Generation of Environmental Representation of a Large Indoor Parking Lot

Jung-Ming Wang[1], Chih-Fan Hsu[2], Sei-Wang Chen[2], and Chiou-Shann Fuh[1]

[1] CSIE, National Taiwan University, Taipei, Taiwan
[2] CSIE, National Taiwan Normal University, Taipei, Taiwan
schen@csie.ntnu.edu.tw

Abstract. A technique for generating environmental representations of indoor parking lots is presented. The environmental representation of a parking lot is of use for the surveillance and management of the parking lot. The representation includes a map and an image, called the environmental map and the environmental image, respectively. Given the ceiling and floor plans of an indoor parking lot, we first determine the locations for installing omni-directional (OD) cameras such that the integrated fields of view (FOV) of the cameras cover the entire parking lot. After installing cameras, those monitoring views collected from the various cameras are then dewarped and mosaiced together to form an image of the whole environment. To test our method, we constructed a monitoring system for a model of parking lot that is scaled from a real one.

Keywords: Covering problem, omni-directional camera, camera calibration, path planning, camera deployment, environment map.

1 Introduction

Although parking lots are being continually built, many of the shortcomings associated with older parking lots have also been incorporated in the new ones. Those shortcomings will further lead to the drawbacks in wasted time, fuel consumption, air pollution, and congestion. In view of the drawbacks mentioned above, we have hence proposed an intelligent parking system (IPS), which intends to take advantage of currently available technologies to improve the efficiency, convenience, and security of a parking lot [1].

Omni-directional (OD) cameras have recently received increasing attention from computer vision researchers. Such sensors possess an FOV of 360 degrees, and readily perform multi-directional surveillance. Furthermore, OD cameras are basically static sensors and as a result multi-object tracking can be conducted. Finally, OD cameras are commonly mounted high (e.g., on ceilings) and produce top-view images of scenes. Top-view images minimize mutual occlusions between objects. In this study, OD cameras are considered.

To incorporate vision systems into the IPS, we are confronted with the issue of where and how to deploy cameras so that as few cameras as possible can be utilized to monitor the entire parking garage. We formulate this problem as a covering problem, in which the obstacle-free area of the parking garage is to be covered by the integrated FOVs of as few OD cameras as possible. In references [2][3], a submarine robot is used to map landforms under the sea. Since the sensors mounted on the robot had restricted sensing

T. Huang et al. (Eds.): ICONIP 2012, Part II, LNCS 7664, pp. 307–315, 2012.

areas, a covering technique was developed to systematically collect landform data from a wide area under the sea. In [4], several patterns of motion dealing with the respective situations of moving around an obstacle-free space, reaching a border, approaching an obstacle, running into a corner, and being trapped at a dead-end greatly increased the efficiency of cleaning robots. In [5], a number of high-level actions, including identifying walls, moving along walls, avoiding collisions, turning, and moving sideways, were defined for cleaning robots in order to deal with complex regions. Neural network based algorithms for covering operations proposed in [6] [7] have such advantage.

In recent years, multiple sensors are designed for covering path planning. In [8], cleaning robots were designed for tidying up dangerous areas. They achieved the purpose by developing algorithms that find optimal paths wandering through the areas to be cleaned. In [9], the authors proposed an algorithm combined with several existing methods, such as cellular decomposition, flow network, and template-based path planning for coverage problem. Two cleaning robots are designed to minimize the path moving in the whole area.

Although there is much research [10] about camera sensor network, most of them focus on the deployment of the camera with a limited degree field of view. In [11] and [12], the methods to detect the full-view coverage field are developed, and those methods help to improve the covering range of a given set of camera sensors. The coverage problem for disk sensors [13] has been studied in the past few years, but most of such research [14] focus on the wireless sensors.

The remainder of this paper is organized as follows. In Section 2, we address the vision system for our IPS. Sections 3 detail the steps of path planning and camera deployment. In Section 4, experimental results are presented, followed by concluding remarks and suggestions for further studies in Section 5.

2 System Overview

The camera deployment problem under consideration is converted into a covering problem: Given a parking garage and a collection of OD cameras, we wish to monitor the parking garage with as few cameras as possible. In practice, the ceiling and floor plans of the parking garage are provided. Fig. 1(a) shows a floor plan example, in which the black areas indicate obstacles (possibly pillars or walls) and the white area, denoted by P, is the area to be monitored. In Fig. 1(b), the circle marked with q represents the IFOV of an OD camera. Our objective here is to use as few q's as possible to cover the entire P. Fig. 1(c) shows a possible result. A ceiling plan, however, is not necessary the same as the floor plan. The black areas indicate holes that we cannot mount any camera and the white area is the mountable area for the cameras.

