MOTION PLANNING OF HUMANOID ROBOT BASED ON EMBEDDED VISION

ZHONG Qiu-bo[1, 2*], ZHAO Jie[1] and TONG Chun-ya [2]

[1] State Key laboratory of Robotics and System, Harbin Institute of Technology, Harbin, 150001, China;
[2] School of Electronic and Information Engineering, Ningbo University of Technology, 315016, China.

Email: chunya.tong01@gmail.com

ABSTRACT

According to the characteristics of humanoid robot, a control method of motion planning based on embedded vision is present. By using HSI color space model, an improved algorithm based on look-up table to accelerate the identification of image is proposed. The image is filtered by floating threshold clustering and the result is used to plan the robot motion by finite state machine (FSM). Results are shown by digital simulation and validated in humanoid robot penalty kick.

Keywords: HSI color space, Floating threshold clustering, Motion planning.

1. INTRODUCTION

Since the 21st century, the robot research has made the significant progress. Autonomous humanoid robot is very popular in the robot research area and attracted many researchers. FIRA (Federation of International Robosoccer Association) is increasingly focusing on humanoid robot competitions to promote the project. The embedded vision system is the base of autonomous humanoid robot system and it plays a key role in the promotion overall performance of robot. Because robot target identification, location, path and motion planning must be built on the basis of visual system.

Humanoid robot race are commonly used a sign based on color feature, which for target identification and location tracking. HIT-2 humanoid robot using PDA (Personal Digital Assistant) as its image processing, data processing can reduce the burden of low-level processing for the robot. While the images processed through the PDA, features will be more evident, this provides a reliable basis for the robot target tracking and self-orientation. Technology mainly used in image segmentation is based on the principle of threshold segmentation algorithm.[1] Whatever the segmentation algorithm, the prerequisite is to use a reasonable color space.[2] There is no color space which can replace the others applying to all color image segmentation. Therefore, right color space need to be choose according to the competition venue features for robot soccer

XIA etc [3]present the progress of research on motion planning for humanoid robots, he divided the methods for motion planning into off-line path planning based on opportunities and game theory and on-line based on sensor information fusion. Rolling planning strategy to solve the path planning problem under dynamic environment is used on Asimo by Philipp etc[4]. Using this method, the robot not only can adapt to the dynamic changes of the environment,

but also can greatly reduce the scale of the problem solving and achieve a very good experimental result by a local optimum in the path planning space instead of global optimization.

In this paper, HSI color space is used to extract the pixel feature, the traditional look-up table method of recognition has been improved and image processing results are got through the variable threshold clustering. Finally, path planning for humanoid robot in penalty game is present by rolling planning strategy based on deterministic FSM.

2. EMBEDDED VISUAL PROCESSING

In RGB color space, since the values of R, G, B have a strong correlation, RGB color space is not suitable directly as image processing

HSI color space is widely used in machine vision research areas [5]. Where HSI respectively represents Hue, Saturation and Intensity. RGB to HSI conversion formulas [6] is a non-linear reversible.

$$H = \arctan'(\sqrt{3}(G - B), 2R - G - B)$$
$$I = \frac{R + G + B}{3} \tag{1}$$
$$S = 1 - \frac{\min(R, G, B)}{I}$$

$$Th_{sat}(I) = 1.0 - \frac{aI}{255} \tag{2}$$

By formula (1) we can get that the value of hue H is ranged from $[0, 2\pi]$; Saturation S represents the depth or purity of a color, whose range is [0, 1]. I, the general range of intensity I is [0,255]. In the case of S = 0, the value of I is growing from

0 and the color is from black to white varying in gray-degree. With the value of saturation S, influence on that pixel by value of hue and intensity can be obtained. By formula (2), we also can get that when the value of s is bigger than $Th_{sat}(I)$, the value of H is going to be as the main feature of that pixel and vise verse is the value of I, where a is an empirical value.

2.1 Recognition of the color space based on improved Look-up table

Suppose the visual system needs to identify the N kinds of different colors, and each kind of color represented by $COLOR_i$ is corresponding to an interval of HSI, which represented as $\{ [HMIN_i, HMAX_i], [SMIN_i SMAX_i], [IMIN_i, IMAX_i] \}$, this equals a 'box' with its length, width and height are $\Delta H_i, \Delta S_i, \Delta I_i$. Where $\Delta H_i = HMAX_i - HMIN_i$, $\Delta S_i = SMAX_i - SMIN_i$, $\Delta I_i = IMAX_i - IMIN_i$. N kinds of colors are corresponding to N 'box'. Clustering of pixels based on color is to judge whether the coordination of every pixel in the color space is just in some 'box'. For instance, if a pixel represented as P_i is in the BOX_i, P_i is belonged to $COLOR_i$.

