World Modeling Using Imaging Sensors in Middle-Size League

Behnam Azimi, Shahin Mohammadi, and Amin Yazdani

Satrap Group
Tehran, Iran
{Azimi, Mohammadi, Yazdani}@Satrapteam.ir
http://www.Satrapteam.ir

Abstract. This article is aimed to summarize the total process which is needed for image Acquisition,Processing and Recognition, and modeling the world using the information achieved from this process. This is a fundamental work, in order to feed the decision making block. During last few years, MSL is improved in electrical and mechanical execution, but yet there is no strategy manager or expert coach on this field. We here present a simple, but useful method for world modeling, and preparing data for further high-level processing.

1 Introduction

In last few years Robocup competitions has vast amount of achievements, but all of these efforts are leading in separate ways. In order to reach our ultimate goal in 2050, it's necessary to combine these works. For example, Simulation field focused on Artificial Intelligence of the Agents, but there is less focus in other leagues, like Middle-size, on designing multi agent systems[1]. First step towards implementing intelligent robots is to prepare a model of the outside world for their decision making module. In describing a world-model, there are two important facts that always should be balanced, accuracy and speed, which are related to the details needed in higher levels. Different sensors maybe used to gather information about outside world. This include INS, GPS, Sonar, Infrared, vision sensors and so on[2]. These sensors merging together, in order to cook data ready for interpretation. This article presents an omni vision-based method[3][4] for world sampling. Nowadays Omni vision is the best vision solution, for fast data acquiring in 360 degrees, but with less accuracy and more complexity to process.

Next Section describes an overview of the work and explain our approach. Section Three contains topics on image acquisition. Section Four explains useful methods for preprocessing the image. Section Five is an introduction to Segmentation and Object Recognition. Section Six is about representing the objects and obtaining information needed for updating world model.

2 Overview

As mentioned above, data received from sensors, have to be cooked for perception. What exactly have to be done? In this section we want to clearly define the problem and give a block diagram of the works which has to be accomplished.

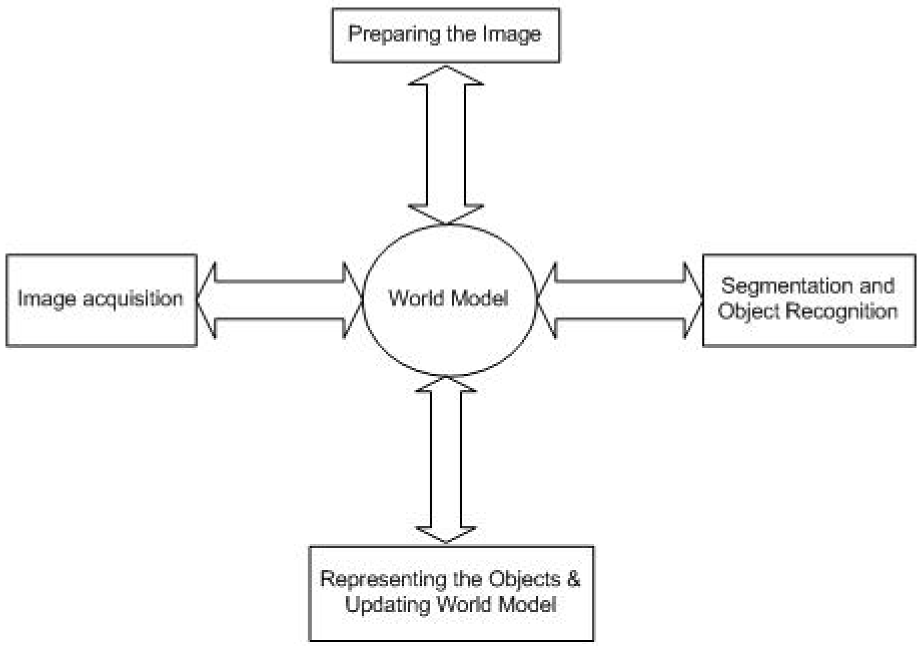

Fig. 1. Overall work Block Diagram

Agents are fundamental components in simulation and decision making. Each agent interact with its environment in three steps [5]:

Sense Gather data from environment and prepare it for further processing

Think Processing the cooked data and extract desired information from it, to model the outside world

Action Performing an action, depends on the state of the agent and its environment

In analogy to MSL, Sense cycle contains gathering data from sensors, which in this article specialized on "Omni Vision" sensors. Gathering data categorized in two steps, "Image Acquisition", which is receiving raw data from imaging devices, and representing it in appropriate format, and "Preprocessing", which is Noise removal, and image enhancements

Think cycle can clearly bind to "Segmentation and Object Recognition", which aimed to extract entities with some special features. It is notable here that decision making also classify like Think cycle, but it works in the different level.

