Stereo-MAS: Multi-Agent System for Image Stereo Processing

Sara Rodríguez[1], Juan F. De Paz[1], Javier Bajo[2], Dante I. Tapia[1], and Belén Pérez[1]

[1] University of Salamanca
[2] Pontifical University of Salamanca
{srg,fcofds,jbajope,dante,lancho}@usal.es

Abstract. This article presents a distributed agent-based architecture that can process the visual information obtained by stereoscopic cameras. The system is embedded within a global project whose objective is to develop an intelligent environment for location and identification within dependent environments that merge with other types of technologies. Vision algorithms are very costly and take a lot of time to respond, which is highly inconvenient if we consider that many applications can require action to be taken in real time. An agent architecture can automate the process of analyzing images obtained by cameras, and optimize the procedure.

Keywords: Stereoscopy, stereo cameras, artificial vision, MAS, agents, correspondence analysis, dependent environments.

1 Introduction

One of the greatest challenges for Europe and the scientific community is to find more effective means of providing care for the growing number of people that make up the disabled and elderly sector. The importance of developing new and more cost effective methods for administering medical care and assistance to this sector of the population is underscored when we consider the current tendencies. Multi-agent systems (MAS) and intelligent device based architectures have been examined recently as potential medical care supervisory systems [1][7][6][3] for elderly and dependent persons, given that they could provide continual support in the daily lives of these individuals.

The study of artificial vision, specifically stereoscopic vision, has been the object of considerable attention within the scientific community over the last few years. Image processing applications are varied and include aspects such as remote measurements, biomedical images analysis, character recognition, virtual reality applications, and enhanced reality in collaborative systems, among others.

The main topic of our research is part of a larger, global project whose objective is to develop a system for the care and supervision of patients in dependent environments, providing an environment capable of automatically carrying out location, identification and patient monitoring tasks. Such an environment would also allow medical personnel to supervise patients and simulate situations remotely via a virtual environment. In order to reach this objective, artificial intelligence techniques, intelligent agents and wireless technologies are used.

J. Cabestany et al. (Eds.): IWANN 2009, Part I, LNCS 5517, pp. 1256–1263, 2009.
© Springer-Verlag Berlin Heidelberg 2009

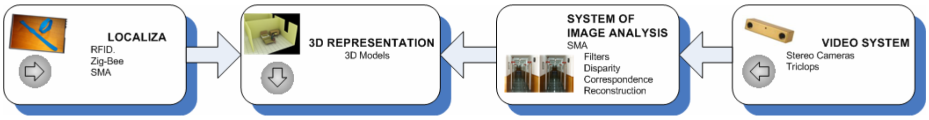

Fig. 1. System for the care and supervision of patients in dependent environments

Within the larger scope of the project, the majority of the current attention is being focused on video surveillance, analysis and 3D representation modules. Our research specifically focuses on the development of an agent-based distributed architecture that allows the visual information obtained by stereoscopic cameras to be processed.

The description of the overall project, along with a summary of the current state of the art is presented in sections 1 and 2 of this article. Section 3 focuses on the techniques used for analyzing the images. Once all the tools have been demonstrated, sections 4 and 5 will present the system proposal itself, and the results and conclusions that have been obtained.

2 State of the Art

This section will focus on stereoscopy and multi-agent systems (MAS), the primary mechanisms used in our system. Traditionally, the use of stereoscopy as a technique for reconstructing images has dealt with two problems. Using a two-dimensional pair of images with spatial coordinates (u,v), the left image (L) and right image (R), *the correspondence problem* attempts to find which two pixels $m_L(u_L,v_L)$ from the left image and $m_R(u_R,v_R)$ from the right image correspond to the same pixel M in three-dimensional space (X,Y,Z). Once these pixels have been found, *the reconstruction problem* attempts to find the coordinates for pixel M[10].

Addressing the correspondence problem is undoubtedly the most difficult task. Since there are generally several possibilities for choosing the element in image *R* that corresponds to the element in image *L*, the stereo correspondence problem is said to be ambiguous. It is necessary to determine which characteristics can be applied in order to reduce the ambiguity as much as possible.

The ultimate goal of *reconstruction* is to find the coordinates for pixel M (x,y,z) based on the coordinates from the projections for the same point over the images (u_L,v_L) and (u_R,v_R) [10].

