Embedded System Design Project Proposal

Auto-Parking System Implemented with FPGA Car

Ruoqing Fu (rf2475), Po-Yen Chou (pc2621), Hao-Yu Hsieh (hh2539)

Our group proposes to build a FPGA based smart car that can detect empty parking lots and automatically pull in. We have thought of two possible ways to implement it for different scenarios as described below:

1. Use ultrasonic wave or radar to detect distance

In this method, the car is initially set to go straight forward on a road. Meanwhile, by using either ultrasonic wave or radar, the car will start to look for the potential parking space on its left or right side. When the car detects an available parking spot, it will enable the internal auto parking mechanism and keep adjusting its position based on the feedback from ultrasound/radar system until it is perfectly parked. In this implementation, we assume that all other cars nearby are parked perfectly.

2. Use camera with the help of black tape on white floor

In this scenario, we use black tape to represent a parking spot for camera to recognize. Again as described in the first method, the car initially moves straight forward trying to find an empty parking spot. But it will use a camera instead to detect the parking space, and start the parking mechanism correspondingly. The car makes necessary adjustments by what it can "see" from the camera until it's completely in. To simplify the problem, we assume that no other cars are present between the departing location and the slot.

A big advantage in method 1 is that we do not need to prepare a slot for the car. It can park itself as long as there is enough space. However, it is difficult to determine if there is an empty spot nearby. Because this method is highly dependent on the distance between the car and the surrounding objects, the car has to move with respect to other permanent obstacles/objects on the road. If the car tilts or moves out of the way, it may cause inaccurate parking space detections. To simplify the problem, we may assume that the car moves straight forward with other cars, which are parked perfectly at the beginning. When it detects a big distance difference, it means there could be an empty parking space (and we will use other algorithm to make sure the space is large enough, such as having the car move way farther in order to make an accurate conclusion that the recently predicted parking space is real). These assumptions limit the practical application of this system.

However, in method 2 ,the car can initially move in any direction on any route as long as the slot will eventually appear in the camera. Nevertheless, the downsides are also obvious - we need to prepare a slot for it, and the vision recognizing algorithm will be extremely complex if other objects are involved.

The two methods described above are just a general guide of how we plan to build the smart parking system conceptually. At this point, we are still not sure which method is better and could possibly be implemented given all the resources we have. We are hoping to have an discussion with professor and TAs on this matter and seek more advice from your perspective. There is no doubt that more details need to be understood and illustrated in terms of how to use the proposed techniques to detect parking space or develop an appropriate algorithm to continuously adjust its positions . We will provide more info as the project proceeds along. In addition, we understand that the FPGA boards in lab aren't set up to be mobile, so for the vehicle part, we will find one for our own use.