



Tecnura

ISSN: 0123-921X

tecnura@udistrital.edu.co

Universidad Distrital Francisco José de Caldas
Colombia

Martinez Sarmiento, Fredy Hernán; Delgado, Jesús Alberto
Wireless Visual Sensor Network Robots- Based for the Emulation of Collective Behavior
Tecnura, vol. 16, núm. 31, enero-marzo, 2012, pp. 10-18
Universidad Distrital Francisco José de Caldas
Bogotá, Colombia

Available in: <http://www.redalyc.org/articulo.oa?id=257023013002>

- How to cite
- Complete issue
- More information about this article
- Journal's homepage in redalyc.org

redalyc.org

Scientific Information System
Network of Scientific Journals from Latin America, the Caribbean, Spain and Portugal
Non-profit academic project, developed under the open access initiative

Wireless Visual Sensor Network Robots-Based for the Emulation of Collective Behavior

Red inalámbrica de sensores visuales basada en robots para la emulación de comportamiento colectivo

Fredy Hernán Martínez Sarmiento

Electrical engineer, PhD candidate in engineering. Professor at the Distrital University Francisco José de Caldas. Bogotá, Colombia.
fhmartinezs@udistrital.edu.co.

Jesús Alberto Delgado

Electrical engineer, master in electrical engineer, PhD in cybernetics. Professor at the National University of Colombia. Bogotá, Colombia.
jadelgador@unal.edu.co.

Clasificación del artículo: Investigación (conciencias)

Fecha de recepción: 22 de julio de 2011

Fecha de aceptación: 28 de noviembre de 2011

Palabras clave: Cámara nodo, comportamiento colectivo, quorum sensing, red visual de sensores.

Keywords: Camera node, collective behavior, quorum sensing, visual sensor network.

ABSTRACT

We consider the problem of bacterial quorum sensing emulate on small mobile robots. Robots that reflect the behavior of bacteria are designed as mobile wireless camera nodes. They are able to structure a dynamic wireless sensor network. Emulated behavior corresponds to a simplification of bacterial quorum sensing, where the action of a network node is conditioned by the population density of robots (nodes) in a given area. The population density reading is done visually using a camera. The robot makes an estimate of the population density of the images, and acts according to

this information. The operation of the camera is done with a custom firmware, reducing the complexity of the node without loss of performance. It was noted the route planning and the collective behavior of robots without the use of any other external or local communication. Neither was it necessary to develop a model system, precise state estimation or state feedback.

RESUMEN

Se considera el problema de la emulación del quorum sensing bacterial sobre pequeños robots móviles. Los robots que reflejan el comportamiento

de las bacterias se han diseñado como nodos móviles inalámbricos con cámaras. Ellos son capaces de estructurar una red dinámica inalámbrica de sensores. El comportamiento emulado corresponde a una simplificación del quorum sensing bacterial, en donde la acción de un nodo de la red es condicionada por la densidad poblacional de robots (nodos) en un área dada. La lectura de la densidad poblacional se hace visualmente utilizando una cámara. El robot realiza una estimación de la densidad poblacional a partir de las imágenes,

y actúa de acuerdo con esta información. La operación de la cámara es realizada por medio de un firmware personalizado, lo que permite reducir la complejidad del nodo sin pérdida de desempeño. Se observó la planificación de ruta y el comportamiento colectivo de los robots sin el uso de ningún otro tipo de comunicación, externa o local. Tampoco fue necesario desarrollar un modelo del sistema, una estimación precisa de estados o realimentación alguna de estados.

* * *

1. INTRODUCTION

The incorporation of visual sensors in mobile robotic systems is one of the most important successes [1]. A vision system gives to the autonomous robots the ability to locate and recognize objects at distance, which simplifies navigation and interaction in dynamic y/or unknown environments.

However, extracting relevant information from the video stream is a very complex process, requiring a significant amount of computing power [2]. In addition, an error in the information processed produce incorrect operation of the robot, which can be very dangerous for humans to interact with it.

Today, most mobile robots on the market are physically large and expensive, which makes them unsuitable for many research projects in robotics. Many are developed without considering the final application, without taking into account that many tasks can be solved with very simple structures [3].