The use of deliberative BDI (*Belief, Desire, Intention*) agents [2][4][8][17] is essential in the development of the platform we are proposing. Apparently, the human visual system deals with a high level of specialization when it comes to classifying and processing the visual information that it receives, such as reconstructing an image by texture, shadow, depth, etc. Computationally, it is difficult to compete with such specialization and separate from an image only the relevant information for any particular purpose. In response to this problem, we propose implementing an algorithm over a distributed agent-based architecture that will allow visual information contained in an image to be processed in real time. Because the system is capable of generating knowledge and experience, the effort involved in programming multiple tasks will also be reduced since it would only be necessary to specify overall objectives, allowing the agents to cooperate and achieve the stated objectives [14].

3 Image Analysis: Phases and Techniques

Techniques for three-dimensional image reconstruction can be classified as both passive and active[9]. Each type presents characteristics that are specific to the process of visual interpretation, but all of them present what could be considered to be common phases in our analysis. The idea is that after processing an image, we can define the resulting depiction as an iconic representation of the visible world. This "iconic model" does not need to represent all of the details from the image, only those that are necessary for developing the task we are interested in. The sequence of actions to take in order to complete the image analysis has been divided into the following modules (included within the *image analysis system* that is illustrated in figure 1):

The *data entry module* that captures the images. Define the number of cameras that will be used, their placement, etc. In this case, the camera that was chosen for this case is the Point Grey [15] Bumblebee2, model BB2-COL-ICX424. The *filtering model* reduces noise, improves contrast, sharpens edges or corrects blurriness. Some of these actions can be carried out at the hardware level, which is to say with the features included with the camera [15]. The *processing module* can be considered the heart of the system since it is where the algorithms are applied to analyze disparities and the correspondence of the stereoscopic pairs, and where the distance measurements for the camera are obtained. The measurements will prove useful in the next phase for reconstructing the image. For this phase, the *position recognition* and *3D representation modules* will model the image with the data that is received. We will focus on the *processing module*.

According to Marr and Poggio[11], there are three steps involved in image reconstruction. They are (i) select a specific pixel from the object in one of the images (preprocessing); (ii) find the same pixel in the corresponding image (correspondence analysis); (iii) measure the relative difference between the two pixels (disparity analysis and obtaining distance).

The aim of *preprocessing* is to identify the representative characteristics of each image [13] that will be used to complete the analysis. A *characteristic* is a relevant piece of information for completing the computational task. It can be an edge, corner or blob. Many vision algorithms apply characteristics detector as the first step, and as a result, there is a large number of detectors that have already been developed[5][19][12]. With artificial vision, edge detection is the most commonly used and reliable techique. The Canny algorithm[5] is considered one of the best methods for edge detection.

Regarding the problem for *obtaining correspondence* described in section 2, there are several strategies that can be classified in different ways[9].

Area based techniques consider the captured images to be a transferred two-dimensional signal. For each one of the pixels in the image, they try to make a transfer, minimizing certain criteria (correlation). One of the most simple techniques is the Sum of Absolute Differences (SAD), since it operates exclusively with whole numbers. Given a pixel with coordinates (x, y) in the left image, a correlation index C(x, y, s) is calculated for each displacement s for the correlation window in the right image. To calculate the correlation index,

$C(x,y,s)=\sum_{u=-w,v=-w}^{u=w,v=w}\left| I_l(x+u,y+v)-I_r(x+u+s,y+v)\right|$ where 2w + 1 is the size of the window centered on the pixel located at position (x, y) and I_l, I_r are the gray values for the pixels in the left and right images respectively. The disparity $d_l(x, y)$ between the left and right image pixels is defined as displacement s which minimizes the correlation index: $d_l(x,y) = arg\ min_s\ C(x,y,s)$. The bookstore that was used in the project (Triclops SDK) establishes a correspondence between the images using this technique.

The *techniques based on features* obtain high quality primitives (edge, segments, curves, regions, etc.) that store a set of properties that remain unchanged with the projection. Among the different methods based on characteristics, there is the Marr-Poggio computational theory[11], the Pollard-Mayhew-Frisby computational theory[16], or the techniques based on border segments. The PMF theory assumes two fundamental restrictions: the features contained in line “n” from the left image should likewise appear (allowing for certain disparity) in line “n” from the right image, thus the correspondence process would be carried out only between those features that are located on the same line in both images. The second restriction is given by the gradient disparity (GD) concept.

