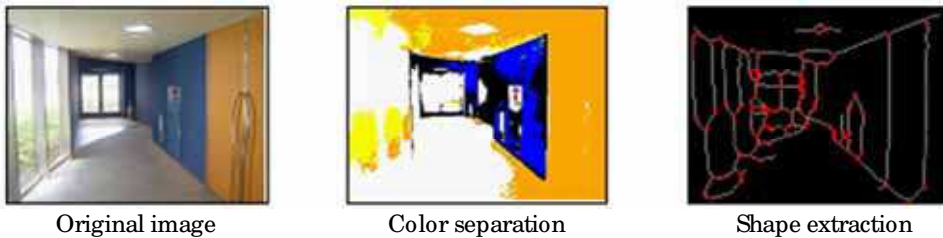
Fig. 7. Example of colour separation work and shape extraction work carried out on an image

Based on the extraction process which explained earlier, each captured image (either for the database data or for the position identification process) will produce a set of visual features data after going through the features extraction processes. The visual features data possess a total of 35 data where 33 data are from the colour features and another 2 data are from the shape features data. The data set can be seen in Table 2. C in Table 2 is colour area ratio, x and y are the coordinates of the colour area ratio, while S is representing the shape features.

C[0]	C[1]	C[2]	C[3]	C[4]	C[5]	C[6]	C[7]	C[8]	C[9]
$x[0]$	$x[1]$	$x[2]$	$x[3]$	$x[4]$	$x[5]$	$x[6]$	$x[7]$	$x[8]$	$x[9]$
$y[0]$	$y[1]$	$y[2]$	$y[3]$	$y[4]$	$y[5]$	$y[6]$	$y[7]$	$y[8]$	$y[9]$
S[1]	S[2]	S[3]	S[4]	S[5]	S[6]	S[7]	S[8]	S[9]	S[10]
S[11]									

Table 2. Architecture of the visual features data

6. Evaluation system

Neural networks (NN) are about *associative memory* or *content-addressable* memory. If content is given to the network, then an address or identification will be return back. Images of object could be storage in the network. When image of an object is shown to the network, it will return the name of the object, or some other identification, e.g. the shape of the object.

A NN is a massive system of parallel-distributed processing elements (neurons) connected in a graph topology. They consist of a network of processors that operate in parallel. This means they will operate very fast. To date, the most complex NN that operate in parallel consist of a few hundred neurons. But the technology is evolving fast. To recognize images, one needs about one processor per pixel. The processors, also called neurons, are very simple, so they can be kept small.

NN will learn to associate a given output with a given input by adapting its weight. The weight adaptation algorithm considered here is the steepest descent algorithm to minimize a nonlinear function. For NN, this is called *backpropagation* and was made popular in (Mcclelland et al., 1986; Rumelhart et al., 1986). One of the advantages of the backpropagation algorithm, when implemented in parallel, is that it only uses the communication channels already used for the operation of the network itself. The algorithm presented incorporates a minimal amount of tuning parameters so that it can be used in most practical problems. Backpropagation is really a simple algorithm, and it has been used in several forms well before it was invented.

NN is recommended for intelligent control as a part of well known structures with adaptive critic. Recently, much research has been done on applications of NNs for control of nonlinear dynamic processes. These works are supported by two of the most important capabilities of NNs; their ability to learn and their good performance for the approximation of nonlinear functions. At present, most of the works on system control using NNs are based on multilayer feedforward neural networks with backpropagation learning or more efficient variations of this algorithm.
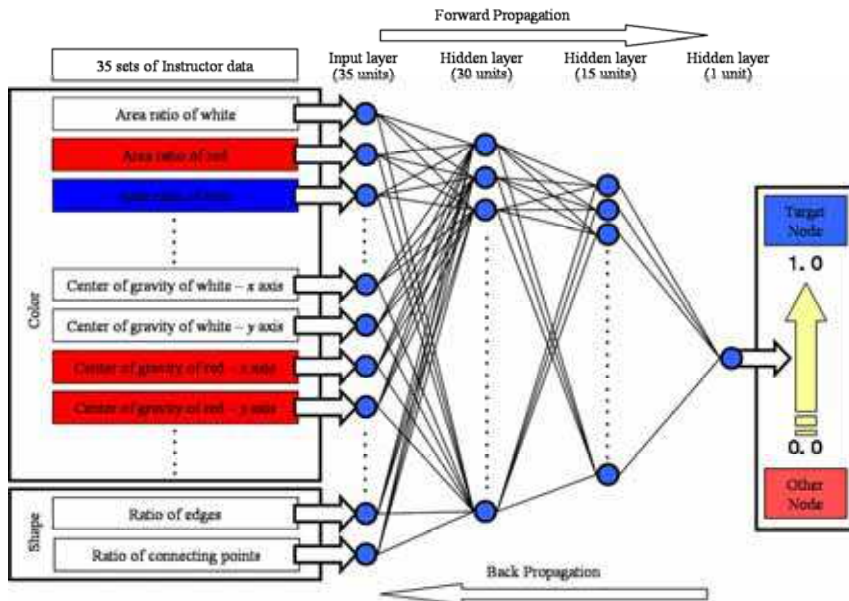

Fig. 8. Multilayer perceptron neural network

Essentially, NN deals with cognitive tasks such as learning, adaptation, generalization and optimization. NN improve the learning and adaptation capabilities related to variations in

the environment where information is qualitative, inaccurate, uncertain or incomplete. The processing of imprecise or noisy data by the NN is more efficient than classical techniques because NN are highly tolerant to noises.

Furthermore, evaluation using NN will make recognition process robust and decreases the computational cost and time. NN offer a number of advantages, including requiring less formal statistical training and the availability of multiple training algorithms. With this consideration, NN has been choose in the proposed approach as the computational tool to evaluate the similarity between images captured during learning and localization phase.

In this work, a stratum type of multilayer perceptron NN as shown in Fig. 8 is used as the computation tool. The 4 layers NN is used where the number of input layer depends on the number of combination data between colour features and shape features, and the unit number of each middle layer is depends on the number of input data.

6.1 Instructor data

In order to let the robot in the proposed approach achieve a sufficiently cursory and stable recognition during the navigation, if the robot arrives in the proximity around the centre of the memorized node, the robot must be able to identify that it has reached the node. It is important to provide each node with a domain of area which can be identified by the robot.