The smart camera networks have recently received much attention, partly due to the wide availability of low cost sensors, and another largely because of its ability to simplify and improve existing applications already popular in wireless sensor networks [4]. Applications that have shown great interest in these new sensors include environmental monitoring, tracking and surveillance (malls, offices, parks, etc.).

Some mobile robotic systems are remotely controlled, and their vision system only acquires a video stream, which is sent to a central server [5], [6]. This design scheme is not suitable for applications where cost and location of the system are critical. The Visual Sensor Networks (VSNs), and more, networks in which nodes are replaced by small mobile robots, need a structure with sufficient capacity for local processing (on-board), so as to minimize the amount of information to be transmitted, to optimize the bandwidth and reduce interference in the environment.

Based on these ideas, we seek to develop a robotic platform as simple as possible, allowing us to explore the problem of collective agent navigation in unknown environments, based on visual landmarks. Possible applications such as patrol and coverage allowed us to consider its design from the point of view of a wireless sensor network. Implementation of the model of behavior based on bacterial quorum sensing seeks to propose a new design strategy in which, for the specific case of navigation is not necessary to develop a system model and state feedback, instead of this, the navigation is performed according to landmarks, which may correspond to the same agents in the case of collective navigation.

The contribution of the research presented here is focused on the benefits of the prototype developed. Firstly, we devised a design methodology for the development of the camera node focused on the final application of the prototype. In this way, we

achieve a compact and minimalist design suitable for laboratory research. Secondly, to complement our design methodology, we always consider the total cost of the platform, achieving a system quite economical. With regard to bacterial quorum sensing emulation, the ability to observe the collective behavior of artificial agents allows hardware designers to establish new design criteria; here the specific case of route planning for a group of robots in a given environment without the use of local or external communication. The remainder of this paper is organized as follows. Section 2 presents the problem formulation. Section 3 introduces approach to system-level design of the camera node according to its use for research in mobile robotics and VSNs. In Section 4, we present the details of construction of the prototype. Section 5 we present a brief feature comparison with other cameras node reported in the literature. Section 6 describes the algorithm used in the emulation of quorum sensing-based behavior in the agent, as well as observations of the implementation in the laboratory. Finally, Section 7 summarizes the main findings of our work.

2. PROBLEM FORMULATION

Let $W \subset \mathbb{R}^2$ be the closure of an open set in the plane that has a connected open interior with obstacles that represent inaccessible regions [7]. Let $E \subset W$ be the free space, which is the open subset of W with the obstacles removed. Let Γ be a set of boundary marks, each of which is an open linear subset of E . These marks are line segments with both end points on the boundary of E . For example, in Fig. 1 the marks are labeled $\Gamma = \{a, b, c, d\}$.

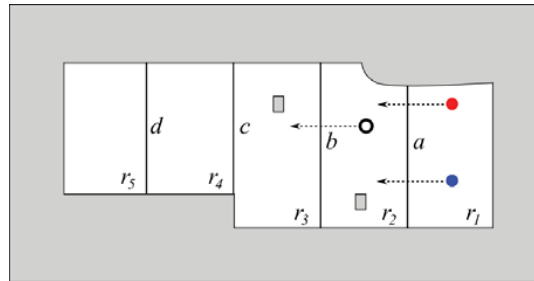


Fig. 1. Three agents moving in the environment.

The collection of obstacles (gray areas inside and

outer boundary) and marks allows a decomposition of E into connected regions. The environment is decomposed into a set R of regions:

$$R = \{r_1, r_2, r_3, r_4, r_5\} \quad (1)$$

These regions are places of interest that can correspond, for instance, to activity levels of bacteria, where each level corresponds to a specific activity to be developed by the bacteria, either go and colonize the area, move to a new area or stay inactive. For example, the marks in Fig. 2 divide E into five two-dimensional regions.

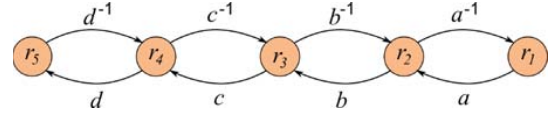


Fig. 2. Region graph.

There will be a number n , unknown but finite, of agents moving inside the environment. The trajectory of a single agent is represented by:

$$\tilde{x}_i: [0, t_f] \rightarrow E \quad (2)$$

$[0, t_f]$ represents a time interval and t_f is the final time.