Rubio de Lemus [18] applies a comparison and evaluation methodology of search algorithms for correspondences in stereoscopy, showing their most relevant aspects.

The *disparity calculation* allows us to obtain the depth for each of the pixels on the image, obtaining one single image as the disparity map. Given that there is a direct correlation between the depth of the objects in an image and the disparity with a stereo pair, we can use the information from the disparity map as relative values for the depth of the objects.

4 Stereo-MAS

The process of stereoscopy vision is implemented over a distributed agent-based architecture, which allows it to run tasks in parallel using each service as an independent processing unit. It should be noted that because of the high computational costs, not all

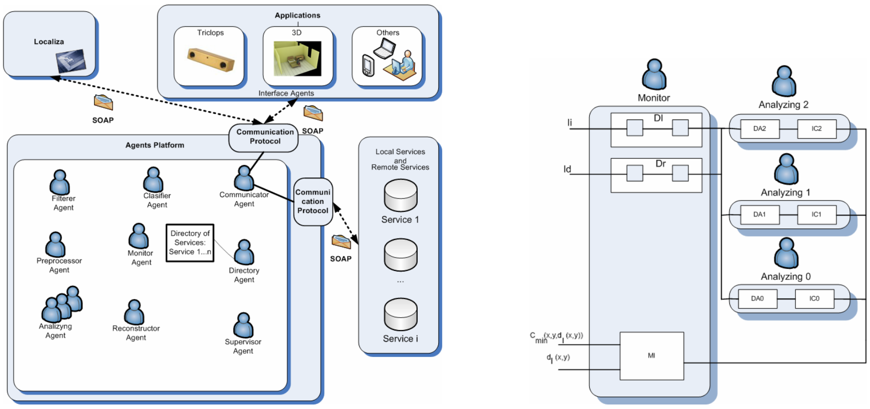

Fig. 2. (a) Diagram of the architecture. (b) Structure of the SAD algorithm

of the processes initially carried by the system are implemented to operate in parallel; only the correspondence algorithm. The system is comprised of a set of agents with defined roles that share information and services. The analysis of images supposes a complex process where each agent executes its task with the information available at each moment. Figure 2a illustrates the proposed multi-agent architecture.

The applications consist of each of the programs can be used for accessing the system functionalities. The applications are dynamic and adaptable to the context, and react differently to particular situations (for example, if it finds itself in one particular room or another, or if the user has some type of disability) and the type of service requested. The services represent the bulk of the system functionalities at the information processing, submission and retrieval levels. The system should count on a flexible directory of local and remote services that can be modified, added or eliminated dynamically and on demand.

The MAS function supports the coordination of agents that cooperate and share information that is necessary during the process of analysis. By breaking down the process of analysis described in the previous sections, the tasks generated are distributed among each of the system roles. The following roles were obtained: *Classifier*: in charge of receiving images, verifying the format, and changing it when necessary. *Filterer*: in charge of the basic filtering tasks. Coordinates filtering during the first phases either from the applications embedded in the cameras, or from the actual system of analysis. *Preprocessor*: in charge of extracting characteristics. It can use the extraction service deemed most convenient (Canny, etc.). *Monitor*: is in charge of controlling processes, assigns tasks to the other agents. Divides the images in blocks and assigns analysis tasks to the analyzing agents. The analyzing agents return the results and the *Monitor* sends the compressed information to the Reconstructor agent that is in charge of representing it three-dimensionally. *Interface*: this type of agent is designed to be embedded in user applications for direct communication with the platform agents. *Analyzing*: carries out the analysis of all correspondence. *Reconstructor*: gathers the information that has been analyzed and sends to the applications for 3D representation. *Communicator*: is responsible for the communications between applications and platform, and services and platform. It manages the requests for entry from the applications in order to be processed by the services. *Supervisor*: analyzes the structure and syntax of all of the messages entering and exiting the system, and supervises the correct functioning of the other system agents (periodically verify the state of all the agents registered in the architecture by sending ping messages). *Directory:* keep a list of all active services architecture.

This case has worked on a prototype, implementing an analytical component by following two of the techniques studied for the analysis of correspondence: the PMF algorithm, based on characteristics, and the pairing algorithm SAD, based on areas. In both cases we optimizes the use of the algorithms by taking advantage of the benefits that they provide to the platform agents, distributing tasks and parallelization, thus reducing the overall processing time.

