Vision-Based Human-Vehicle Interaction and Skill Training

Xiaoming Yin, Jiangzhou Lu and **Ming Xie**

School of Mechanical and Production Engineering
Nanyang Technological University,
Nanyang Avenue, Singapore 639798
p145567279@ntu.edu.sg, mjzlu@ntu.edu.sg, mmxie@ntu.edu.sg

Abstract

This paper presents a vision-based human-vehicle interaction system implemented on an intelligent vehicle. The system separates hand images from complex backgrounds using RCE neural network based color segmentation approach and recognizes hand gestures by analyzing the topological features of the segmented hand. Hand postures are used for gesture-based human-vehicle interaction and skill training. In the parking scenario, the combination of gesture guidance and automatic parking makes intelligent vehicle more user-friendly.

1 Introduction

Many driver assistance systems or autonomous driving systems for different aims have been developed and demonstrated. One of the most important issues for these systems is the vision-based reactive motion control. In the area of vision-based intelligent vehicle, although a lot of algorithms have been proposed, there are still a lot of problems and limitations. Most of the intelligent systems just consider structural environment and ignore the interaction between human and intelligent vehicle. In practice, intelligent vehicles often face unstructural environment and lack sufficient sensor messages. They are difficult to complete complex tasks without the skill training and help of human being. In order to make intelligent vehicle to undertake more complex tasks, we propose a vision-based human-vehicle interaction and skill training system in this paper.

Although a few researchers directly study on this issue, a lot of related schemes have been proposed for vision-based intelligent vehicle, including lane following control, obstacle detection and automatic parking. The most popular application for vehicle vision is road following or lane following. A vision system detects the road edges and markings, and measures the curvature of the lane for vehicle speed and lateral control. Several detection algorithms have been developed for a personal vehicle system [1] and automated highway vehicle systems [2] [3]. The earliest vision system for obstacle detection employed for an intelligent vehicle was developed in Japan. The principle of obstacle detection is based on parallax [4]. Different parking algorithms have been developed in INRIA [5] and NTU [6].

For skill-based problem, A. Wolfgang has proposed a skill-based visual parking control using neural and fuzzy network [7]. Their approach is based on acquisition and transfer of an experienced human driver's skills to an automatic parking controller. The controller processes visual information from a video sensor and generates the corresponding steering commands by the use of neural networks.

Figure 1: The smart car in our lab

Fig. 1 shows the smart car in our lab. It is equipped with a vision system including two stereo cameras and a processing unit, which can detect the environment within its field of view in a range from 5m to 20m ahead of the vehicle in real time. The vision-based intelligent system is composed of several separated but related modules, such as GUI module, vision module, motion controller module, communication module, etc. In this paper, we discuss the problem of vision-based human-vehicle interaction and focus on gesture-based human-vehicle interaction and parking skill training.

The rest of the paper is organized as follows. The problems of hand image segmentation and hand posture recognition are discussed in the next section. Section 3 describes the process of gesture-based human-vehicle interaction. Section 4 demonstrates the application of gestures in the parking scenario. Finally, conclusions are given in Section 5.

2 Vision-Based Hand Gesture Recognition

Gesture based interaction was firstly proposed by Krueger as an innovation of human-computer interaction in the middle of the seventies. There has been a growing interest in it recently, because hand gesture is a natural and intuitive form of both interaction and communication for human, and can be used to convey ideas more efficiently than speech in noisy environment.

2.1 Hand Image Segmentation

Hand image segmentation separates hand images from backgrounds. It is the first important step in every hand gesture recognition system. We have developed a novel color segmentation approach based on Restricted Coulomb Energy (RCE) neural network for hand image segmentation in [8].

Color segmentation techniques rely on not only the segmentation algorithms, but also the color spaces used. After exploring RGB, HSI and L*a*b* color space respectively, we found L*a*b* color space was the most suitable for hand image segmentation. L*a*b* color space is the uniform color space defined by the CIE (Commission International de l'Eclairage). It maps equal Euclidean distance in the color space to equal perceived color difference. The color images captured by cameras can be converted into L*a*b* color space from RGB color space.

A common belief is that different people have different skin colors, but some studies show that such a difference lies largely in intensity rather than in color. We quantitatively investigated the skin color distribution of different human hands under different lighting conditions. It is found that skin colors cluster in a small region in L*a*b* color space and have a translation along the lightness axis with the change of the lighting conditions.