$D = \{-1, 1\}$ is the set of mark directions. Thereby, the model depicted in Fig. 1 is obtained by a mapping:

$$h: E \rightarrow Y \quad (3)$$

In which $Y = \Gamma \times D$. This will define the only two possible paths of the agent. For each $\alpha \in \Gamma$, α denotes the forward direction and α^{-1} denotes the backward direction. In the example in Fig. 1, right to left represent the forward direction and left to right represent the backward direction. A possible observation string (crossing sequence) is $\square = abb^{-1}$, where we indicate that the agent moves from r_1 to r_2 , then from r_2 to r_3 , and then returns to r_2 . This way we can detail the ways of the agents in the environment.

We define the Region Graph G as follows. Every vertex in G corresponds to a region in E . A directed edge is made from $r_i \in R$ to $r_j \in R$ if and

only if an agent can cross a single mark to go from r_i to r_j . The corresponding mark label α is placed on the edge. We also create an edge from $r_j \in R$ to $r_i \in R$ with the mark label α^{-1} for the opposite direction. The corresponding region graph for Fig. 1 is presented in Fig. 2.

The initial study problem can be formulated as follows: Given the region graph G induced by marks and obstacles, and the observation string σ , shows the navigation of a group of agents without direct communication between them using visual information from agents and activating the action according the behavior of agents.

In the previous problem, we assume a model of bacterial quorum sensing in which a bacterium is a pair of the form [8].

$$V = (f, P) \quad (4)$$

Where, f is a nonnegative integer ($f \in \mathbb{Z}$) that indicate the amount of neighboring bacteria, and P is a point in q -dimensional space ($P \in \mathbb{R}^q$). For the particular case of the hardware prototype developed in this research, f take values 0, 1 and 2 (three agents, three robots interacting) for each bacterium over time, and P is a point in two-dimensional plane ($q = 2, \Rightarrow P \in \mathbb{R}^2 \Rightarrow P = (p_1, p_2)$).

3. METHODOLOGY

The flowchart of Fig. 3 conceptualizes our proposed methodology design of visual sensor node according to its use in research. This is an inverse design problem, where the outputs, i.e. the final performance are known but its inputs, i.e., operating parameters, are unknown.

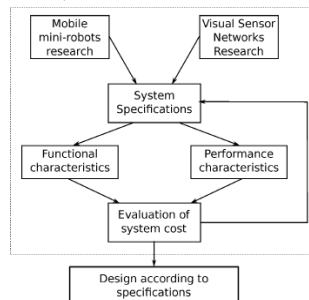


Fig. 3. Visual sensor node flowchart of application-oriented design

The first step is to generate a comprehensive set of system specifications based on the two approaches proposed for use: mobile mini-robots research and VSN research. These specifications are based on hardware characteristics of existing mini-robots in the lab, and functionality to be achieved in both the fixed sensor networks, as in dynamic networks created with the robots.

The functional characteristics refer to the network topology (included features of the environment in which the robots move), type of sensors used in each node, processing (local and distributed processing) and communication capacity, and energy resources. The latter two are of great importance given the autonomy and low cost that we want in the system. The performance characteristics are concerned with network robustness, accuracy, response rate, and autonomy. The solution is not necessarily a single point in the space of possible designs, in fact there are multiple solutions that meet the design criteria. The final choice was made considering both simplicity of design and component availability.

The design according to specifications made by the previous tuning process has the following key features:

- As usual in the design of these camera nodes (Table 1), the system has three basic elements: image sensor, embedded processor and wireless communication.
- To optimize the bandwidth of the wireless communication unit, is performed local processing of images to extract from them relevant information.
- The network is a distributed system, so it is also contemplated the possibility of distributed processing, which reduces the processor requirement, and therefore energy (planned research).

The configuration of the visual sensor node is shown in Fig. 4. In search of a processing unit with high performance, low cost and flexibility, we conclude that the most appropriate device for the node prototype was a CPLD (Complex Programmable Logic Device), the Altera MAX II EPM570T100C5N. The CPLD exceeds the small microcontrollers in performance and big ones in

flexibility, because the firmware allows to run parallel algorithms in hardware faster than a sequential operating system, often at a lower cost, and even without considering its high speed clock. The CPLD controls the optical sensor, temporarily stores the images in SDRAM (Synchronous Dynamic Random Access Memory) memory, and transmits/receives data via the wireless transceiver. It can set the parameters of the imager, instructs the imager to capture a frame and run simple local computation on the image to produce an inference (background subtraction and frame differentiation).