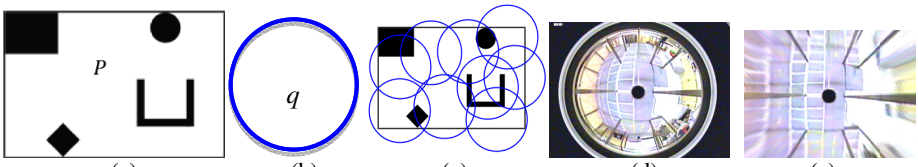


Fig. 1. Sample images of (a) floor plan, (b) IFOV of OD camera, (c) covering result, (d) OD image and (e) dewarping result

A flowchart for solving the problem described above is depicted in Fig. 2(a). To begin, the given ceiling and floor plans of parking garage are digitized using a scanner to obtain a digital image of the plans. Both the resolution of the image and the scale of the plans are then used for camera calibration, in which the mapping from the image plane of an OD camera to the real world space is established. Based on this mapping, the IFOV of an OD camera is determined. During path planning, a path wandering through the obstacle-free space in the ceiling is generated. Having determined the path, we next deploy cameras using the path as a guide in an attempt to minimize the required number of cameras.

Fig. 2 (b) describes the architecture of the vehicle tracking system [1]. This system can be divided into two layers, a front-end layer and a back-end layer. The front-end layer consists of the omni-directional cameras arranged by the path planning method. These top-down views (Fig. 1(d)) from the various cameras are dewarped (Fig. 1(e)) and mosaiced together to form an image of the whole environment. Foreground objects are extracted in the front-end layer and their feature values are transmitted to the back-end layer to integrate.

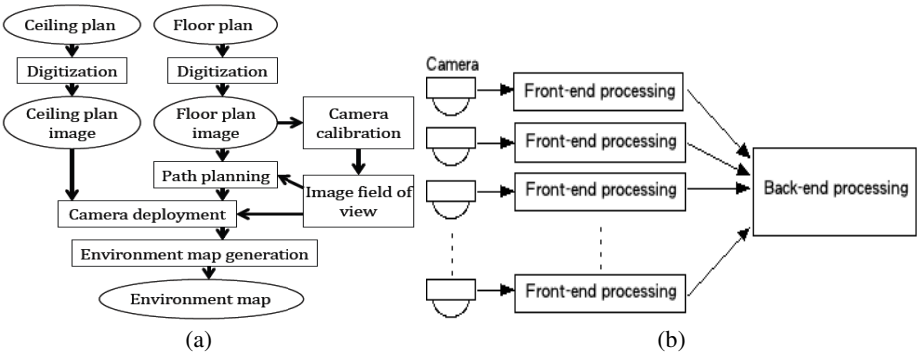


Fig. 2. System flowchart of (a) camera installation and (b) tracking system

3 Camera Installation

3.1 Path Planning

We achieve the covering problem by finding a path that meanders through the mountable area of the ceiling using a neural network [7]. Fig. 3(a) depicts the architecture of the neural network, in which the small circles represent neurons that are 8-connected

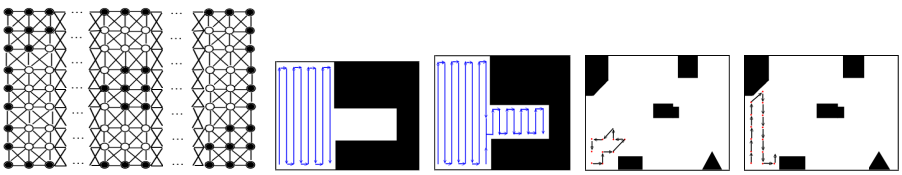


Fig. 3. (a)Architecture of the proposed neural network; (b) the path generation trapped at deadends; (c) backtracking is introduced for deadend problem; and path generate without (d) and with (e) momentum.

to each other to form a 2D layer corresponding to the digital image of the given ceiling plan. The black circles indicate the locations where neurons are falling in obstructed areas and the white circles indicate locations where neurons are falling in obstacle-free areas. In the other words, the cameras can be mounted at the white circles while they cannot be mounted at the black circles.

