VISION BASED REAL-TIME LOCALIZATION OF MULTIPLE MOBILE ROBOTS

António Paulo Moreira, Armando Sousa and Paulo Costa
DEEC-ISR, FEUP, Porto, Portugal
Phone Int +351 22 508 14 00, Fax Int +351 22 508 14 40, E-mail: {amoreira, asousa, paco}@fe.up.pt

Abstract: This paper describes the design of a vision based real-time localization system capable of tracking multiple robots in a real-time fashion. A fuzzy framework is used to describe the system in a way that is implementation independent while being easily deployed using standard hardware and software. The proposed architecture allows the use of multiple cameras and can be easily implemented in a parallel system. One of the advantages of this system is that it provides not only an estimate of a robot position but also a certainty measure of that estimate. That can be very useful, for example, if we want to feed the estimate to a Kalman filter.

Keywords: mobile robots, vision, localization, fuzzy

1 INTRODUCTION

The introduction of mobile robots in a certain environment should not require extensive changes to accommodate them. One way to alleviate those changes is to use existing systems like surveillance cameras. If those cameras can be used to track the position of the robots, more evolved localization systems can be left out. The system presented here was successfully used to track the position of up to ten robots during some RoboSoccer competitions [Costa et al., 1999].

Formulating the operations performed by the system using a Fuzzy Set framework allows for a very implementation independent description. The key strategy for an efficient system is to restrict the operations performed on all pixels to a few that can be represented by a table lookup. Another key advantage is the use of the marker size to estimate the quality of that observation. Spurious pixels will appear in small clusters that are easily dismissed as outliars.

2 SYSTEM DESCRIPTION

The localization of a mobile robot in an image can be simplified if a known marker is present on the robot. One likely candidate is a coloured ball placed on top of the robot. One advantage of that shape is that it is invariant with the observed angle. If the maximum and minimum distance between the camera and the robot are similar, the marker will have a fairly consistent size over all the possible robot positions. That way the perceived size can be used, not to estimate distance but the quality of the observation.

The digital representation of an image can be viewed as a pixel matrix where each pixel is a vector storing the intensity of each basic colour [Pratt, 1991].

We can define the function:

$$\begin{aligned} rgb:\ & I \rightarrow \Re^3 \\ & p \rightarrow (r, g, b) \end{aligned} \qquad (1)$$

that maps each pixel with the associated colour. As we have a digital representation, the r, g and b components will be discretized.

It is considered that 8 bits for each component is enough so that the human eye cannot perceive that effect. That means each pixel requires 3 bytes to be stored. Some gain in the storage space needed can be achieved if only 5 bits are used for each component. The resulting 15 bits can be stored in only 2 bytes thereby reducing the storage requirements to 2/3. The main advantage of this compromise is that a function that maps any color to a certain quantity:

$$\begin{aligned} f:\ & \Re^3 \rightarrow \Re^n \\ & (r, g, b) \rightarrow X \end{aligned} \qquad (2)$$

can be represented by a table with no more than 2^{15} entries. For example, if X can be stored in 4 bytes, we will have a table with 128 Kbytes. That is a very reasonable value for the available computers. This technique can be used to speed up the calculus of some transformations.

The space of all possible colour can be perceived as a cube with vertices (0,0,0) and ($max(r)$, $max(g)$, $max(b)$), with r,g and b discrete we have a three-dimensional grid embedded in the cube. Let’s define the set F_i as the set of colours that the i-th marker can show on the image.

To build this set from real observations we need a histogram that counts the number of times a certain colour was seen in a part of the image that represents the marker. This histogram H_i can be defined as a function;

$$\begin{aligned} H_i: \quad & \Re^3 \rightarrow N_o^+ \\ & (r, g, b) \rightarrow h \end{aligned} \tag{3}$$

With H_i, F_i can be represented using a function m_{Fi}. This function as seen in [Zadeh, 1965] and [Kosko, 1992] represents the degree in each element of the universe belongs to the set.

Composing H_i with the function r that is defined as follows:

$$\begin{aligned} r: \quad & \Re_o^+ \rightarrow [\,0\,,\ 1] \\ & h \rightarrow m \end{aligned} \tag{4}$$

With:

$$r(h) = \begin{cases} 0 & \text{if } h < T_m \\ \dfrac{(h - T_m)}{(T_M - T_m)} & \text{if } T_m \leq h \leq T_M \\ 1 & \text{if } h > T_M \end{cases} \tag{5}$$

we can obtain m_{Fi}:

$$m_{Fi}(r, g, b) = r(H_i(r, g, b)) \tag{6}$$

The parameters T_M and T_m represent the number of times a colour must be seen to have full membership to the set Fi (T_M), and the number of times a colour must be seen to have a membership degree above zero (T_m).

Figures 1 and 2 show the sets found for four markers.