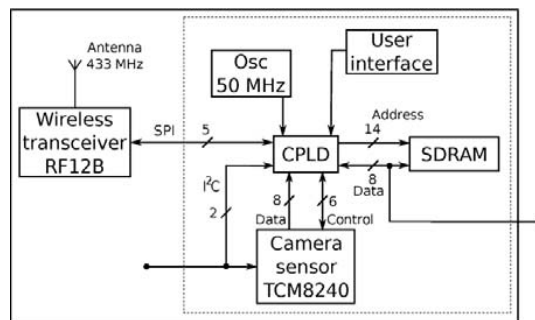


Fig. 4. Visual sensor node hardware architecture.

As optical sensor, we chose a high quality imager with moderate clock speed requirements and low power consumption in its sleep state. It is a 1.3 Mega-pixels resolution C2MOS camera module (TCM8240MD) from Toshiba. The sensor has 1/3.3 inch of optical format. Its clock frequency can be set to anywhere from 6 to 20 MHz. We set the imager clock to 6 MHz to provide the CPLD (which operates at 50 MHz clock), enough time to grab an image pixel and copy it into the memory. The image sensor has SXGA resolution (1280×1024). The camera configuration is done by I2C. For convenience, the system has added an I2C port to configure the camera directly from an external system. The image processing unit is capable of generating two formats: 4:2:2 YUV and 5:6:5 RGB as multiplexed 8 bit parallel output. The visual sensor node only supports RGB color.

The visual sensor node uses external SDRAM to provide the necessary memory for image storage and manipulation. The external memory provides us on-demand access to memory resources at the

capture and computation time. According to the capacity of the selected sensor, we need a minimum storage space of $2 \times 1280 \times 1024 = 2'621.440$ Bytes per frame (SXGA - Super eXtended Graphics Array in 5:6:5 RGB format). We selected the Micron SDRAM MT48LC8M8A2TG memory of 64 MB (8 MB x 8), meeting the requirements of cost and performance.

As wireless communication module we use a low power, low cost, ISM Band FSK transceiver (RF12B) from Hope RF. We designed a card for this module with an antenna on-chip to work in the band of 433 MHz. This module communicates directly with the CPLD through SPI compatible serial control interface. Its integrated digital data processing features (data filtering, clock recovery, data pattern recognition, integrated FIFO and TX data register) reduces the processing load on the CPLD, and is capable of transmitting up to 115.2 kbps in digital mode. The emulation does not use this module.

4. CAMERA NODE PROTOTYPE

The prototype is implemented on a printed double layer board of 4,7cm x 3,6 cm. On this card are the CPLD, the optical sensor and SDRAM. This has four connectors: power, I2C, SDRAM data bus and a connector for access to CPLD via JTAG. Fig. 5 shows the transceiver board. The on-chip antenna has a length of 165 mm, designed to work in the 433 MHz band. This card is designed to work with platforms of 3.3 V, as is the case of our node, or on platforms 5 V, as in the case of mid-range microcontrollers. Fig. 6 shows the differential robotic platform.

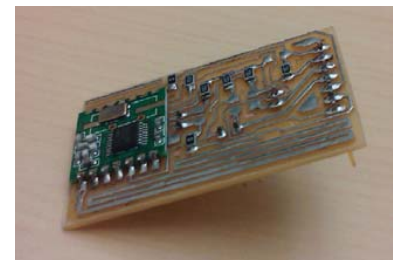


Fig. 5. ISM transceiver and board with antenna

5. FEATURE COMPARISON

All platforms were selected as background and starting point for our prototype design have two basic characteristics: capacity for local processing and transmission of processed information. This implies major restrictions with regard to the processing unit, optical sensor, storage and wireless communication features that are then used for this comparison. For us, one of the most important was the selection of the digital processor, as this has serious implications for the performance of the prototype and power consumption. Something similar happened to the external memory required. Table 1 shows a quick comparison of these key

features. This table shows a general trend as the use of 32-bit microcontrollers as main processing unit of the nodes, even on platforms with very discrete graphics resolutions. Another widely observed feature is the use of an embedded operating system for processing images. Our prototype is quite different from these standards, because it implements a parallel hardware through its firmware to increase system performance, which also reduces energy consumption and its final cost.