For that, a set of the instructor data is need to be provided with consideration that the robot will be able to localize itself within a certain area around each place, rather than the exact centre of the place. The robot is assign to capture few images around the centre of the node during the recording run. From the captured images, visual features data are extracted.

Data taken at all nodes along the expected path is used as the instructor data for the NN. The back-propagation learning rules are then used to find the set of weight values that will cause the output from the NN to match the actual target values as closely as possible. Standard back-propagation uses steepest gradient descent technique to minimize the sum-of squared error over instructor data.

7. Motion control system design

To navigate along the topological map, we still have to define a suitable algorithm for the position and orientation recognition in order to let the robot move on the true expected path. This is mainly due to the inconsistency of the floor surface flatness level of where the robot is navigating, that might cause to large error in robot displacement and orientation after travelling for a certain distance.

In this work paper, we introduce a new method for mobile robot navigation, where the robot recognizes its own orientation on every step of movement through NN, using the same method for position recognition. In the proposed method, NN data for orientation recognition is prepared separately with NN data for the position recognition. By separating the NN into 2 functions, the width of the domain area for position recognition can be organized without giving influence to the orientation recognition. Furthermore, it is believed that through this, extensively efficient orientation recognition is achievable.

As mentioned earlier in is paper, in the proposed approach the robot need to conduct a recording run first to capture images for instructor data collection. The robot will be brought

to the environment where it will have to navigate. At each selected node, the robot will allow to capture images to prepare instructor data and NN data for both position and orientation recognition. Topological map of the environment is prepared as soon as the recording run is completed.

Next, when the robot performs autonomous run in the environment, visual features extracted from the captured image is fed through the NN for position recognition first. This recognition operation will be based on the NN data prepared for the position recognition. If the result shows that the robot is not arriving near the target node yet, the orientation recognition process will be executed. The robot will apply the same visual features to each NN data of 5 different directions of the target node. This process is illustrated in Fig. 9. The robot will need to capture only 1 image at each movement which will help to improve the robot navigation time.
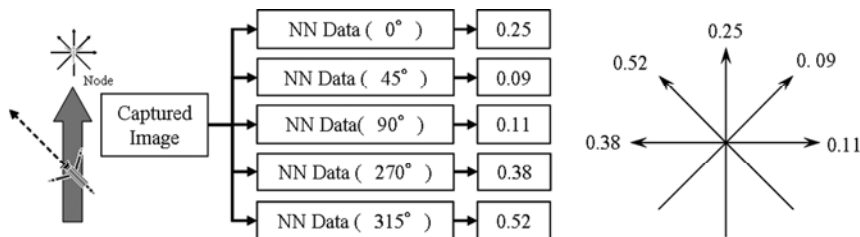
Fig. 9. NN evaluation against NN data of the 5 different directions

From the results produced by the 5 NN data, the robot will be able to determine its current progress direction by computing the median point of the results. An example of this evaluation method is demonstrated by Fig. 10. After the current progress direction is understood, the robot will calculate the difference in angle between target node and current progress direction, and correct its direction towards the target node. The same procedure is repeated until the robot able to recognize that it is within the domain of the target node and stop. This exercise will help keeping the robot on the true expected path. Through this, the robot will able to move on the path to the target destination. Fig. 11 presents an algorithm of the robot movement from TNm (target node of m order) to TNn including the orientation recognition process.

7.1 Positioning control

At every step of the robot movement, the robot will perform a self-localization task first in order to determine whether it has reached the target node or not. In order to perform the self-localization task, the robot can simply capture one image and feed the extracted visual features into the NN. Based on our empirical studies, we found out that the NN output is gradually increasing when the robot is approaching towards the target node.

The work in this research study does not put aim on letting the robot to perform a precise localization task. Therefore, when preparing the instructor data for NN of each node, a fact that the robot will be able to localize itself within a certain area around each node, rather than the exact centre of the node, need to be considered. This is necessary to ensure that the robot will not stop rather far from the centre of the node. Thus it will help the robot to avoid such problems like turning earlier than expected at a corner place which might cause a crash to the wall.
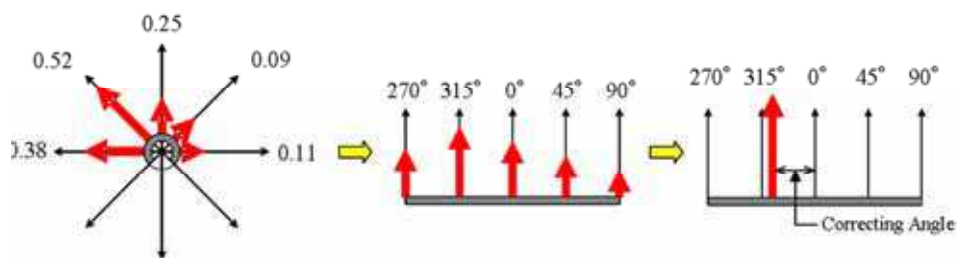
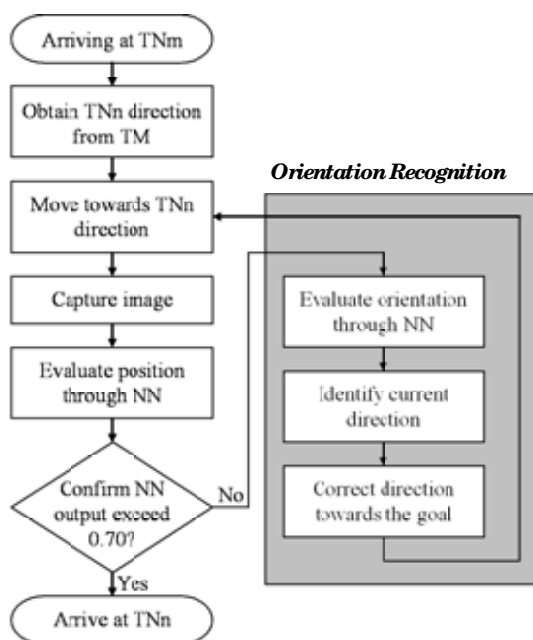Fig. 10. Evaluation method for current progress direction

Fig. 11. Navigation algorithm consists of robot movement from a TNm to TNn

To be able to get a sufficiently robust localization output, the robot is arranged to capture 3 images around the centre of each corresponding nodes. 1 image is captured at the centre of the node and the other 2 images are about 5mm to the left and right from the centre, as shown in Fig. 12 (a). We believe that the domain area of the node which help the robot to robustly identify the node, will be constructed in this way.