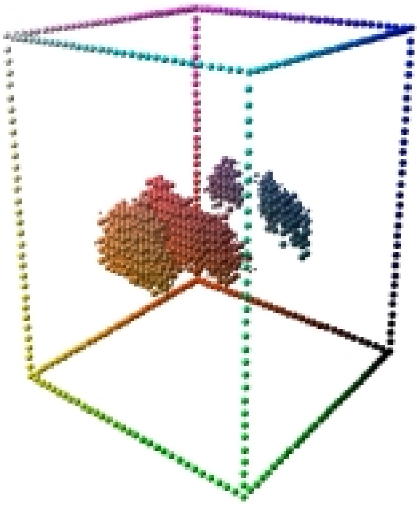

Figure 1. The sets $F_{1...4}$ represented in the *rgb* space

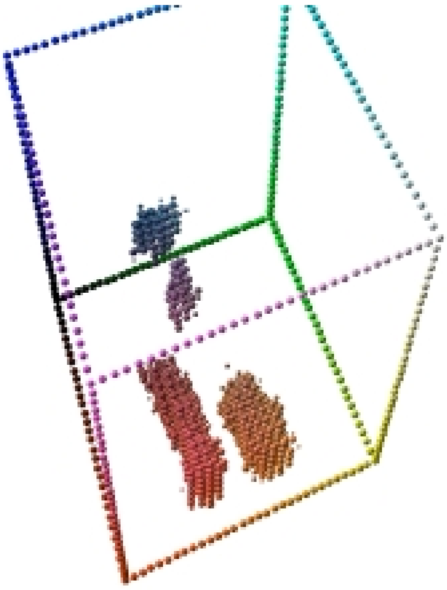

Figure 2. Another view of the fuzzy sets $F_{1...4}$ in the *rgb* space

Ideally, the sets F_i should be disjoint:

$$\forall i \neq j,\ F_i \cap F_j = \phi \tag{7}$$

In practice, a small amount of overlap should be expected.

2.1 Pixel Classification

All the pixels from an image form the set U_{img}. A fuzzy set G_i is the set of pixels from U_{img} that have a colour somewhat similar to the colour of the i-th marker. This set is defined by the function m_{Gi}:

$$m_{Gi}(p_{xy}) = m_{Fi}(rgb(p_{xy})) \tag{8}$$

There is one of these fuzzy sets for each type of marker.

This is the only operation that must be performed over all pixels from the image. Afterwards, any pixel that does not belong or belongs in a low degree to any of these sets can be considered uninteresting and discarded. All subsequent operations will be performed in a small number of pixels lowering dramatically the processing time for each image. More, this operation is strictly local, so it can be performed while the pixels are being fetched, by multiple processors or even performed in hardware.

2.2 Pixel aggregation

Probably, there will be multiple pixel clusters for each colour. The sets G_ic_j should contain pixels of the *i*-marker and the *j* cluster. The clustering can easily be done with a greedy algorithm aided by the fact that usually the clusters will not overlap. So we can create a new cluster each time that for two pixels p_1 and p_2:

$$l^2(p_1,p_2) > R_i m \qquad (9)$$

Where l^2 is the Euclidean norm computed for the difference between the world position of each pixel.

The constant $R_i m$ can be made variable with the position in the image in the case of severe distortion.

2.3 World-Image Mapping

If the camera axis is not orthogonal to the floor or if the lens introduce too much barrel distortion the mapping between the world coordinates and the pixel position in the image can be extremely non linear [Hecht, 1991]. That requires a function:

$$\begin{aligned} m: \quad & Nx \times Ny \rightarrow \Re^2 \\ & (u, v) \rightarrow (x, y) \end{aligned} \qquad (10)$$

that must be able to cope with all those non-linearities. Also, it is important to have the inverse of that function, or a reasonable approximation. The following function:

$$m_x(u, v) = a_1 + a_2u + a_3v + a_4u^2 + a_5v^2 + a_6uv + a_7u|u| + a_8uv^2 \qquad (11)$$

$$m_y(u, v) = b_1 + b_2u + b_3v + b_4u^2 + b_5v^2 + b_6uv + b_7v|v| + a_8vu^2 \qquad (12)$$

can compensate most of the mentioned non-linearities. The parameters, a_1, a_2, …, a_8 and b_1, b_2, …, b_8 can be estimated from known points in the image. At least 8 of those points are needed but more points can improve the estimation accuracy [Strejc, 1980]. Likewise, the inverse function can use a similar structure. While the result is not the true inverse, the approximation can be close enough.

While more terms can be used, these are the enough, and adding more will worsen the numerical problems from this estimation.

We can see in the next figure this function mapping a rectangle in an image with a considerable barrel distortion.