Action cycle could finally be assumed as representing the objects and "Updating World Model". In higher levels it can observe in playing execution.

Next sections introduces a brief explanation of these major steps. In the final section, agent is able to make decisions based on world model created in previous steps.

3 Image Acquisition

Imaging devices converts the electromagnetic signal (light, heat etc) into a pixel value. Most robot vision systems uses CCD cameras which detect visible light. The output is usually analog which is converted to digital by a framegrabber. Output image can be represented in different ways [6]:

Representation of greyscale The framegrabber outputs a greyscale value of each pixel in the frame as a number from 0 to 255, where 0=black and 255=white.

Representation of colour in RGB color space Each color has a value in range 0 to 255, where 255,255,255=white and 0,0,0=black and 0,0,255=blue.

Representation of color in HSB color space Is an alternative for RGB color space. "Hue" value is most important factor to classify colors. It dominant wavelength, and does not change with a robot's relative position "Saturation" shows lack of whiteness. For example red has higher saturation than pink. "Brightness" is quantity of light returned (reflection).

The most important difference between RGB and HSB color space, is that RGB values changes significant with changing the intensity of reflected light. In an circumstance like Robocup, which all the agents moves frequently, light reflection changes and so we stuck into problems calibrating the robots. Nevertheless HSV is more stable, but in the other hand it's more expensive! The trick is to use the slope between RGB colors to classify them. Notice that these slopes varying in a linear manner with increasing the illumination (before 255 is reached by any of channels).

Next step, is to map pixel coordinates into real world coordinates[3]. The simplest method used for this purpose, is to sampling some points and interpolating the points. Depends on behaviour of distance function, least square method could be used to approximate other functions, like exponential.

4 Preprocessing

In this section, the goal is to image enhancement. This process includes couple of eyesight, Spatial Domain and Frequency Domain[7]. In first approach, enhancement done by the means of applying neighborhood operations. While the other one focuses on applying functions on image frequencies. To make it clear, reader should be familiar with FFT, to extract different frequencies in image signals. Couple of important preprocessing operations which used are as follows:

Sharpening By using the 2nd-order derivative of image- Laplacian operator and 1st-order derivative- Gradient. In frequency domain it called high-pass, because discards lower frequencies and let high frequencies extracted from FFT to pass.

Smoothing By simply averaging the neighborhoods. This called low-pass, because in frequency domain it just let lower frequencies to pass.

Now it's time to eliminate the noises from image. This noise may caused by hardware problems, or even noises appear in the playground. These noise can come from different distribution, nevertheless there are three classical methods used for image restoration:

Neighborhood Operations Simple averaging functions like discussed in smoothing part, can help removing the noises.

Mean Using mean value of each 3-square for value of centeral pixel.

Median Using media value of each 3-square for value of centeral pixel.

Now it maybe useful to say a word on usage of sharpening and smoothing filters. Sharpening filter focuses the edges of an image, and as we see in the next section, it can be used to extract objects from images. Smoothing help us avoiding noises, but it has a side effect that reduce the edge visibility.

5 Segmentation and Object Recognition

In this section, we summarize the method for extraction of objects from an image, by focusing on some special properties of them. Four mostly popular method exists which are presented here[8][7].

5.1 Histogram Analysis

The most basic segmentation algorithm, is based on thresholding of a gray level image. The success of this approach hinges upon whether suitable thresholds exist and whether they can be distinguished from the image histogram. This method usually lead to a bimodal or multi-modal histogram, and can be handle using k-mean algorithm.