It is important to evaluate the ability of the robot on recognizing its own position (localization) at around the specified nodes, before the robot is put to navigate in the real environment. Tests were conducted at 3 places (nodes) in a corridor environment to see how a domain of area for localization exists. The robot is moved manually every 5 cm between -80 and 80 cm on the X, Y, Zr and Zl axis of the place midpoint as shown in Fig. 12 (b). The robot is also rotated by 5 degree in an angle range between -60 and 60 degree as depicted in Fig. 12 (c). The result is presented in Fig. 13.
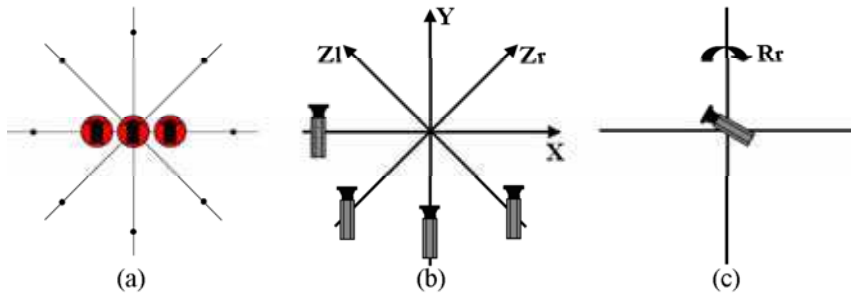
Fig. 12. (a) Positions where the image of instructor data is captured. (b)(c) Robot moves along each X, Y Zr, Zl and R axis of the place midpoint to evaluate the domain around the place
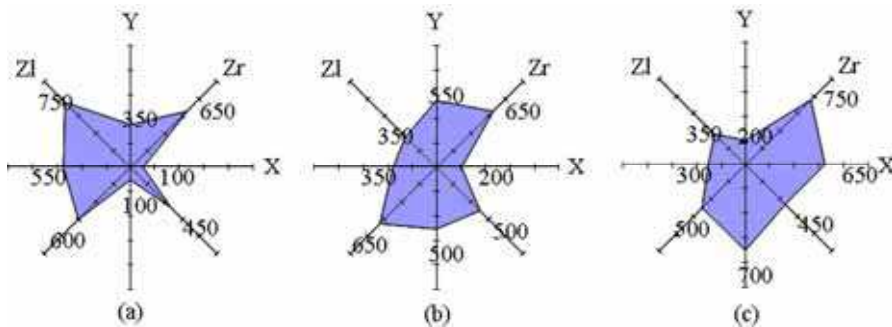
Fig. 13. Domain area acquired at the 3 places; (a) Place 1, (b) Place 2, (c) Place 3

7.2 Localization assessment

A manual localization experiment was performed in order to see the ability of the robot to recognize its own position and thus localize the selected nodes. The experiment took place at the 3rd floor corridor environment of the Mechanical Engineering Department building in our university (Fig. 14). In this experiment, the localization is carried out against all the 3 nodes.
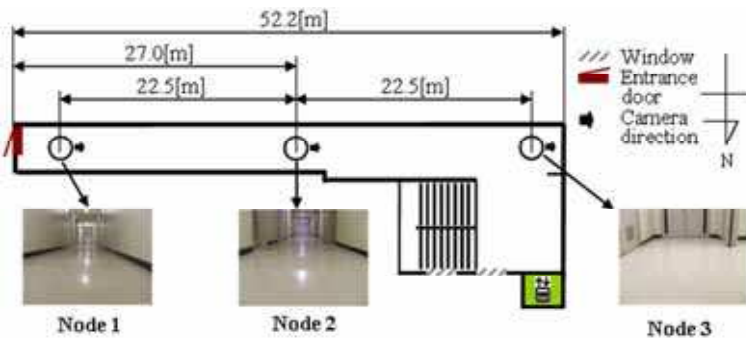
Fig. 14. Experiment layout of the corridor environment with the selected 3 nodes

After the instructor data is fed into NN for NN data preparation, the robot was brought back to the corridor and manually moved to capture image at every 15mm from a starting position which is about 150cm away from the Node 1 (see Fig. 14) until the last position which is about 150cm ahead of Node 3. All the captured images are tested against NN data of each 3 nodes.

The result of the localization experiment is shown in Fig. 15 (the graph with blue color). A localization is considered achieved when NN output is higher than 0.7 (red line in the result graph).
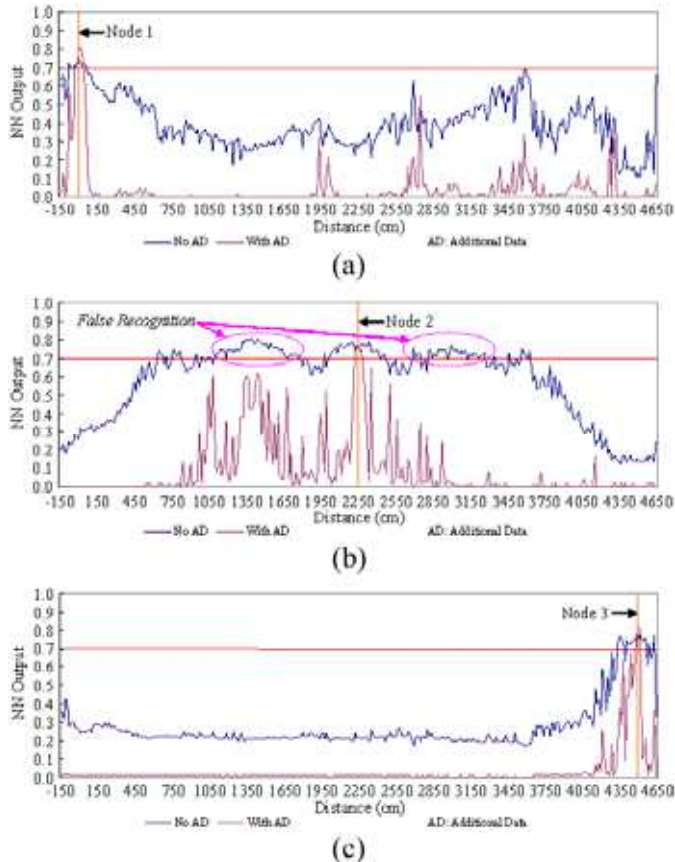
Fig. 15. Result of the test images localization test against NN data of each node; (a) Output from Node 1 NN, (b) Output from Node 2 NN, (c) Output from Node 3 NN

There are some errors occurred in the result against NN data of Node 2, where images around distance of 1200 ~ 1800cm mistakenly responded to the NN of Node 2 (Fig. 15 (b)). The same phenomenon can be seen at the distance of about 3000 ~ 3600cm from the starting point.