PMF: The basis of the *PMF parallelization of the correspondence algorithm* was the restriction of the "correspondence between lines". Based on this premise, the correspondence algorithm was executed for independent processing units, using only the two blocks of data contained in the same "line number" for each image as a parameter. After the images were reduced to their basic features, they were divided into 20 blocks of 24 lines each (as the images used were 640*480). The length of each block is based on the size of the rows in the image to be processes. It is important to mention that we forced the formation of the blocks so that the 24 lines that mage up each block would not be listed sequentially in the image. That is, they were selected so that there are 20 lines of separation between them. An example of the contents of one block would be: A={0,20,40,60,80,100,120,140,160,180,200,220,240,260,280,300, 320,340,360,380,400,420,440,460} and continuing likewise for the remaining blocks. This convention was adopted with the intention of uniforming the workload distribution among the agents, since there are usually certain areas in the image with a greater concentration of information.

Once the blocks have been formed, the Monitor agent sends them for parallel processing in independent agents (Analyzing). When the last agent finishes its process, the information is compressed by the Monitor in order to create a 3D reconstruction.

SAD: The stereo calculation is made with the Triclops library[20], which defaults to a pairing algorithm based on the Sum of Absolute Differences (SAD). As it is the method used in the libraries provided by Point Grey [15], we also implemented it, along with a proposal for optimizing the algorithm via the parallelization of the tasks by the algorithm.

The algorithm used should be broken down into a series of sub-functions, which perform sequentially in the dataflow. These sub-functions are assigned to the Analyzing agents by the Monitor agent who is in charge of identifying the sub-functions that must be carried out in the correspondence analysis, and assigning the tasks to each of the Analyzing agents. The structure of the algorithm is shown in figure 2b. In order to have access to the neighboring pixels in the correlation window, once they have been scanned, the FIFO (First In, First Out) system is used in the Monitor agent for the left and right images (Il, Ir), represented as Dl and Dr. The absolute difference between the pixels in the right and left images is calculated in the modules DAl, where l = 1, 2, … dmax, and dmax is the maximum displacement in the correlation window. The correlation index is calculated in modules ICi. The calculation of the absolute difference and that of the correlation index are performed in parallel by the Analyzing agents. The disparity dl(x, y) is determined by comparing all of the correlation indexes that correspond to the dmax displacements. The MI module determines the minimum correlation index and provides the disparity dl(x, y) as well as the minimum correlation index Cmin(x, y, dl(x, y)), which can be used to calculate a confidence index for the pairing.

5 Results and Conclusions

The proposed agent-based architecture allows us to automate our analysis and optimize its performance. The use of agents provides a great deal of flexibility since they can help move the code to places where immediate action is required and answers can be obtained in execution time, autonomously and without interrupting services. Additionally, we reduce the need to program multiple tasks since it is only necessary to specify the overall objectives for the agents to cooperate and work together to achieve the stated objectives. The stereoscopic vision algorithms are implements in the architecture, allowing tasks to be carried out in parallel, using each service as an independent processing unit.

Initially, the algorithm we chose to optimize was the correspondence algorithm, since computationally it is the most expensive. We defined two proposals for the parallelization of correspondence techniques: the PMF algorithm, based on characteristics, and the SAD algorithm, based on areas. The “Analyzing” agent prototype was implemented because of its ability to carry out this type of analysis.

The first 5 images in figure 3 show the processing of a sample PPM (*Portable Pixmap Format*) image composed of three stereo images.

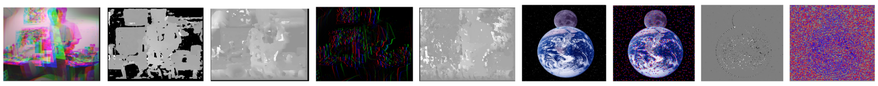

Fig. 3. Processed image results. From left to right: original image; disparity map using edge detection; disparity map obtained without using edge detection; use of the Sobel method on the first sample image; depth map obtained using Sobel (default mechanisms used in Triclops); original sample stereo image right; extraction of characteristics by number of lines n on the original image; edge detection and extraction of characteristics by number of lines m>n.