Commonly used method of clustering is to compare value of H, S, I for each pixel with N kinds of color corresponding to the HSI interval determining whether in the interval or not. For every pixel, the times of comparison operation for clustering operation need to be N × 6 integer variables. It is inefficient if the same clustering operation to all the pixels of the color images is implemented and which affect the real-time target recognition.

In the HSI color space, hue is one of the biggest differences variables among the various objects. Therefore, the formula of hue is unchanging.

When the HSI color cone changes along the radial orientation, which saturation of color changes, the color attributes are the same, but a change in purity, that is by the color cone color, in the outer part of the net is bright comparison with close to the center part being impure, and gradually moving gray. In fact, if the purity of an object is relatively high, only lower bound of saturation is consider to filter out variegated instead of defining the upper bound. Therefore, only value of I is considered to define to prevent from judging by mistake. By this method, only N × 6 integer variables need to be comparison for execution of every pixel.

As we known, using look-up table approach can further speed up the clustering speed. This approach adopts operation of array indexes instead of N×4 times comparison operation, which greatly improves the efficiency of computing for clustering algorithm. However, the disadvantage is its too much storage space. According to this problem, an algorithm based on improved look-up table is present to complete the color space clustering.

The use of projection transformation can project the pixels of image in the color space corresponding to the coordinates onto the three axes of color components respectively. Thus, the number of dimensions of clustering an array of look-up table reduced to one dimension. J kinds of color space can be expressed by three arrays of unsigned integers with j-th bit. (HClass[0…255], SClass[0…255], IClass[0…255]). Where array 'HClass, SClass and IClass' represent each value of color respectively. The j-th bit of elements in every array denotes the binarization results of the j-th color. The clustering of pixels is achieved through bitwise AND operation for arrays.

The nature of approach of improved look-up table present in this paper is based on the multi-threshold for color image segmentation. There are two advantages of this approach.

1. By adopting j-th bit of unsigned integers to represent the elements of array, only two times for bitwise AND operation can determine the color category of pixels, which greatly increased the speed.

2. By projecting three-dimensional color space onto a one-dimensional axis, only three one-dimensional arrays can express the color space and predefined color category, which greatly save the memory space.

2.2 clustering based on improved variable thresholds seed fill algorithm

Clustering method commonly used requires repeated image scanning, which cannot meet the real-time requirement based on the embedded vision system. Seed fill algorithm is commonly used in a class of interactive graphics filling algorithm. It can generate all the clustering just scanning the image for once. The most advantages of this algorithm are fast and the ability of filling the region with variable complex boundary.

If the light is uniform and the color of object is purity, the application of this method can be a very good effective recognition. However, if the light is not uniform, the effect of recognition will be worse. Although adjusting the threshold of hue can improve the effect of recognition under this bad situation, if the threshold of hue is changed to lower, only partly object can be recognized or even misrecognition. Vice verse, if the threshold is higher, interference color blocks is going to affect the accuracy of recognition. Therefore, an improved clustering seed fill algorithm based on variable threshold is proposed. In the robot game, target objects are solid. When the algorithm scans the initial pixel of image, lower threshold of hue *Thl* is adopt and when clustering starts, higher threshold of hue *Thh* is used. The numbers of pixels belonged to lower and higher threshold of object are recorded, and the value λ described in formula (3) is used to filter the interference region. If the value.. is lower, this means the number of pixels in the lower threshold is smaller than in the higher threshold, and the probability of result belonged to the interference region is higher. Vice verse, if the value λ is higher, this will bring to the higher probability of the object. The value λ can be gotten through experiments described in table 1 bellow.

$$N_{Thl} \xrightarrow{\quad Numbers \quad} Thl, \quad N_{Thh} \xrightarrow{\quad Numbers \quad} Thh, \quad \frac{N_{Thl}}{N_{Thh}} = \lambda \tag{3}$$

3. MOTION PLANNING FOR HUMANOID ROBOT BASED ON PENALTY KICK

3.1 Motion planning and decision for penalty kick based on FSM

A series of trajectories can be combined for humanoid robot by the basic gait generated off-line.

For instance, in the motion of penalty kick for humanoid robot, the models of FSM include six states such as finding ball, approaching ball, adjusting the shoot direction, adjusting the foot, shooting and goal.