Skin colors cluster in a specific small region in the color space, but the shape of skin color distribution region is complicated and irregular. Common color segmentation techniques based on histogram are not effective enough to segment hand images from the complex and dynamic background. In this paper, the RCE neural network based color segmentation method is used. RCE neural network was designed as a general purpose, adaptive pattern classification engine. It consists of three layers of neuron cells. Three cells on the input layer of the network are designed to represent the L*a*b* color values of a pixel in the image. The middle layer cells are called prototype cells, and each cell contains color information about an example of the skin color class that occurred in the training data. The cell on the output layer corresponds to the skin color class.

During training procedure, the RCE network allocates the positions of prototype cells and modifies the sizes of their corresponding spherical influence fields, so as to cover the arbitrarily complex distribution region of skin colors in the color space. During running, REC responds to input color signals in the fast response mode. If an input color signal falls into the distribution region of skin colors, this input color signal belongs to the skin color class, and the pixel represented by this color signal is identified as skin texture in the image.

The RCE network identifies all the skin-tone pixels in the image during running procedure. There are occasions that other skin-tone objects such as faces are segmented, or some non-skin pixels are falsely detected due to the effects of lighting conditions. We assume the hand is the largest skin-tone object in the image, and use the technique of grouping by connectivity of primitive pixels to further identify the region of the hand. With numerous skin color prototype cells together with their different spherical influence fields, the RCE network is capable to segment various hand images under variable lighting conditions from the complex background after being trained properly.

2.2 Hand Posture Recognition

Based on this hand image segmentation algorithm, we have developed a new method for accurate recognition of 2D hand postures in [9]. In this method, topological features of the hand, such as the number and positions of the extended fingers, are extracted from the binary image of the segmented hand region, and hand postures are recognized on the basis of the analysis of these features.

To find the number and positions of the extended fingers, the edge points of the fingers are the most useful features. We use the following proposed algorithm to extract the edge points of the fingers.

1. Calculate the center of mass of the hand from the binary image of the segmented hand region, in that pixel value 0 represents the background and 1 represents the hand image;
2. Draw the search circle with the radius r at the position of the center of mass;
3. Find all the points $\mathbf{E} = \{P_i, i = 0, 1, 2, ..., n\}$ that have the transition either from pixel value 0 to 1, or 1 to 0 along the circle;
4. Delete P_i and P_{i-1}, if the distance between two conjoint points $D = |P_iP_{i-1}| <$ threshold λ_d;
5. Increment the radius r and iterate Step 2 to 4, until $r > \frac{1}{2}$(the width of the hand region).

The purpose of Step 4 is to remove the falsely detected edge points resulted from imperfect segmentation. Then the extracted feature points accurately characterize the edge points of branches of the hand, including the extended fingers and the elbow. Fig. 2(a) shows the segmented hand image. Fig. 2(b) shows the part of Fig. 2(a) with the scale of 200%, in that the green circles represent the search circles and the red points represent the extracted feature points.

For each branch, two edge points can be found on the search circle, so half of the feature points found on the search circle just indicate the branch number of the hand posture. However, the feature points on the different search circles are varied. In this paper, we select the biggest one among the numbers whose occurrences are bigger than a threshold λ_n as the branch number BN. After the branch number BN is determined, we selected the middle one of the search circles on that there are BN branches to obtain the branch phase. The branch phase is the positions of the detected branches on the search circle, described by angle. Fig. 2(c) shows the relationship between the branch number and the radius of the search circle, and Fig. 2(d) shows the radius of the selected search circle as well as the branch phase on this circle.

After the branch phase is determined, the width of each branch BW_i can be obtained easily from the branch phase. The widest branch must be the elbow, and is used as the base branch B_0. Then the distance from other branch B_i to B_0 can be calculated, that is just the distance between the finger and the elbow BD_i. Using these parameters mentioned above: the branch number BN, the width of the branch BW_i, the distance between the finger and the elbow BD_i, the hand posture can be recognized accurately.

3 Gesture-Based Human-Vehicle Interaction

The parameters that are used for recognition are all very simple and easy to estimate in real time, but they are distinctive enough to differentiate those hand postures defined explicitly. The recognition algorithm also possesses the property of rotational invariance and user independence because the topological features of human hands are quite similar and stable. So our posture recognition algorithm is suitable for all kinds of human-machine interaction, such as human-robot interaction, human-vehicle interaction. In [9], we uses postures for human-robot interaction and gesture-based robot programming. In this paper, we discuss the application of gestures on human-vehicle interaction.