We can use the information extracted from the disparity map as values relative to the depth of the objects. The darkest values represent the farthest objects on the image, while the lightest values represent the closest objects. Preliminary results showed that the edge detection is one step that, while it may not be necessary, is very useful since it helps us with environments in which lighting conditions change notably. For this reason we decided to expand the prototype by adding more functionalities. In this case, we developed the Sobel [19] and Canny [5] edge detection algorithms. We can see how the disparity map reflects the depth of objects much better than the original image, even though it uses a gray-scale.

The next step was to develop the following correspondence analysis technique based on PMF characteristics. The last 4 images to the right in figure 4 illustrate the sequence of the results obtained.

In order to carry out an objective comparison, the algorithms were applied without enhancements. In order to perfect the results, we developed filters and masks that can adapt to the characteristics of the image (for example, converting JPG stereo images to PPM, and converting to color to gray scale), although the results are not presented in this paper. Additionally, we proposed an improvement in both algorithms using an agent architecture for their parallelization.

Acknowledgements. This research has been partially supported by the project PET2008_0036 and FEDER funds.

References

1. Angulo, C., Tellez, R.: Distributed Intelligence for smart home appliances. Red Española de Minería de Datos (2004)

2. Bajo, J., Corchado, J.M., de Paz, Y., de Paz, J.M., Rodríguez, S., Martín, Q., Abraham, A.: SHOMAS: Intelligent Guidance and Suggestions in Shopping Centres. Applied Soft Computing 9(2), 851–862 (2009)
3. Bajo, J., de Paz, J.F., de Paz, Y., Corchado, J.M.: Integrating Case-based Planning and RPTW Neural Networks to Construct an Intelligent Environment for Health Care. Expert Systems with Applications, part 2 36(3), 5844–5858 (2009)
4. Brenner, W., Wittig, H., Zarnekow, R.: Intelligent software agents: Foundations and applications. Springer, Secaucus, NJ, USA (1998)
5. Canny, J.: A computational approach to edge detection. IEEE Trans Pattern Analysis and Machine Intelligence 8(6), 679–698 (1986)
6. Corchado, J.M., Bajo, J., Abraham, A.: GERAmI: Improving the delivery of health care. IEEE Intelligent Systems 23(2), 19–25 (2008)
7. Corchado, J.M., Bajo, J., De Paz, Y., Tapia, D.I.: Intelligent Environment for Monitoring Alzheimer Patients, Agent Technology for Health Care. Decision Support Systems. Elsevier Science, Amsterdam (2006)
8. Corchado, J.M., Glez-Bedia, M., de Paz, Y., Bajo, J., de Paz, J.F.: Replanning mechanism for deliberative agents in dynamic changing environments. Computational Intelligence 24(2), 77–107 (2008)
9. Dhond, U.R., Aggarwal, J.K.: Structure From Stereo - A Review. IEEE Trans. on Systems, Man. and Cybernetics 19(6) (November/December 1989)
10. López-Valles, et al.: Revista Iberoamericana de Inteligencia Artificial 9(27), 35–62 (2005), ISSN: 1137-3601
11. Marr, D., Poggio, T.: A computational theory of human stereo vision. Proc. R. Soc. Lond., 301–328 (1979)
12. Marr, D., Hildreth, E.C.: Theory of edge detection. Proc. Roy. Soc. London B-207, 187–217 (1980)
13. Pearson, Don: Image Processing. McGrawHill, Great Britain (1991)
14. Pecora, F., Cesta, A.: Dcop for smart homes: A case study. Computational Intelligence 23(4), 395–419 (2007)
15. Point Grey Research Inc. (2008), http://www.ptgrey.com/
16. Pollard, S.B., Mayhew, J.E.W., Frisby, J.P.: A stereo correspondence algorithm using a disparity gradient constraint. Perception 14, 445–470 (1985)
17. Rao, A.S., Georgeff, M.P.: BDI agents: from theory to practice. In: Proceedings of ICMAS 1995, San Francisco, CA, USA (1995)
18. Rubio De Lemus, P.: Aplicacion De Una Metodologia De Evaluacion De Sistemas De Emparejamiento En Vision Tridimensional. Psicothema ISSN edición en papel: 0214-9915 5(1), 135–159 (1993)
19. Sobel, I., Feldman, G.: A 3x3 Isotropic Gradient Operator for Image Processing. In: Presentado en la conferencia Stanford Artificial Project (1968)
20. Triclops StereoVision System Manual Version 3.1. User guide and command reference. Point Grey Research Inc. (2003)