Table 1. Comparison of visual sensor platforms.

Platform	MCU	Memory	Radio	Resolution	Encoding	OS/Software
Cyclops [9]	ATmega129L (8 bit)	512 KB Flash, 64 KB SRAM	CC1000 38.4 Kbps	352x288	16bit YCbCr, 24bit RGB	Tiny OS
MeshEye [10]	AT91SAM7S (32 bit)	256 KB Flash, 64 KB SRAM	CC2420 250 Kbps	640x480	6bit greyscale, 24 bit color	No OS
Panoptes [11]	StrongARM (32 bit)	64 MB Flash	IEEE 802.11	640x480	32 bit RGB	Linux
Meerkats [12]	PXA255 (32 bit)	32 MB Flash, 64MB DRAM	IEEE 802.11	640x480	32 bit RGB	Linux
FireFly Mosaic [13]	LPC2106 (32 bit)	128 MB Flash, 64 KB RAM	CC2420 250 Kbps	352x288	8bit RGB, 8bit YCbCr	Nano-RK
MicrelEye [14]	FPSLIC with AT40K (32 bit)	1 MB SRAM	LMX9820A 230.4 Kbps	320x240	4:2:2 YUV	No OS (C)
CITRIC [15]	PXA270 (32 bit)	16 MB Flash, 64 MB SDRAM	CC2420 250 Kbps	1280x1024	8bit	Linux
Vision Mesh [16]	9261 ARM 9 (32 bit)	128 MB Flash, 64 MB DRAM	CC2430 250 Kbps	640x480	.	Linux
Camara Node Proposed	EPM570T100C5N CPLD	64 MB SDRAM	RF12B 11,52 Kbps	1280x1024	16bit 5:6:5 RGB	No OS (firmware)

6. RESULTS

We use the camera and the CPLD in the agent to process information from the environment. The initial goal was to achieve collective image-based navigation but with local processing of information. In addition, we were looking the least possible environmental interference, especially when thinking about human interaction (in real applications, many people feel uncomfortable and violated by the presence of camera). For this reason we decided not to transmit information, and take the minimum information necessary for the captured images.

Among the agents we define two types: leader and follower, similar to the types of bacteria identified

in [8] as virulent and non-virulent. Each type of agent has a specific behavior:

- The leader moves forward only when there is a large population of agents in the same region r_i (in our emulation, when there are two followers in the same region of the leader). There is only one leader in the group.
- Other agents, the type follower, move forward if the leader is in another region r_j .

The leader's behavior can be assimilated to that of the virulent bacteria, which are at rest when the bacterial population is below the threshold of population density h , and becomes virulent when the population exceeds this threshold [8]. On the other hand, the behavior of the followers can be assimilated to that of bacteria in reproduction, which in

con-ciencias

our case moving forward when the neighborhood threshold ρ is very low due the lack of the leader.

The values of these two variables are determined autonomously by the agents from the images captured by cameras. Images are captured in RGB 5:6:5 format as a string of 16 bits. The three matrices are processed individually to determine how many zones of red, blue and white are in the picture.

The decision is taken from the map that each agent has both the environment and other agents. To facilitate recognition, the three agents were built in different colors: the leader is white, a follower is red and the other one is blue. In addition, we were looking that these colors are not significantly was present in the environment, except for marks used to delimit the regions. This makes it easy to extract information from the presence of agents of the image, which can be made locally by each agent.

Since agents have no prior knowledge of the topology of the environment, navigation is done by identifying the regions using color marks on the floor. Any other obstacle is read by the agent through a touch sensor, which tells the agent that must move in a different direction. The random search and the color marks enable the agent to navigate the environment.

In our implementation in laboratory, we program the white agent as the leader, and the two others as followers. In the experiment shown in Fig. 6, we locate the follower agents in the region r_1 , and leader agent in the region r_2 . Blue and red agents responded to the position of leader, and moved to the region r_2 . Shortly after, the white agent responded and moved to the region r_3 .