In our human-vehicle interaction system, live images with the size of 384×288 are captured through two CCD video cameras (EVID31, SONY) that are installed in the front of the smart car. At the end of each video field the system processes the pair of images, and output the detected hand information. The processing is divided into two phases: hand tracking phase and posture recognition phase. At the beginning, we have to segment the whole image to locate the hand, because we have not any information on the position of the hand. After the initial search, we do not need to segment the whole image, but a smaller region surrounding the hand, since we can assume continuity of the position of the hand during tracking.

At the tracking phase, the hand is segmented from a low resolution sampling of a small region surrounding the hand that has been detected in the previous frame. The system also detects the motion features of the hand such as pauses during the tracking phase. Once a pause is confirmed, the system stops the tracking, crops a high resolution image tightly around the hand and performs a more accurate segmentation based on the same techniques. Then the topological features of the hand is extracted from the segmented hand image and the hand posture is classified based on the

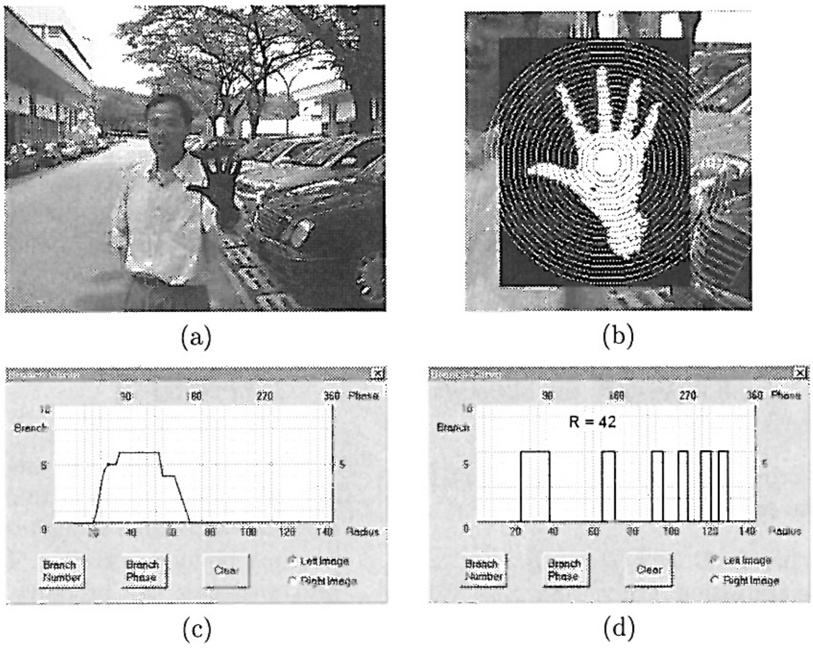


Figure 2: (a) segmented hand image, (b) Feature points extracted from the binary image of the segmented hand region, (c) Plot of branch number of the hand vs the radius of the search circle, (d) Plot of branch phase of the hand on the selected search circle.

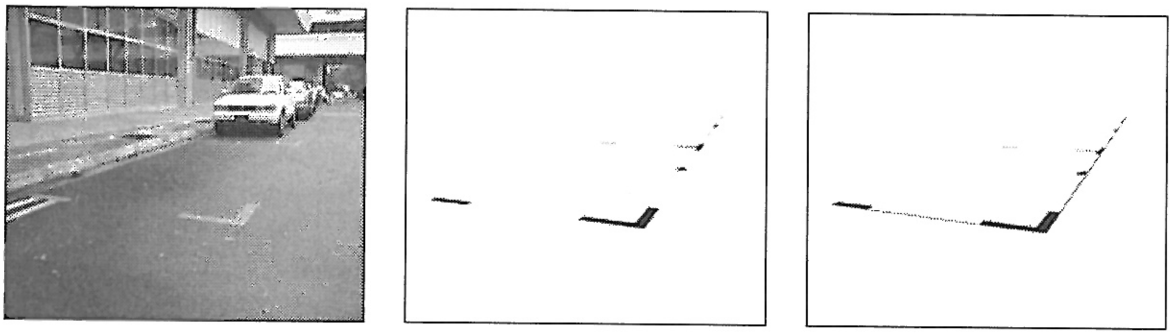
analysis of these features as described in Section 2. If the segmented hand image is recognized correctly as one of the defined postures, the vehicle will generate motion which is associated with this posture. If the segmented image can not be recognized because of the present of noise, the vehicle will not output any response. After the posture recognition phase is finished, the system continues to track the hand until another pause is detected. In our implementation, the hand can be tracked at the speed of 4-6HZ on a normal PC (450MHz CPU). The time spent on the segmentation of the high resolution image is less than 0.5 second, and the whole recognition phase can be accomplished within 1 second.