The set of instructor data described in previous section does not result in any particular width, depth, and shape of the domain area for position recognition. Even though we do not

aim at the exact center of the node, it is desirable to have a way to somewhat control the size and shape of the domain area so that it is more compact and the robot will be able to converge and stop as much nearer as possible to the center of the node. In fact, the unnecessary mis-recognitions as occurred in the localization result against NN data of Node 2 (Fig. 15 (b)), should be prevented. The idea is to add few instructor data to the current method of preparation, which is believed will reduce the NN output just outside the desired area.

After some preliminary tests, result of the tests indicated that if 2 instructor data of images taken at distanced points of front and back from the center of the node, is added, and trained the NN to output 0.1 for them, the localization domain of the node will be less smaller (Fig. 16). However, distance of the additional data is different for each node to obtain the best domain area for localization. It should be at 90cm for Node 1, 100cm for Node 2 and 30 cm for Node 3. Localization results against NN data of the new method of instructor data is shown in purple graph of Fig. 15.
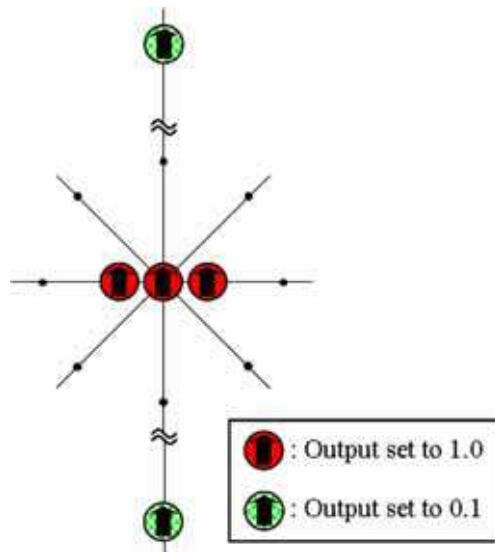
Fig. 16. New method of acquiring instructor data; additional data whose output is set to 0.1 assigned at distanced points of front and back from the centre of the node

As a conclusion to this result, the width, depth and shape of the domain area might be suffering by influences from the environment condition. A different method for preparing the instructor data (with unfixed distance of additional data which trained to output 0.1) might be necessary for different nodes even in the same environment. The shape and the width of the domain area are not identical on every places. They vary and depend much on the environment condition. For example when setting a place at a corner in the corridor, it is necessary to put a consideration on the width of the domain area in order to avoid a problem where the robot turns earlier and crashes into the wall. Furthermore, to let a robot navigate in a narrow environment, it may be necessary to provide a smaller domain area of each place.

7.3 Orientation control

When the robot is navigating through the environment, it is important for the robot to know its own moving direction in order to move on the true expected path. By knowing its own moving direction, the robot will be able to correct the direction towards the target destination.

In the approach which is proposed in this research study, visual features from the image captured at each step of the robot movement will be fed into the 5 NN data of different directions of the target destination (node). The visual features are the same features used for the position recognition. From the output results of the 5 NNs, the robot will be able to determine its current moving direction.

To prepare the instructor data and NN data for the orientation recognition, the robot is arranged to capture 5 images as shown in Fig. 17; 1 image at the centre and 1 image each to forward and backward from the centre along the x and y axis. For each node, the instructor data images are captured on 5 different directions during the recording run phase, as depicted in Fig. 18. A set of instructor data, which consist of visual features extracted from the images is prepared for each direction. The instructor data comprises 5 sets of features data of the direction whose output is set to 1, and 1 set of features data from each other directions whose output is set to 0. The instructor data are then going through a learning process in the NN to obtain NN data. 5 sets of NN data in total are prepared for each node in an environment.
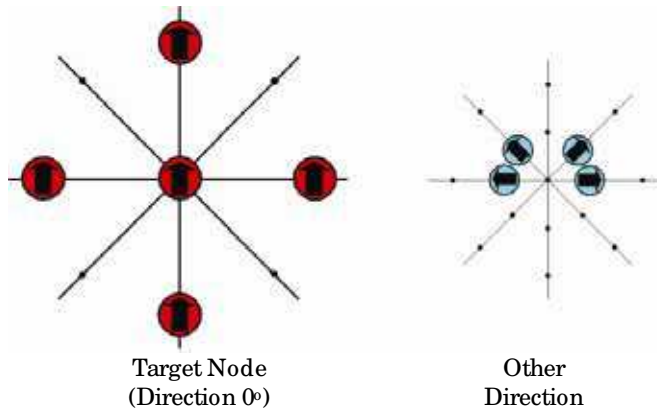

Target Node
(Direction 0º)

Other
Direction

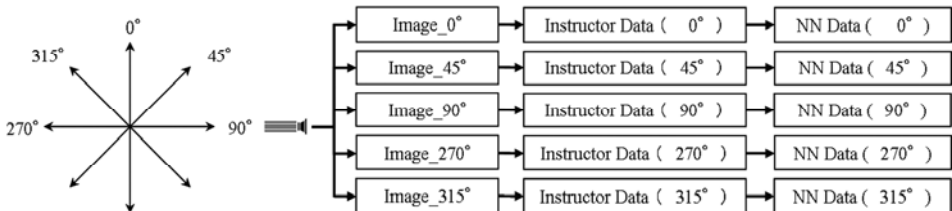Fig. 17. Positions where images for instructor data are captured for orientation recognition