The behavior of neural activation [8] for any neuron is described by Grossberg's shunting equation [15], which in turn was inspired by Hodgkin and Huxley's dynamic equation of voltage across a small patch of nerve cell membrane t. The shunting equation governing the activation, x_i, of neuron n_i is

$$\frac{dx_i}{dt} = -Ax_i + (B - x_i)J_i^+ - (C + x_i)J_i^-, \tag{1}$$

where A, B, and C are positive constants, J_i^+ and J_i^- are the net excitatory and inhibitory inputs to neuron n_i given by

$$J_i^+ = \left([I_i]^+ + \sum_{n_j \in N_i} w_{ij}[x_j]^+ \right), \quad \text{and} \quad J_i^- = [I_i]^-, \tag{2}$$

in which $[\bullet]^+$ and $[\bullet]^-$ are functions defined as $[a]^+ = max\{a, 0\}$ and $[a]^- = max\{-a, 0\}$, and I_i is the external input

$$I_i = \begin{cases} -D & \text{if } n_i \text{ is in an obstcle area,} \\ D & \text{if } n_i \text{ is in an obstacle-free area and hasn't been visited,} \\ 0 & \text{if } n_i \text{ has been visited,} \end{cases} \tag{3}$$

where D is a constant. Weight w_{ij} in Eq. (2) is the synaptic strength of the link connecting neuron n_i and its neighbor $n_j \in N_i$, where N_i is the set of 8-connected neighbors of n_i. Weight w_{ij} is defined as $w_{ij} = f(\|\boldsymbol{r_i} - \boldsymbol{r_j}\|)$, where $\boldsymbol{r_i}$ and $\boldsymbol{r_j}$ are the position vectors of neurons n_i and n_j, respectively, and f is a monotonically decreasing function defined as $f(z) = c/z^2$, where c is a positive constant.

According to Eq. (1), x_i will decay at the rate of A if there is no input to neuron n_i. However, if the excitatory input dominates the inhibitory input, x_i will be influenced solely by the middle term. In this case, the closer x_i is to constant B, the slower the change in x_i, and in the limit, x_i equals B. B establishes the upper limit for x_i. With the same inference, $(C+x_i)\ [I_i]^-$ forces x_i to always be greater than $-C$. Hence, $-C$ is the lower limit for x_i. In summary, x_i will be restricted to lie in the range $[B, -C]$. If the activation, x_i, of neuron n_i turns out to be positive, the neuron pulls the path toward itself; otherwise, the path is pushed away.

Initially, the corner neuron in the parking area is selected as the starting neuron. This selection will help to decide the initial moving direction by defining that along the boundary of the obstacle-free area. Consider the solutions x_i to Eq. (4), the neuron in the neighborhood of the selected neuron with the largest calculated activation is then chosen as the next neuron to be visited by the path:

$$n_{next} = \arg\max_{n_j \in N} \{ x_j \}, \tag{4}$$

where N denote the neighborhood of a neuron. In order to exit from a deadend (Fig 3(b)), we introduce a backtracking that returns along the generated path until either a neuron is discovered which has an unvisited neighbor in an obstacle-free area or the starting neuron of the path is reached. Besides, the above process may generate a path with many abrupt changes in direction as in Fig. 3(d). We introduce a momentum for smoothing the path during path generation. Let $\boldsymbol{v_p}$ and $\boldsymbol{v_n}$ denote the directional vectors of the previous and next neurons, respectively. The angle θ between $\boldsymbol{v_p}$ and $\boldsymbol{v_n}$ can be calculated. The following equation extended from Eq. (4) incorporates the momentum in the path planning process.

$$n_{next} = \arg\max_{n_j \in N} \{ x_j + b(1-\theta_j) \}, \tag{5}$$

where b is a weight for balancing the neural activation and angle. Fig. 3(e) shows the smooth result.

3.2 Deployment of Cameras

Having determined the path, we look for adequate positions for installing cameras using the path as a guide in an attempt to use as few cameras as possible to monitor the entire parking garage. To this end, we first specify four criteria: (1) the overall distribution of cameras should be as uniform as possible, (2) the greater the distances between cameras the better, (3) overlap between the FOV's of adjacent cameras is required, and (4) all the obstacle-free areas of the parking garage should be covered by the integrated FOVs of the cameras. The first two criteria are intended to reduce the number of cameras employed, while the last two ensure that no obstacle-free area of the parking garage is missed.