4 Gesture-Guided Parking Training

4.1 Vision-Based Automatic Parking

Parking is a typical scenario of automatic control for intelligent vehicle. Numerous methods have been proposed for automatic parking. In [6], an algorithm of vision-based automatic parking was developed for the smart car. In this algorithm, the RCE-based color segmentation approach is used to detect the marks of the empty slot in the image. Then the outline of the parking slot is estimated from the detected marks as the geometric features. For example in Fig 3, (a) shows a real image of a car park, (b) shows the detected parking marks and (c) shows the estimated outline of the parking slot. After the geometric parameters of the parking slot have been extracted from the image, they are transformed from the image coordinate system to the 2D coordinate system that is attached to the ground plane using the method of 2D vision, then the position and orientation of the empty parking slot in the real word is known.

After a local map has been built based on the information of the empty slot, the motions that bring the car from the current configuration (position and orientation) to the goal configuration can be planned. The motion planing problem involves two steps. The first is the path planning and the second is the trajectory planning. As to path planning, a new concept of parking strategy is proposed. We divide the whole motion into several steps (forward and backward), instead of planing the whole procedure as one step. We have developed a two-step algorithm and adopted quintic polynomial curves to plan a smooth path. After the path is obtained, the trajectory of the car along that path can be designed easily. Since a car-like vehicle must move in its longitude direction, the only possible orientation on the path is the tangential direction



(a) (b) (c)

Figure 3: (a) A real image of a car park; (b) Segmented parking marks; (c) Estimated outline of the parking slot.

at that point. Thus the orientation of the car along the planned path is fully determined by the following function:

$$\theta = arctan(f'(x)) \tag{1}$$

where θ denotes the orientation of the car, and $f'(x)$ is the path function. Once the path and trajectory are determined, motion commands (velocity and steering angle) that make the car to follow the designed path can be generated.

4.2 Gesture-Guided Parking Training

Automatic parking can be implemented only if the empty parking slot is detected, or the goal configuration of the car is determined. However, there are occasions that the marks of the parking slot can not be seen until the car comes close to the empty slot. For example, when a car enters a large park with lot of cars, an empty parking slot can not be found at once even by a human driver. In such case, human's guidance is needed. Hand gestures have been used to help a human driver find an empty parking slot in conventional car parks. For intelligent vehicles without human drivers, hand gestures are also a promising way to interact with and guide them.

The four postures shown in Fig. 4 all have distinctive topologic features to be recognized. We use them to guide the smart car to approach the empty parking slot and park in it. The detection and recognition of these postures are based on the techniques described in Section 2 and 3. In Fig. 4, Gesture (a) is used to guide the car to turn right, (b) to turn left, (c) to forward and (d) to backward. We use the any gesture other than these four gestures to stop the car, in order to avoid any danger caused by misoperation in the case that a hand gesture is not detected or recognized correctly.

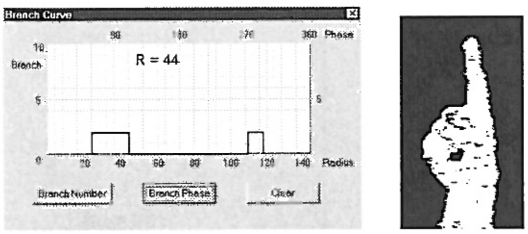
During the procedure of moving to the parking slot, the smart generates motion series according to the gestures presented by the human guide. Synchronously, all the motions except stop occurring on the way are also saved in the computer system of the smart car. Then next time, if the car will park at the same slot, it can go to the slot automatically according to the motion commands generated during guidance. If the slot is changed, the smart car can be trained again by new guidance of human.