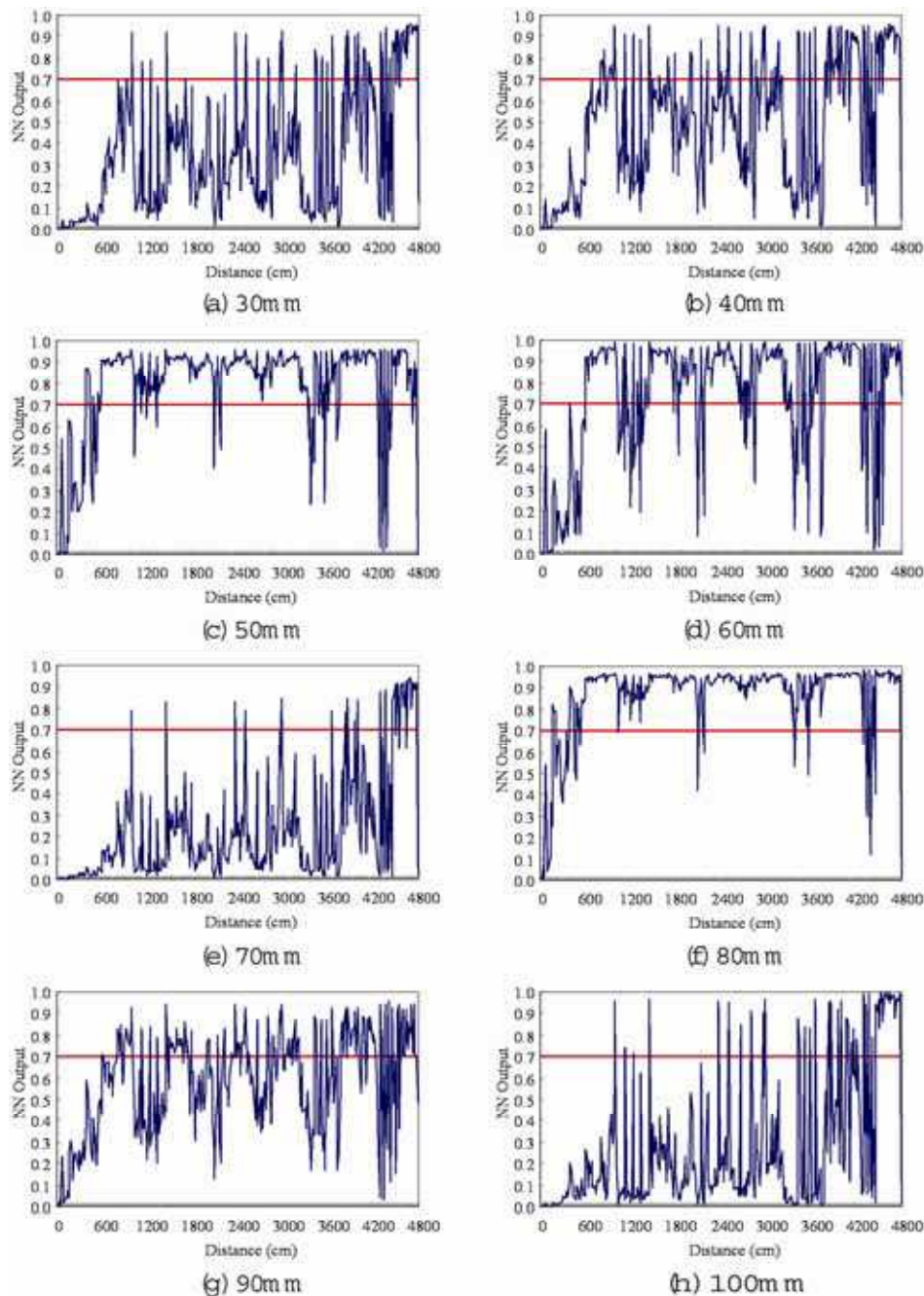Fig. 18. Preparing the instructor data for orientation recognition of 5 different directions

Fig. 19. Results of recognition capability against NN for orientation recognition in which distance for images captured at x axis is fixed

In order to recognize the finest distance for the 4 images on the x and y axis, a series of tests have been conducted at the Node 3 of the corridor environment (see Fig. 14). The test has been divided into two stages where in the first stage, 2 images of x axis are fixed to the distance of 30mm from the centre. As for the y axis, the 2 images are captured on every 10mm between distances of 30mm to 100mm, which means 8 sets of instructor data are prepared. After the instructor data is fed into NN for NN data preparation, a set of test images have been captured at every 15mm from a starting position which is about 4650cm far from the node. The output results of the test images which are tested against each NN data are shown in Fig. 19.

The results show that NN data which consists of instructor data images captured at 80mm from the centre of the node on the y axis produced the best recognition result. The test images output is mostly constant above 0.7 from a distance of about 4000cm, with few failure recognitions happened. With this condition, it is believed that mobile robot will be able to determine the current progress direction from as far as 40m from the centre of the node.

Next, in the second stage of the test, the 2 images on the y axis have been fixed to the distance of 80mm from the centre of the node. This is appropriate to the result of the first stage test. Images have been captured at 4 distances of 5mm, 10mm, 15mm and 30mm on x axis. 4 sets of instructor data and NN data are prepared. Using the same set of test images from the first stage test, a series of tests have been conducted against the 3 sets of NN data. The results which presented in Fig. 20 show that NN data with instructor data images captured at the distance of 30mm produced the finest result.
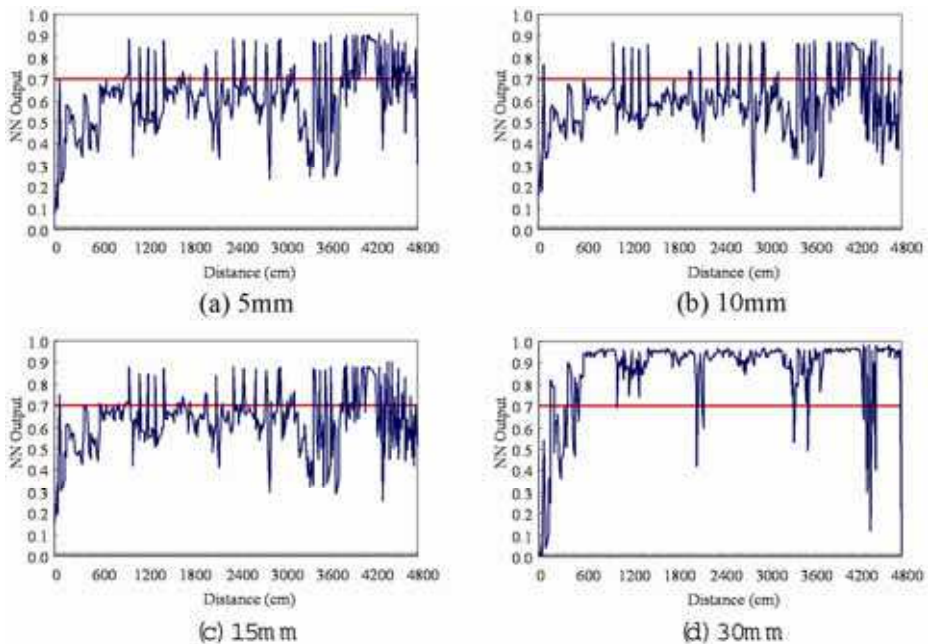
Fig. 20. Results of recognition capability against NN for orientation recognition in which distance for images captured at y axis is fixed