A neural network is developed for the camera deployment step. Fig. 4 illustrates the architecture of the neural network. We first uniformly place the neurons along the path determined in the previous step on the floor plan (see Fig. 4(a)). Each neuron is then connected to its neighbors, the distances of which are shorter than the radius of the IFOV of an OD camera (Fig. 4(b)). However, if one of those neighbors is blocked by an obstacle on the floor, no connection is made for the neighbor as the lower neuron denoted P_2 in Fig. 4(b).

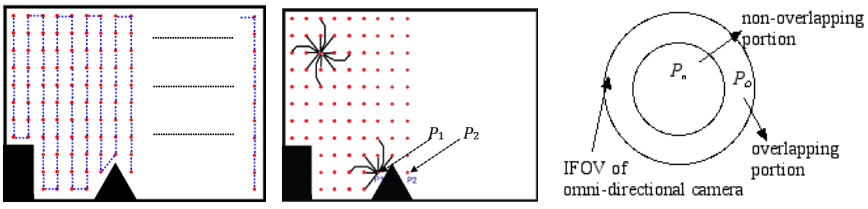


Fig. 4. Architecture of the neural network: (a) neurons are placed along the path; (b) each neuron (ex. P_1) is connected to its neighbors except for the lower neuron denoted P_2; and (c) OD camera is divided into two regions.

Again, the Grossberg shunting equation is used to describe the dynamic activation of any neuron as they shown in Eq. (1), (2), and (3). The different is that we define the external input according to the camera distribution:

$$I_j = \begin{cases} D' & \text{if } n_j \text{ hasn't been installed and hasn't been covered} \\ 0 & \text{if } n_j \text{ hasn't been installed and has been covered} \\ -D' & \text{if } n_j \text{ has been installed a camera} \end{cases}, \tag{6}$$

in which D' is a positive constant. Initially, we set all neural activations to one. A starting neuron n is then selected from the path. We then calculate the activations for the neurons, n_j, in N using Eq. (4). The neuron with the largest activation is chosen as the next neuron for a camera to be installed. We designate neuron n_{next} *as* "installed" and its neighbors are marked as "covered". For each neighbor, n_k, of n_{next} we then update its activation, x_k, according to $x_k = \min\{x_k, x_k'\}$, where

$$x_k' = \begin{cases} d_k / R & \text{if } n_k \in P_o \\ (d_k - R_n)/R_n & \text{if } n_k \in P_n \\ x_k & \text{otherwise} \end{cases}, \tag{7}$$

in which d_k is the distance between neurons n_k and n_{next}, and R, R_n, P_O, and P_n are defined below. Referring to Fig. 4 (c), we divide the IFOV of an OD camera into two regions: overlapping (P_O) and non-overlapping (P_n). The radiuses of IFOV and P_n are R and R_n, respectively.

After installing a camera on neuron n_{next}, we proceed along the path to find a neuron, in which a camera hasn't been installed and is not covered by the IFOV of any camera. Once such a neuron is found, the above process is repeated for this neuron. Installing cameras using the above process often results in an irregular deployment of cameras. To compensate for this drawback, we modify Eq. (4) in order improve the alignment of cameras.

Suppose we are considering a neuron n on the path. We wish to determine if it is appropriate to install a camera. Let S be a subset of N and P_S be the positions of the neurons in S, at which cameras have been installed. For each position (a_S,b_S) in P_S, we calculate $\eta_s = \min\{|a_j - a_s|, |b_j - b_s|\}/R$, where (a_j,b_j) is the position of neuron n_j. The larger the value of η_s the worse the alignment.

Increasing the degree of alignment of cameras may also increase the number of gaps between the IFOVs of the cameras, which will increase the number of cameras employed to cover the gaps. In order to reduce the number of gaps, we calculate

$$\eta_t = \begin{cases} d_{jt} / d_{\max} & \text{if } d_{jt} \le d_{\max} \\ 0 & \text{otherwise} \end{cases},$$

where d_{jt} is the distance between n_t and n_j and $d_{\max} = \sqrt{2}R$. The smaller the calculated value of η_t the better the separation between neurons n_t and n_j. Finally, we obtain

$$n_{next} = \arg\max_{n_j \in N}\{(1-\alpha)\beta y_j\}, \tag{8}$$

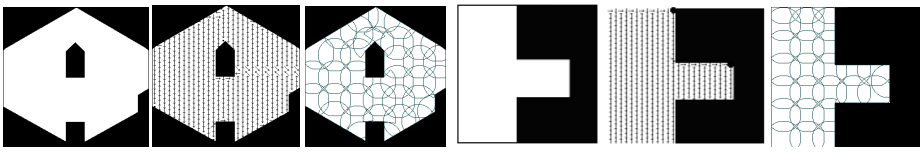
where $\alpha = \max_{(a_s,b_s)\in P_S}\{\eta_s\}$ and $\beta = \min_{(a_t,b_t)\in P_S}\{\eta_t\}$.