1. Finding ball state: mark q_0, it is the initial state of the FSM. When the ball is not in the visual range, system will go into this state.

2. Approaching ball state: mark q_1, when the distance between robot and ball is longer, system will go into this state.

3. Adjusting the shoot direction state: mark q_2, when the robot approaches the ball, begins to adjust the shoot direction by recognized goal and keeper information.

4. Adjusting the foot state: mark q_3, when the robot has adjusted the shoot direction and begins to select the foot for shooting.

5. Shooting state: mark q_4, when the condition for shooting is satisfied.

6. Goal state: mark q_5, when the robot finishes shooting, the system will go into this state. This is the final state of FSM. The robot will wait for the next turn of penalty kick.

The collection of states above consist of the collection for definitely finite sate machine Q, $Q = \{q_0, q_1, q_2, q_3, q_4, q_5\}$, where q_0 is the initial state and q_5 is the finial state for FSM.

When constructing the alphabet for FSM, data of object availably captured by visual system is processed first, and abstraction for these data is adopted. As an input of alphabet for FSM, seven input letters is designed as follows:

1. w_1: the robot found the ball, however, the distance is longer;

2. w_2: the ball is out of the vision of robot;

3. w_3: the distance is enough for the robot to adjust the shoot direction;

4. w_4: the shoot direction is adjusted;

5. w_5: condition for shooting is satisfied;

6. w_6: success for shooting;

7. w_7: failed for shooting.

The condition of states transition consists of input alphabet of FSM Σ, $\sum = \{w_1, w_2, \ldots w_7\}$. Inputting the alphabet will result in the FSM to transfer according to the following state transition function, which described as formula (4):

$$\delta(q_0, w_1) = q_1, \delta(q_0, w_3) = q_2, \delta(q_0, w_4) = q_3,$$
$$\delta(q_1, w_2) = q_0, \delta(q_1, w_3) = q_2, \delta(q_2, w_4) = q_3,$$
$$\delta(q_3, w_2) = q_0, \delta(q_3, w_5) = q_4, \delta(q_4, w_6) = q_5,$$
$$\delta(q_4, w_7) = q_5 \tag{4}$$

Because it is a definite FSM, every input will lead to one state only, and the auto-machine can transfer the current state into other one. When the robot is under some state, a character is input, if it is conformed by state transition function in formula (5), then the robot will go to the next state. Otherwise, the robot will still in the current state and execute the action corresponding to the state.

According to the state, alphabet, state transition functions, initial state and final state, a definite FSM A can be constructed.

Where $A = \begin{cases} \{q_0, q_1, q_2, q_3, q_4, q_5\}, \\ \{w_1, w_2, \ldots w_7\}, \\ \begin{cases} \delta(q_0, w_1) = q_1, \\ \delta(q_0, w_3) = q_2, \\ L, \\ \delta(q_4, w_7) = q_5 \end{cases}, \\ q_0, \\ q_5 \end{cases}$, described in Fig.1.
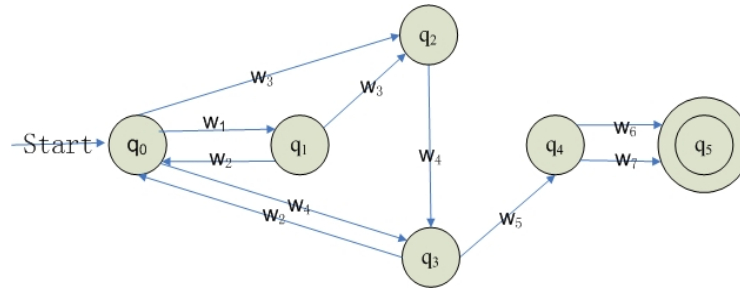

Figure 1. FSM diagram of shooting

Every state of FSM includes in a series of decisions and actions corresponding to current state. When the transition of state begins, switching between the decision and actions will happen, which means that it can achieve the motion planning for the corresponding task. For instance, the sentence accepted by FSM is $\omega = w_1 w_3 w_4 w_5 w_6$, and the switching for robot is $q_0 \rightarrow q_1 \rightarrow q_2 \rightarrow q_3 \rightarrow q_4 \rightarrow q_5$, where the action sequence for robot to execute is finding ball $\rightarrow$ approaching ball $\rightarrow$ adjusting the shoot direction $\rightarrow$ adjusting the foot $\rightarrow$ shooting $\rightarrow$ goal. In the process of state transition, task and relative motion planning are completed by the robot.