This experiment is scalable, with more leaders and followers, making it possible to have different paths, allowing different types of applications.

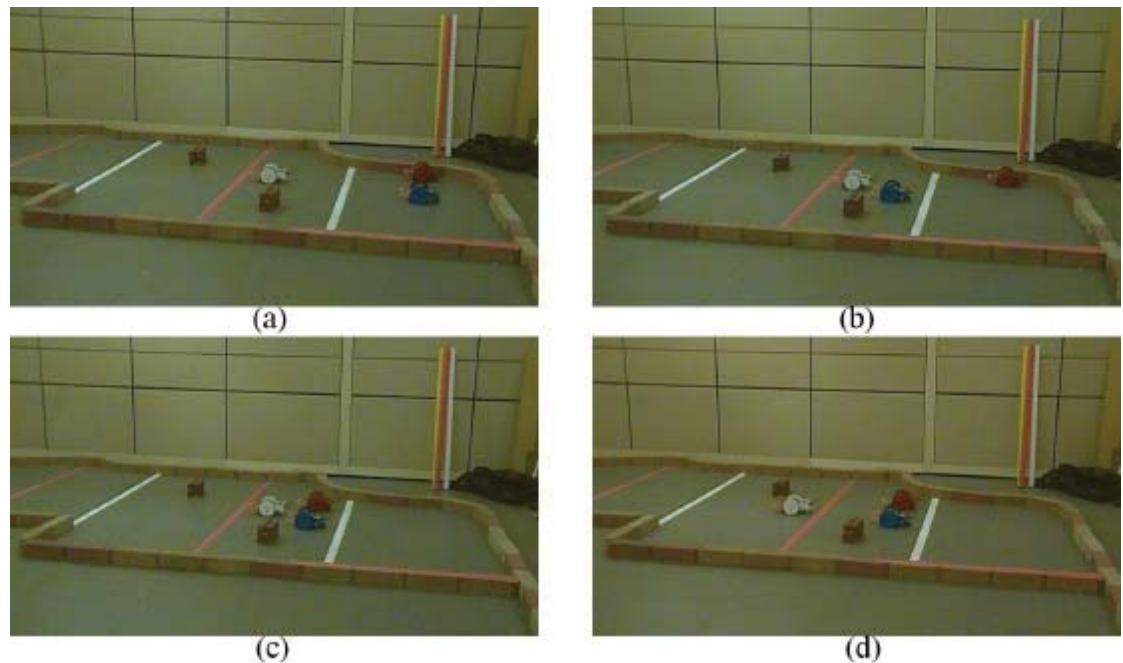


Fig. 6. Some snapshots of the experiment: (a) The agent with the camera is waiting in a region ahead of the other agents; (b) the blue agent moves to the new region; (c) the red agent moves to the new region; (d) the white agent detects the other agents and it moves to the next region.

7. CONCLUSIONS

In this paper, we have presented a novel visual sensor node for robotics and VSNs research. We formulate a special design methodology for the particularity of the future research, and this was used to derive the design of the prototype. The visual sensor node was design for VSNs research project in dynamic environments, so its versatility, size, weight and cost are considered important final design criteria. To show the minimalist features of the proposed design, we make a comparison of the main hardware features with other designs reported in research. The visual sensor node also provides an excellent platform for bio-inspired robotics research, especially in active vision evolution and distributed processing systems.

Using this platform with a total of three agents, it was possible to demonstrate the feasibility of con-

ducting collective navigation of agents in an unknown environment using only visual information embedded in the agent, and without performing complex image processing or local or external communication of the agents. Agents with simple structure, and without precise knowledge of the system model, and without performing state feedback, just using touch sensors and basic processing of visual information, may reflect complex biological behavior, as is the case of collective navigation.

The next step in this research is to develop a simple language of routes based on landmarks for navigation, so the agents can carry out specific tasks in the environment. For example, using color marks on the floor and walls, and in a complex environment, to achieve the classification of agents in different regions of the environment.