Obviously, different starting neuron selected in path planning will generate a different path and will influence the selection of neuron to deploy the first camera in deployment. We intend to reduce the number of cameras employed and no obstacle-free area of the parking garage is missed in our approach. Eq. (8) helps us to achieve these proposes and let the deployment cannot be significantly influenced by the path planning result.

4 Experiments

Both synthetic and real data were used in our experiments. Both synthetic and real plans were digitized at a resolution of 600 dpi. For synthetic plans, we set the radius of the IFOV of the cameras to 200 pixels. For real plans, the radius of the IFOV of cameras is 212 pixels, which was determined from the image resolution (600dpi), the scale of the floor plans (7000:1), and the height (2.8m) at which the cameras were installed. Recall that there are two neural networks involved in the proposed technique. The parameters associated with the first neural network were A=9, B=1, C=0.006, D=10 and t=0.25. Those for the second neural network were A=1, B=1, C= 0.006 and D=10.

Two synthetic floor plans were chosen to illustrate the proposed technique. Fig. 5 shows the experimental results for two synthetic plans. The input floor plan is the same as the ceiling plan, and both of them are depicted in Fig. 5(a) and (d); the results of path planning are shown in Fig. 5(b) and (e). The generated path starts at the bottom left corner of the bottom obstacle and roams back and forth across the entire obstacle-free area. The path successfully avoided the obstacles and achieved reasonable changes in direction under the control of momentum. Fig. 5(c) and (f) show the camera placement results. As we can see, the entire obstacle-free area is completely covered by the total IFOVs of the cameras. The locations in each region are covered by the same cameras, which can show that our solution is 100% coverage.

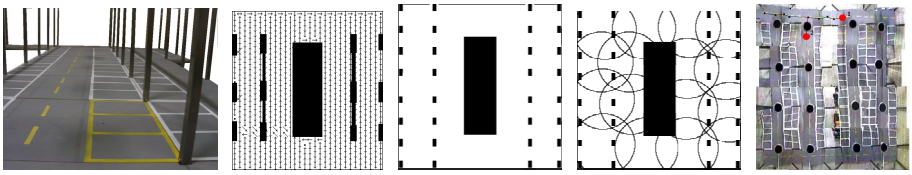


(a) (b) (c) (d) (e) (f)

Fig. 5. The experimental results for two synthetic plans: (a)(d) floor plan, (b)(e) path planning, and (c)(f) camera deployment of both synthetic plans, respectively

We construct a model of parking lot (Fig. 6(a)) that is scaled from a real one. In this case, the ceiling plan is different from the floor plan. Fig. 6(b) shows the ceiling plan of this parking lot and its corresponding path planning. In this figure, white area

means the mountable area, while parts mean the pillars or walls that the camera cannot be mounted on. Around the parking lot, a wall is surrounded and shown as the boundary in the figure. After applying the deployment algorithm, we can obtain the camera deployment (Fig. 6(d)) on the floor plan (Fig. 6(c)). According to the camera deployment, we mounted 16 omni-directional cameras on the ceiling in the parking lot. The resolution of a camera image is reduced to 320×240, and each camera captures 30 frames per second. Fig. 15(e) shows the mosaiced image and the tracking results using the method proposed in [1].



(a) (b) (c) (d) (e)

Fig. 6. Our model of parking lot: (a) an overview of the model, (b) ceiling plan and its corresponding path planning, (c) floor plan, (d) camera deployment, and the actual parking area

5 Conclusions and Further Work

A technique for automatically arranging cameras to monitor a parking garage is presented. We formulate the arrangement problem as a covering problem. In this problem, we seeks a path with as few turns as possible, but that visits all obstacle-free areas of the parking garage. After the path panning, we determines the appropriate positions for installing cameras using the path as a guide in an attempt to use as few cameras as possible to monitor the entire parking garage. The proposed approach does not guarantee optimal results because different starting conditions during path planning and camera deployment would lead to different camera arrangements. However, no significant difference in the number of cameras employed was observed in our experiments for the different conditions described above.