4. SIMULATION AND EXPERIMENT RESULTS

In the experiments, images of ball and goal as described in Fig.2 are gathered under different lights. The software of simulation is Matlab, and the results of simulation are described in Fig.3. Fig.4 shows the effect of simulation, from which the variation of H and the amplitude are larger because of the recognition with insufficient light.

As shown in Tab.1, it is linear between adjacent rows. If the number of pixels resulted from clustering is between the

two rows, corresponding ratio of lower threshold can be calculated by linear relation. If the number of pixels is bigger than 10240, the ratio is defined as 0.3. The interference region can be wiped off to obtain the correct recognition results by this algorithm.

Table1. The lower bound table of filter algorithm

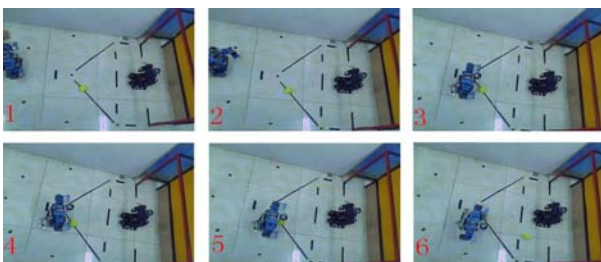| Numbers of pixel for higher threshold	λ (Ratio of lower threshold)
20	0.15
40	0.25
80	0.45
160	0.5
320	0.5
640	0.45
1280	0.4
2560	0.4
5120	0.35
10240	0.3

Figure 2. Ball and goal under different lightings
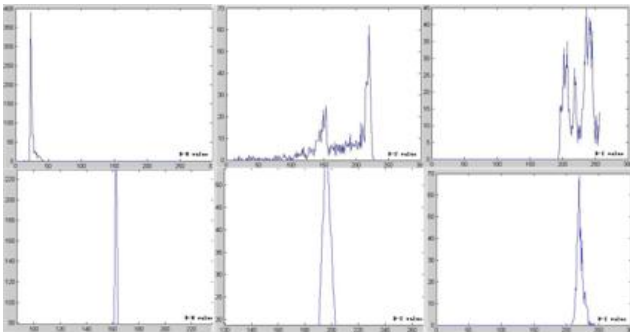
Figure 3. Simulation of ball and goal under different lightings

Figure 4. Effect of recognition for ball and goal
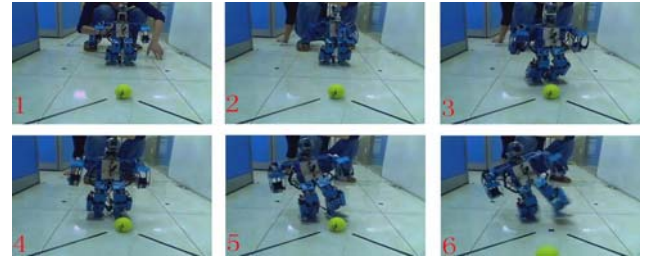
Figure 5. Screenshots of penalty kick for HIT-2 humanoid robot

Screenshots of penalty kick for HIT-2 humanoid robot are shown in Fig.5, including finding ball, approaching ball, adjusting shooting direction, adjusting the foot and shooting.

5. CONCLUSION

According to the research on vision system for humanoid robot, an improved algorithm for image recognition is present by HSI color space. Motion planning on penalty kick for robot is proposed based on definite FSM. Experiments on HIT-2 humanoid robot are completed.

REFERENCES

1. J. Bruce, T. Balch and A. M. Manuel, Fast and Inexpensive Color Image Segmentation for Interactive Robots, *International Conference on Robots and System*, Takamatsu, pp. 2061–2066, 2000.
2. B. C. Tan, Y. X. Mou and Z. Y. Chen, Research On Image Segmentation of Vision System for Autonomous Mobile Robot, *Journal of Xi'an University of Technology*, Vol. 27, pp. 471-473, 2007.
3. Z. Y. Xia, K. Chen and J. Xiong, Research Process on Motion Planning for Humanoid Robot, *Journal of High Technology Communication*, Vol. 17, pp. 1092-1099, 2007.
4. M. Philipp, J. Chestnutt and J. Chuffer, Vision-guided Humanoid Footstep Planning for Dynamic Environments, *Proceedings of IEEE/ RAS International Conference on Humanoid Robots*, pp. 13-18, 2005.
5. B. S. Manjunath, J. R. Ohm and V. V. Vasudevan, Color and Texture Descriptors, *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 6, pp. 703-715, 2001.
6. G. Stockman and L. Shapiro, Computer Vision, Prentice-Hall, 2001.