4.3 Combine Gesture Guidance and Automatic Parking

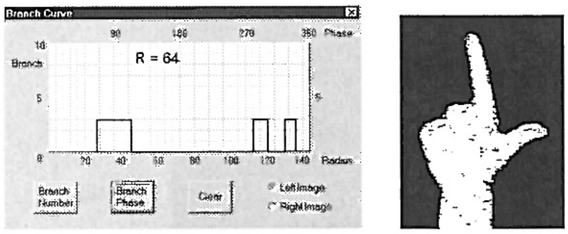
Gesture guidance and automatic parking can be combined to make intelligent vehicle more feasible and flexible. When the car just enter the car park, and an empty parking slot can not been seen, hand gestures are used to guide the car close to an empty slot. After the empty slot has entered the view field of the cameras of the car, the marks of the slot are detected, and automatic parking is implemented.

5 Conclusions

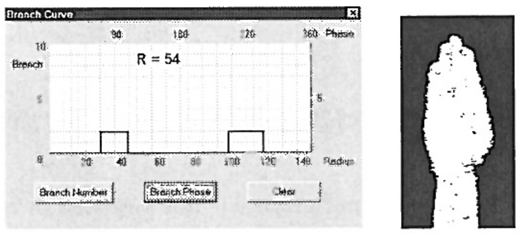
Human-vehicle interaction is an important issue for intelligent vehicle. This paper focuses on the discussion of vision-based hand gesture recognition and gesture-based human-vehicle interaction. In the system, human hand images are captured by the onboard cameras and segmented with the color segmentation approach based on the RCE neural network. The topological features of the hand are extracted from the



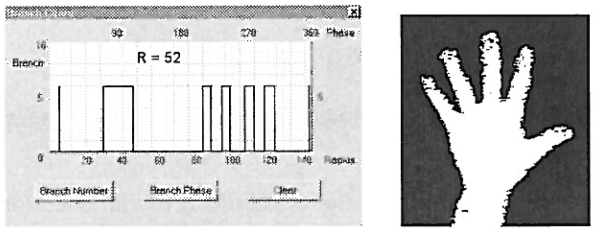
(a) Turn right



(b) Turn left



(c) Forward



(d) Backward

Figure 4: Hand postures used for human-vehicle interaction

binary image of the segmented hand region, and the hand posture is recognized on the basis of the analysis of those features. In the parking scenario, four hand postures have been used to guide the smart car to find an empty parking slot. And combination of gesture guidance and automatic parking makes intelligent vehicle more feasible and flexible.

References

[1] S. Tsugawa, "Vision-based vehicle in japan: the machine vision system and driving control system," pp. 278–285, 1993.

[2] C. A. Richards, C. E. Smith, and N. P. Papanikolopoulos, "Vision-based intelligent control of transportation systems," in *Proceedings of IEEE International Symposium on Intelligent Control*, pp. 519 - 524, 1995.

[3] B. Ma, S. Lakahmanan, and A. Hero, "Road and lane edge detection with multisensor fusion methods," in *Proceedings of International Conference on Image Processing*, pp. 686 - 690, 1999.

[4] A. Hattoti, A. Hosaka, and M. Taniguchi, "Driving control system for an autonomous vehicle using multiple observed point information," in *Proceedings of IEEE Intelligent Vehicle Symposium*, pp. 207–212, 1992.

[5] I. E. Paromtchik and C. Laugier, "Autonomous parallel parking of a nonholonomic vehicle," in *Proceedings of IEEE intelligent Vehicle Symposium*, (Tokyo, Japan), pp. 13–18, 1996.

[6] J. Xu, G. Chen, and M. Xie, "Vision-guided automatic parking for smart car," in *Proceedings of IEEE intelligent Vehicle Symposium*, (Dearborn, USA), pp. p725–730, 2000.

[7] W. A. Daxwanger and G. K. Schmidt, "Skill-based parking control using neural and fuzzy networks," in *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, pp. 1659 -1664, 1995.

[8] X. Yin, D. Guo, and M. Xie, "Hand image segmentation using color and rce neural network," *International journal of Robotics and Autonomous System*, vol. 34, pp. 235–250, March 2001.

[9] X. M. Yin and M. Xie, "Finger identification in hand gesture based human robot interaction," in *Proceedings of the IEEE/RAS International Conference on Humanoid Robots*, (MIT, Boston, USA), September 2000.