8. Navigation experiments

Navigation experiments have been scheduled in two different environments; the 3rd floor corridor environment and the 1st floor hall environment of the Mechanical Engineering Department building. The layout of the corridor environment can be seen in Fig. 14 and for the hall environment, the layout and representative images is presented in Fig. 21.

The corridor has been prepared with a total of 3 nodes separated from each other about 22.5m. The total length of the corridor is about 52.2m with 1.74m in width. Meanwhile, in the hall environment, 5 nodes have been arranged. Distance between each node is vary, where the longest distance is between node 2 and node 3 which is about 4 meter as shown in Fig. 21.
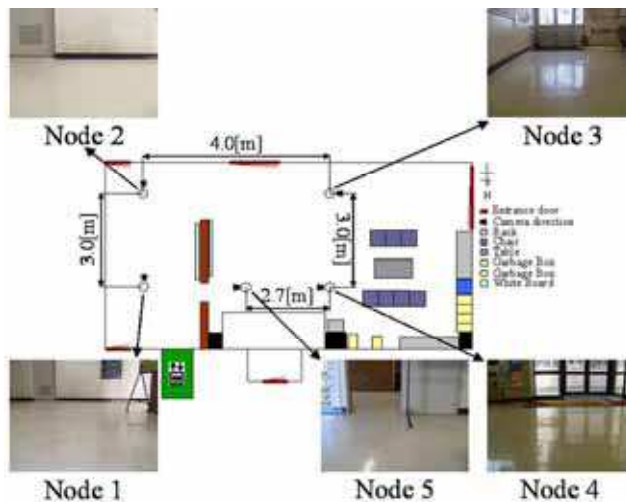
Fig. 21. Experiment layout of the hall environment with representative images of each node

8.1 Experimental setup

For the real-world experiments outlined in this research study, the ZEN360 autonomous mobile robot was used. It is equipped with a CCD colour video camera. The robot system is explained in section 4. Each image acquired by the system has a resolution of 320 x 240.

The robot is scheduled to navigate from Node 1 to Node 3, passing through Node 2 at the middle of the navigation in the corridor environment. Meanwhile, in the hall environment, the robot will have to navigate from Node 1 to Node 5 following the sequences of the node, and is expected to perform a turning task at most of the nodes.

The robot was first brought to the environments and a recording run has been executed. The robot is organized to capture images in order to supply environmental visual features for both position and orientation identification. The images were captured following the method explained in section 7.1 and 7.3, at around each specified nodes. Then, the robot generated a topological map and the visual features were used for training NNs.

After the recording run, the robot was brought once again to the environments to perform the autonomous run. Before start moving, the robot will identify its current position and based on the input of target destination, it will then plan the path to move on. Path planning

involves determining how to get from one place (node) to another, usually in the shortest manner possible. The work in this research study does not deal with this problem explicitly, though the topological map produced can be used as input to any standard graph planning algorithm.

A number of autonomous runs were conducted to see the performance of the proposed navigation method. In the experiments conducted at the corridor, the robot navigates from Node 1 and while moving towards Node 2, the robot corrects its own orientation at each step of movement based on the result of a comparison between visual features of the captured image against the 5 directions NN data of Node 2. The same procedure is used for a movement towards Node 3 from Node 2. The robot is set to localize itself at each node along the path during the navigation. An identical process is employed by the robot when navigating in the hall environment, where it starts navigating from Node 1 to Node 2, and followed by Node 3 and 4 before finished at the Node 5.

8.2 Experiment results

The result of the navigation experiments are displayed in Fig. 22 and Fig. 23. The robot successfully moved on the expected path towards Node 3 in each run of the result in Fig. 22. Even though at some points, especially during the first run test, the robot moved slightly away from the expected path on the centre of the corridor (to the right and left), it still came back to the path. The results demonstrated the proposed method to be asymptotically dexterous as the robot displacement in x axis along the expected path during the navigation is small.

Simultaneously, the experiments conducted at the hall environment are producing successful results as well (Fig. 23). The robot was able to navigate along the expected path, identified Node 2, 3 and 4 and turned safely towards the next node. Incidentally, after navigating half of the journey between Node 2 and Node 3 in the second run, the robot movement fell out from the path (to the left). Nevertheless, it still accomplished to move back to the path just before recognizing Node 3. This proved that the robot is able to determine its own moving direction and correct it towards the target.

The localized positions were very much near to the centre of the nodes except for Node 4 where the robot identified the node a bit earlier. The environmental factor surrounding might give influence to the localization performance that caused the robot to localize the node slightly far before reaching near the node. As the node is assigned quite near to the door at the north side of the hall environment, and furthermore the door width is quite large, there are possibilities that sunlight from the outside might entering the door and affected the robot localization performance. In fact, the robot is facing directly towards the door when navigating from Node 3 to Node 4. Although the discussed factors may give influences to the robot localization performance, the robot is still able to turn to the right successfully and move towards the correct path and arrived at Node 5, safely and successfully.