5.2 Region Growing and Split-and-Merge Algorithms

Region growing algorithms start with a number of seed pixels or seed regions and grow these regions by adding to a region previously unassigned neighboring pixels that correspond to some similarity criterion for that region. The rule employed for accepting/rejecting a candidate pixel for merging with the neighboring region may be defined as either on the basis of a pixel and its immediate neighbors or on its relationship to the mean of a region. Growing algorithms that link neighboring pixels by analyzing contrast use edge-strength to measure the compatibility between neighboring pixels (edge strength) and thus require a completely closed boundary of high-edge strength pixels, otherwise neighboring regions will be merged. Growing algorithms that measure similarity by comparing a pixel with some measure of the whole region (e.g. mean value) are more robust but impose a stronger homogeneity criterion upon the regions being grown. Split-and-merge algorithms have been reported . This method proceeds to successively divide an image into smaller non-overlapping regions if some similarity criterion is not met, otherwise no split of that region is made. The end result of the splitting is an over-segmented image. A merging procedure is then applied to merge neighbouring regions under the same homogeneity predicate that was used for splitting.

5.3 Gradient Operators and Edge finding

Proposed the derivative-of-Gaussian filter as a near-optimal filter with respect to three edge-finding criteria: (a) good localisation of the edge, (b) one response to one edge and (c) high probability of detecting true edge points and low probability of falsely detecting non-edge points. implements a filter with an impulse response similar to that of the derivative of Gaussian, but which lends itself to direct implementation as a recursive filter. This requires fewer coefficients than a transversal or non-recursive filter that has the same impulse response, yielding faster convolution, especially when the width of the filter becomes large. The benefit of these filters is that they impose a smoothing upon the image and as such are band-pass filters rather than simple differentiators and thus they are less sensitive to noise. Smoothing is controlled by a scale parameter which may be tuned for each application. Edges are extracted from the gradient image by choosing pixels with a gradient magnitude that is a maximum in the local direction of the gradient[9].

5.4 Motion Detection using Image Derivatives

This section discuss about extracting motion object from either in a steady state field, or objects moving with constant velocity relative to their environments[9]. This process accomplished using derivation of consequent images. For example of 1st class, reader could have a look at Robocup @Home robot, which has to follow their leaders. For the 2nd class, police speed camera's can be mentioned.

6 Updating World Model

After all these information extracted from the imaging sensors, we have to finally represent them to suitable format, which can be used by decision making module.

This here is a question: Which of these information are useful for representing the object? In the other words, which details are necessary for decision making? In order to answer this question, several efforts have been done. As a result of these researches, some special properties usually come handy. Here are some samples:

Center of Mass It can be use to approximate an object coordinate, and use it for path planning.

Bounding Box It can be uses to find the edges of objects, to avoid the collisions in agent motion.

Total Area It can be used to estimate the distance of the objects from the agent.

After representing the objects, it's time for updating the world model.

7 Future Works

In this article, we focused on designing a method for extracting objects from images and modeling the world. But in fact it can have a two way communication between world model and image processing. What it exactly means? means that we can create an initial model of agents environment, and use it to reduce the computations in next steps. For example, if we know that ball is in coordinate (x, y), we expect to find the ball in rectangle (x-MaxVel, y - MaxVel)-(x+MaxVel, y+MaxVel) in next cycle. A more complex model needed for developing supervised pattern recognition, and using a knowledge base to help image understanding. This will be our field of interest in next researches.

References

1. Ferber, J. In: Multi-Agent System: An Introduction to Distributed Artificial Intelligence. Harlow: Addison Wesley Longman (1999)
2. H. Bau, N. F. DeRooij, B.K. In: Sensors, A Comprehensive Survey, Mechanical Sensors. Volume 7. Wiley (1993) 692
3. Felix v. Hundelshausen: An omnidirectional vision system for soccer robots. Master's thesis (2001)
4. Winters, N., Santos-Victor, J.: Mobile robot navigation using omni-directional vision (1999)
5. Riley, P.: SPADES: System for parallel agent discrete event simulation. AI Magazine **24**(2) (2003) 41–42
6. Zarit, B.D., Super, B.J., Quek, F.K.H.: Comparison of five color models in skin pixel classification. In: RATFG-RTS '99: Proceedings of the International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, Washington, DC, USA, IEEE Computer Society (1999) 58
7. Gonzalez, Woods: Digital Image Processing. 2nd edn. Prentice Hall (2002)
8. Boyle, M.S.V.H.R.: Image Processing, Analysis, and Machine Vision. 2nd edn. International Thomson Publishing (1999)
9. Spacek, L.A.: Edge detection and motion detection. Image Vision Comput. **4**(1) (1986) 43–56