High Speed Object Tracking

Introduction

As systems of automation, specifically in the world of robotics, become faster, there is an increasing need to track objects at higher speeds than ever before. From autonomous cars watching the road to missiles and planes guided by high speed computer vision analysis, the quicker computer vision processing becomes, the more time there is to make artificial intelligence decisions based on that analysis. However, not all designers choose the same solution to the tracking problem. Some rely strictly on visual analysis while others combine other sensory inputs such as ultrasonics or infrared trip wires. This paper analyzes a mix of hardware and software controls used in robotic systems to track objects at high velocities and looks into their applicability in high speed game playing robots.

Historical Implementations

Perhaps one of the first game playing robots to make use of a tracking system was a designed by AT&T Bell Laboratories to play table tennis against a human opponent. With speeds upwards of 25 miles per hour, tracking a ping-pong ball is a complex task. The engineers decided to use a completely visual system to track the ball, utilizing a black room such that the white ball would produce a high contrast, ensuring easy identification. The difficulty with visual processing is that, depending on the complexity of the analysis, the time it takes to track the object may become too slow to control the robotic movements [1].

Another solution to fast object tracking, as implemented by a group of University of Illinois engineering students to track a foosball ball, is to use pairs of LEDs and sensors such that when the beam is interrupted, the ball can be located. By using arranging these into a matrix of cross-beams, the position of the ball may be determined and reacted to. By using this simple method, the speed of processing was increased to a point that a microcontroller could handle all of the tracking. However, in this scheme, the students ran into an issue with the accuracy of tracking because the infrared signal introduced some small, but significant error [2].

Technology Details

When objects are tracked visually, there are some critical decisions of the equipment and methods used to acquire and process each frame. Tracking a very fast object requires that every piece of hardware and software must be designed in a way to reduce latency at each step: image
acquisition, data transport, processing, and motion prediction.

The best way to decrease the time required for image acquisition is to decrease the exposure time. In order to still have the same source image, however, there must be a brighter light to compensate for the lower light absorption time [3]. This is because “a CCD camera's electrical output is proportional to the integral of the number of photons received during the sampling interval” [2].

Typically the step with highest latency is the image processing. However, because of this, it also has the most room for improvement. One way to decrease the image processing time is to create a smaller window on which to operate the image processing functions on. This is because the time required for processing is “proportional to the number of pixels within the window” [4]. Another way to speed up image processing is to focus on a single color. If it is known that the tracked object is the only object in the field of view of a specific color, it becomes much easier to track it in near-real-time environments [5].

Once the image processing is completed, predicting the path of the object to reduce the next frame's window size becomes important. There are different ways to do this. One is to use a Kalman filter, which uses a recursive function based on previous locations [3]. However, this can get computationally expensive for fast tracking, and is sometimes replaced by a simpler linear or quadratic function, as was the case with the ping-pong playing robot [1].

**Conclusion**

Given the vast number of different types of sensors for tracking, the hardware and software combination is often a unique solution that best fits the application. Because of this, there is no “one size fits all” solution for all fast tracking applications. However, there are some pieces of hardware which combine eliminate some of the steps from the process, like the CMUcam, which combines vision acquisition and processing [6]. In more rapid gameplay situations, some other methods may be necessary, using a combination of sensors and predictive algorithms for the fastest possible data processing. This would allow the rest of the robotic system the maximum possible time for analysis and movement.
References