As an overall conclusion, the navigation results proved that the proposed navigation components have successfully operating properly under experimental conditions, allowing the robot to navigate in the environments while successfully recognize its own position and the direction towards the target destination. The robot is able to control its own posture while navigating and moved along the expected path without losing the direction to the target destination.
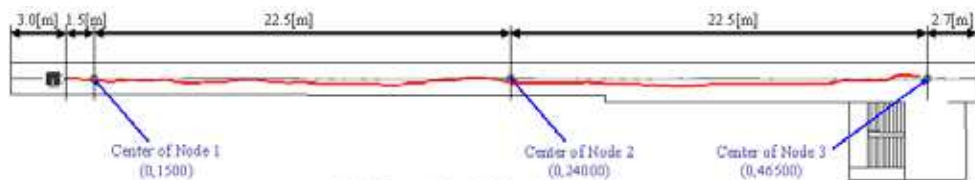
(a) Result of the first run
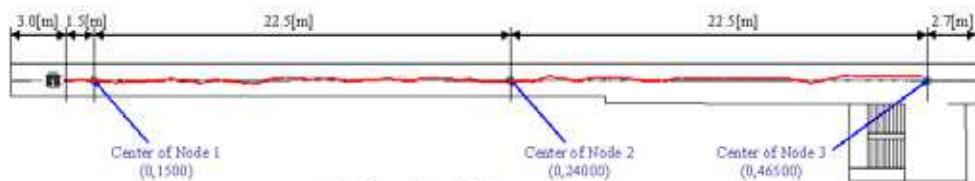
(b) Result of the second run
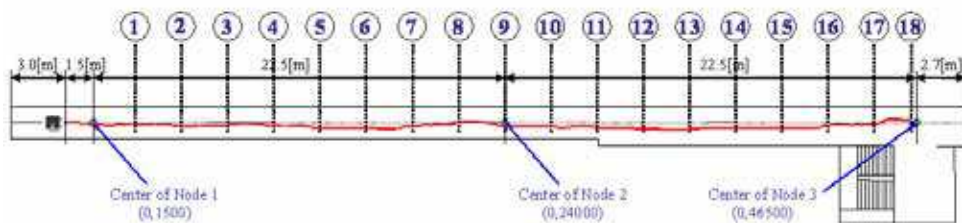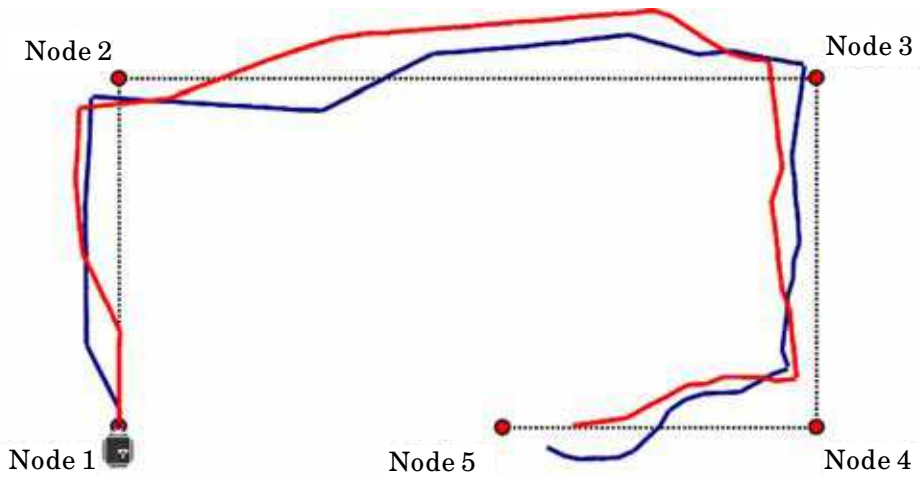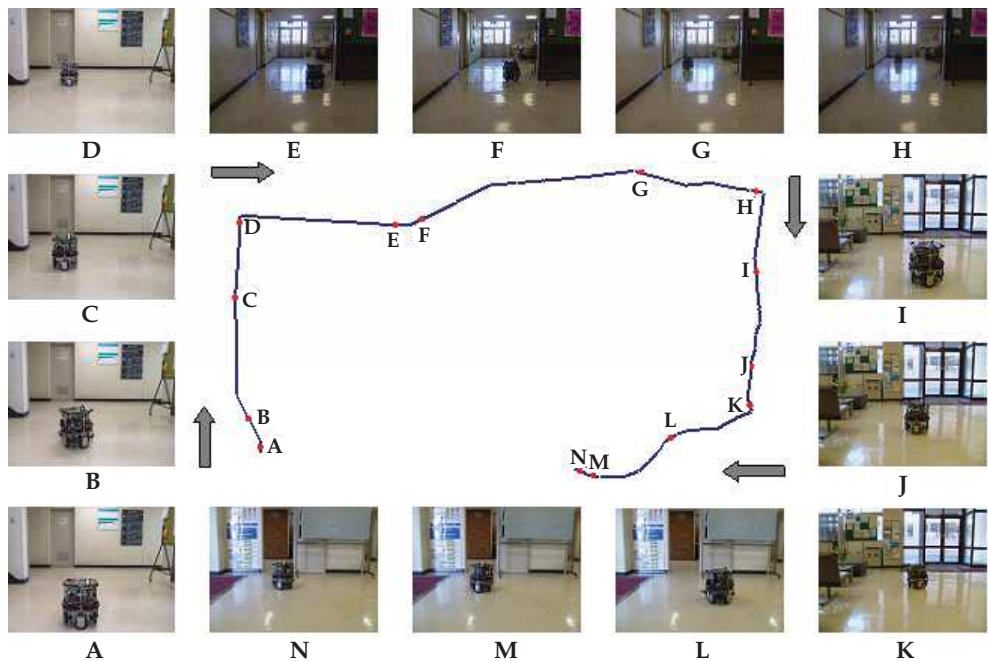

Navigation sceneries at the selected places.

Fig. 22. Results of the navigation experiment conducted at the corridor environment

a) Experimental result; blue path – first run, red path – second run

b) Navigation sceneries at selected places

Fig. 23. Result of the navigation experiment conducted at the hall environment

9. Conclusion

This chapter was concerned with the problem of vision-based mobile robot navigation. It built upon the topological environmental representation described in section 2.1. From the outset of this work, the goal was to build a system which could solve the navigation problem by applying a holistic combination of vision-based localization, a topological environmental representation and a navigation method. This approach was shown to be successful.