REFERENCES

- [1] G. N. Desouza and A.C. Kak, "Vision for Mobile Robot Navigation: A Survey", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, Issue 2, pp. 237-267, 2002.
- [2] M. Domínguez-Morales, P. Iñigo, J.L. Font, D. Cascado, G. Jimenez, F. Díaz, J.L. Sevillano and A. Linares-Barranco, "Frames-to-AER Efficiency Study based on CPUs Performance Counters", *2010 International Symposium on Performance Evaluation of Computer and Telecommunication Systems SPECTS*, pp. 141-148, 2019.
- [3] J. C. Czarnowski, "Minimalist Hardware Architectures for Agent Tracking and Guidance", *M.S. Thesis in Electrical & Computer Eng., University of Illinois at Urbana-Champaign*, 2011.
- [4] L. Yonghe and S.K. Das, "Information-Intensive Wireless Sensor Networks: Potential and Challenges", *IEEE Communications Magazine*, vol. 44, Issue 11, pp. 142-147, 2006.
- [5] S. Goto, T. Naka, Y. Matsuda and N. Egashira, "Teleoperation System of Robot arms combined with remote control and visual servo control", *Proceedings of SICE Annual Conference 2010*, pp. 1975-1981, 2010.
- [6] M. Perrier, "The Visual Servoing System CYCLOPE Designed for Dynamic Stabilisation of AUV and ROV", *Europe Oceans 2005*, Vol. 1, pp. 334-338, 2005.
- [7] S. M. LaValle, *Planning Algorithms*, Cambridge University Press, U.K., ISBN 978-0521862059, pp. 24-76, 2006.
- [8] F. H. Martínez, J.A. Delgado, "Hardware Emulation of Bacterial Quorum Sensing", *Proceedings of the 6th International Conference on Advanced Intelligent Computing Theories and Applications ICIC 2010, Lecture Notes in Computer Science, Springer-Verlag*, vol. 6215/2010, pp. 329-336, 2010.
- [9] M. Rahimi, R. Baer, O.I. Iroez, J.C. Garcia, J. Warrior, D. Estrin and M. Srivastaba, "Cyclops: In Situ Image Sensing and Interpretation in Wireless Sensor Networks", *Proceedings of the 3rd international*

- nal conference on Embedded networked sensor systems SenSys'05*, pp. 192-204, 2005.
- [10] S. Hengstler, D. Prashanth, S. Fong and H. Aghajan, "MeshEye: A Hybrid-Resolution Smart Camera Mote for Applications in Distributed Intelligent Surveillance", *6th International Symposium on Information Processing in Sensor Networks IPSN 2007*, pp. 360-369, 2007.
 - [11] W. C. Feng, E. Kaiser, W.C. Feng and M.L. Baillif, "Panoptes: Scalable Low-Power Video Sensor Networking Technologies", *Proceedings of the eleventh ACM international conference on Multimedia MULTIMEDIA '03*, 2003.
 - [12] J. Boice, X. Lu, C. Margi, G. Stanek, G. Zhang and K. Obraczka, "Meerkats: A Power-Aware, Self-Managing Wireless Camera Network for Wide Area Monitoring", *Distributed Smart Cameras Workshop - SenSys06*, 2006.
 - [13] A. Rowe, D. Goel and R. Rajkumar, "FireFly Mosaic: A Vision-Enabled Wireless Sensor Networking System". *28th IEEE International Real-Time Systems Symposium RTSS 2007*, pp. 459-468, 2007.
 - [14] A. Kerhet, F. Leonardi, A. Boni, P. Lombardo, M. Magno and L. Benini, "Distributed video surveillance using hardware-friendly sparse large margin classifiers", *IEEE Conference on Advanced Video and Signal Based Surveillance AVSS 2007*, pp. 87-92, 2007.
 - [15] P. Chen, P. Ahammad, C. Boyer, S.I. Huang, L. Leon, E. Lobaton, M. Meingast, O. Songhwai, S. Wang, Y. Posu, A.Y. Yang, C. Yeo, L.C. Chang, J.D. Tygar and S.S. Sastry, "CITRIC: A Low-Bandwidth Wireless Camera Network Platform". *Second ACM/IEEE International Conference on Distributed Smart Cameras ICDSC 2008*, pp. 1-10, 2008.
 - [16] M. Zhang and W. Cai, "Vision Mesh: A Novel Video Sensor Networks Platform for Water Conservancy Engineering". *2010 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT)*, vol. 4, pp. 106-109, 2010.