After installing the cameras, the foreground objects are detected from the video sequences captured by the cameras. The detection results can be integrated to determine the paths of moving objects in the environment. Our system has been applied to a model of an indoor parking garage to obtain the paths taken by cars. It has the advantage to handle the vehicles start moving from anywhere of the parking lot. In the future we will estimate an optimal path taken by each car, and combine this with a guidance system to help drivers navigate the parking garage to available parking spaces.

References

1. Wang, J.M., Tsai, C.T., Chen, S., Chen, S.W.: Omni-Directional Camera Networks and Data Fusion for Vehicle Tracking in an Indoor Parking Lot. In: IEEE International Conference on Advanced Video and Signal based Surveillance, Sydney, p. 45 (2006)

2. Hert, S., Tiwari, S., Lumelsky, V.: A Terrain-Covering Algorithm for an AUV. Journal of Autonomous Robots 3, 91–119 (1996)
3. Yilmaz, N.K., Evangelinos, C., Patrikalakis, N.M., Lermusiaux, P.F.J., Haley, P.J., Leslie, W.G., Robinson, A.R., Wang, D., Schmidt, H.: Path Planning Methods for Adaptive Sampling of Environmental and Acoustical Ocean Fields. J. Oceans, 1–6 (2006)
4. De Carvalho, R.N., Vidal, H.A., Vieira, P., Ribeiro, M.I.: Complete Coverage Path Planning and Guidance for Cleaning Robots. In: Proceedings of the IEEE International Symposium on Industrial Electronics, Guimaraes, Portugal, pp. 677–682 (1997)
5. Fu, Y., Lang, S.Y.T.: Fuzzy Logic Based Mobile Robot Area Filling with Vision System for Indoor Environments. In: Proceedings of IEEE International Symposium on Computational Intelligence in Robotics and Automation, Monterey, USA, pp. 326–331 (1999)
6. Luo, C., Yang, S.X., Meng, Q.-H.: Real-time Map Building and Area Coverage in Unknown Environments. In: Int'l Conf. on Roboticss and Automation, Barcelona, pp. 1736–1741 (2005)
7. Yang, S.X., Luo, C.: A Neural Network Approach to Complete Coverage Path Planning. J. Systems, Man, and Cybernetics-Part B: Cybernetics 34(1), 718–725 (2004)
8. Lee, J.H., Choi, J.S., Lee, B.H., Lee, K.W.: Complete Coverage Path Planning for Cleaning Task using Multiple Robots. In: Proceedings of the 2009 IEEE Int'l Conf. on Systems, Man, and Cybernetics, San Antonio, pp. 3618–3622 (2009)
9. Janchiv, A., Batsaikhan, D., Kim, G.H., Lee, S.G.: Complete Coverage Path Planning for Multi-Robits Based on. In: Int'l Conf. on Control, Automation and System, pp. 824–827. KINTEX, Gyeonggi-do (2011)
10. Soro, S., Heinzelman, W.: A Survey of Visual Sensor Networks. J. Advances in Multimedia 2009, 1–22 (2009)
11. Wang, Y.: On Full-view Coverage in Camera Sensor Networks. In: IEEE INFOCOM, Shanghai, pp. 1781–1789 (2011)
12. Hörster, E., Lienhart, R.: On the Optimal Placement of Multiple Visual Sensors. In: ACM Int'l Workshop on Video Surveillance and Sensor Networks, New York, pp. 111–120 (2006)
13. Huang, C.F., Tseng, Y.C.: The Coverage Problem in a Wireless Sensor Network. In: ACM Int'l Conf. on Wireless Sensor Networks and Application, New York, pp. 115–121 (2003)
14. Cardei, M., Wu, J.: Energy-Efficient Coverage Problems in Wireless Ad-hoc Sensor Networks. J. Computer Communication 29(4), 413–420 (2006)
15. Grossberg, S.: Nonlinear Neural Networks: Principles, Mechanisms, and Architecture. J. Neural Networks 1, 17–61 (1988)
16. Hodgkin, A.L., Huxley, A.F.: A Quantitative Description of Membrane Current and its Application to Conduction and Excitation in Nerve. J. Physiology 117, 500–544 (1952)