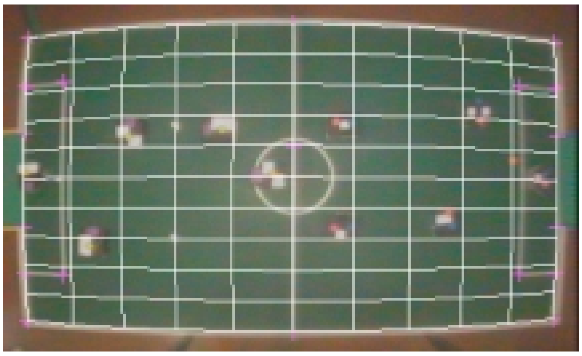

Figure 3. Correction of the barrel distortion

2.4 Cluster quality

From each set G_ic_j, it is possible to extract a marker observation. An estimate for the marker center can be obtained by:

$$\hat{x}_{ij} = \frac{1}{M(G_{iCj})} \sum_x p_{xy} \ , \ p_{xy} \in G_{iCj} \qquad (13)$$

$$\hat{y}_{ij} = \frac{1}{M(G_{iCj})} \sum_y p_{xy} \ , \ p_{xy} \in G_{iCj} \qquad (14)$$

The corresponding covariance:

$$C_{xy,ij} = \begin{bmatrix} c_{xx} & c_{xy} \\ c_{yx} & c_{yy} \end{bmatrix} \qquad (15)$$

can be constructed accounting for:

- The x coordinate covariance can be assume equal to the y coordinate.
- The x coordinate estimate is mostly independent from the y coordinate.

The covariance can now be written:

$$C_{xy,ij} = \begin{bmatrix} c & 0 \\ 0 & c \end{bmatrix} \qquad (16)$$

and c can be approximated by:

$$c(x) = \begin{cases} \left(\frac{ek}{2}\right)^2 x^2 e^{-kx} & if\, x \geq M^{min} \\ 0 & if\, x < M^{min} \end{cases} \qquad (17)$$

with

$$k = \frac{2}{M^{opt}} \qquad (18)$$

This function has a maximum at M^{opt}, meaning that the maximum certainty is achieved when the cluster has M^{otp} associated pixels. A marker with more or less pixels probably should be less trusted because the center estimate will have lower precision.

4 RESULTS

This system was used to track the position of 10 robots in the EuroRobocup in Amsterdam. Two cameras were used, both connected to a 400MHz Pentium II. The PAL signal was acquired at 50 frames per second with a resolution of 384 x 288 pixels. All the processing was located on that PC. The position of each robot could be tracked without any drop in frames and there was even some spare processing time to compute the trajectories for our robots. As five of the robots carried two markers and another five carried one marker, the total number of tracked markers reached 16, counting the ball that was also being tracked.

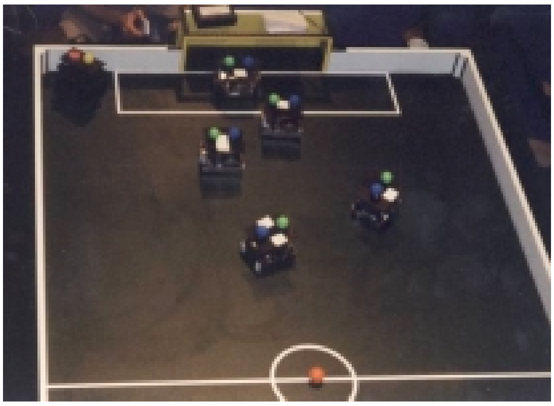

Figure 4. A team of Robots with their markers

5 CONCLUSIONS AND FUTURE WORK

The proposed system, presented in a Fuzzy Set framework, can be implemented in an efficient way. Other implementations are easily pursued due to the neutral way it is presented. The efficiency, in terms of computing power allows the use of such system in a real time environment. That makes possible its use in a closed loop solution for controlling multiple robots. It is very simple to add extra image sources so that we have a distributed vision system reaching an extended area.

Some issues like the optimal size for the markers and the ideal colours are very environment dependent.

Right now, the calibration process is human assisted and must be done for all possible lightning conditions. It should be possible to start with a reference calibration and let the system adapt the calibration for slow changing lightning.

There are some strong assumptions behind the proposed system. Some, like the relative invariance of the marker size over the image, can be relaxed if some constants are made dependent on the marker position over the image. That dependence could be formalized and tested.

REFERENCES

[Costa et al., 1999] Paulo Costa, António Moreira, Armando Sousa, Paulo Marques, Pedro Costa and Aníbal Matos, 5dpo Team Description, Robocup 1999 - Robot World Cup Soccer Games and Conference, Stockolm, Sweeden, pp 85-89, 1999.

[Strejc, 1980] V. Strejc, Least Squares Parameter Estimation, Automatica, Vol 16, pp. 535-550, Pergamon Press Ltd, 1980.

[Pratt, 1991] William K. Pratt, Digital Image Processing, Second Edition, Wiley-Interscience, 1991.

[Kosko, 1992] Bart Kosko, Neural Networks and Fuzzy Systems, a Dynamical Systems Approach to Machine Intelligence, Prentice Hall, 1992.

[Hecht, 1991] Eugene Hecht, Óptica, Fundação Calouste Gulbenkian, 1991.

[Zadeh, 1965] L. A. Zadeh. Fuzzy Sets.Inform Control, 1965.