In the proposed control system, NN data is prepared separately for place and orientation recognition. By separating the NN data of place and orientation recognition, the navigation task was superbly achieved without any effect caused by the recognition domain area. This is mainly due to the fact that the width of the domain area for orientation recognition is practically wide under the method of preparing the instructor data as explained in section 7.3. At the same time, the width of domain area for position recognition is small in order to control the width and to prevent from robot stop slightly early before reaching certainly near around the target destination (node).

Furthermore, results from several navigation experiments lead the research work to identify a new way of preparing the instructor data for position recognition and hence improve the efficiency of localization process during navigation. With the new preparation method, it is believed that the domain area for localization of selected node can be control and the width could be smaller. This condition will help to prevent for early position recognition and help the robot to stop in somehow much more nearer to the centre of the node. Moreover, recognizing node at nearer point, it will help the robot to avoid other problems such as turning to early and crash to wall etc. at a node which is selected at a junction. In fact, the new instructor data acquiring method will help to reduce burdens on the end user during the recording run.

10. References

Asama, H.; Sato, M.; Kaetsu, H.; Ozaki, K.; Matsumoto, A. & Endo, I. (1996). Development of an Omni-directional Mobile Robot with 3 DoF Decoupling Drive Mechanism, *Journal of the Robotics Society of Japan (in Japanese)*, Vol.14, No.2, pp. 249-254, 1996.

Burrascano, P.; Fiori, S.; Frattale-Mascioli, F.M.; Martinelli, G.; Panella, M.; & Rizzi, A. (2002). Visual Path Following and Obstacle Avoidance by Artificial Neural Networks, *In "Enabling Technologies for the PRASSI Autonomous Robot" (S. Taraglio and V. Nanni, Ed.s), ENEA Research Institute*, pp. 30-39, 2002.

Geodome, T.; Tuytelaars, T.; Van Gool, L.; Vanacker, G.; & Nuttin, M. (2005). Omnidirectional Sparse Visual Path Following with Occlusion-robust Feature Tracking, *Proceedings of the 6th Workshop on Omnidirectional Vision, camera Networks and Nonclassical Cameras in conjunction with ICCV*, 2005.

Kawabe, T.; Arai, T.; Maeda, Y.; & Moriya, T. (2006). Map of Colour Histograms for Robot Navigation, *Intelligent Autonomous Systems 9*, pp. 165-172, 2006.

Mariottini, G.; Alunno, E.; Piazzi, J; & Prattichizo, D. (2004). Epipole-based Visual Servoing with Central Catadioptric Camera, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 497-503, 2005.

McClelland, J.L.; Rumelhart, D.E.; & the PDP Research Group (1986). Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 2, Psychological and Biological Models, *MIT Press, Cambridge, Massachusetts*, 1986.

Meng, M. & Kak, A.C. (1993). Mobile Robot Navigation Using Neural Networks and Nonmetrical Environment Models, *IEEE Control Systems Magazine*, Vol.13, No.5, pp. 30-39, 1993.

Morales, Y.; Carballo, A.; Takeuchi, E.; Aburadani , A.; & Tsubouchi, T. (2009). Autonomous Robot Navigation in Outdoor Pedestrian Walkways, *Journal of Field Robotics*, Vol.26, No.2, pp. 609-635, 2009.

Na, Y.K. & Oh, S.Y. (2004). Hybrid Control for Autonomous Mobile Robot Navigation Using Neural Network based Behavior Modules and Environment Classification, *Journal of Autonomous Robots*, Vol.15, No.2, pp. 193-206, 2004.

Park, I. & Kender, J.R. (1995). Topological Direction-giving and Visual Navigation in Large Environments, *Journal of Artificial Intelligence*, Vol.78, No.1-2, pp. 355-395, 1995.

Pomerleau, D.A. (1989). ALVINN: An Autonomous Land Vehicle in a Neural Network, *Technical Report CMU-CS-89-107*, 1989.

Rizzi, A. & Cassinis, R. (2001). A Robot Self-localization System Based on Omnidirectional Colour Images, *Journal of Robotics and Autonomous Systems*, Vol.34, No.1. pp. 23-38, 2001.

Rizzi, A.; Cassinis, R.; & Serana, N. (2002). Neural Networks for Autonomous Path-following with an Omnidirectional Image Sensor, *Journal of Neural Computing & Applications*, Vol.11, No.1. pp. 45-52, 2002.

Rumelhart, D.E.; McClelland, J.L.; & the PDP Research Group (1986). Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1, Foundations, *MIT Press, Cambridge, Massachusetts*, 1986.

Swain, M. & Ballard, D. (1991). Colour Indexing, *International Journal of Computer Vision*, Vol.7, No.1, pp. 11-32, 1991.

Ulrich, I. & Nourbakhsh, I. (2000). Appearance-based Place Recognition for Topological Localization, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1023-1029, 2000.

Vassallo, R.F.; Schneebeli, H.J; & Santos-Victor, J (1998). Visual Navigation: Combining Visual Servoing and Appearance Based Methods, *Proceedings of the 6th International Symposium on Intelligent Robotic Systems*, pp. 137-146, 1998.
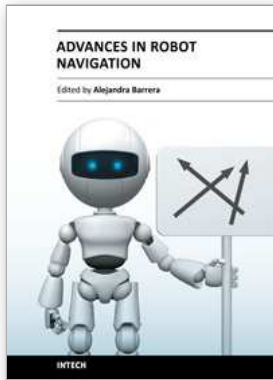
Advances in Robot Navigation

Edited by Prof. Alejandra Barrera

Robot navigation includes different interrelated activities such as perception - obtaining and interpreting sensory information; exploration - the strategy that guides the robot to select the next direction to go; mapping - the construction of a spatial representation by using the sensory information perceived; localization - the strategy to estimate the robot position within the spatial map; path planning - the strategy to find a path towards a goal location being optimal or not; and path execution, where motor actions are determined and adapted to environmental changes. This book integrates results from the research work of authors all over the world, addressing the abovementioned activities and analyzing the critical implications of dealing with dynamic environments. Different solutions providing adaptive navigation are taken from nature inspiration, and diverse applications are described in the context of an important field of study: social robotics.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

INTECH
open science | open minds