Robot Soccer: A BYU Senior Design Project

Alex Wilson, Benjamin Lewis

Background

This senior student design competition started in 2001 with the aim of creating an autonomous team of small robotic soccer players. These robots were to rely on an overhead camera and a remote computer to analyze the playing field and issue commands the robots. The project was intended to provide a realistic engineering problem that demanded teamwork and wide variety of skills and expertise. The project eventually grew to be a fast-paced, 2-on-2 spectacle.

Several years passed and the project was revived again in early 2014 with the additional goal of moving the all of the computation on to the robots themselves. This simple switch meant many new and exciting challenges for the students to solve, such as determining position using visual localization, multi-robot collaboration, and efficient embedded vision processing.

One of the 5 teams that competed last year (2014) and their team logo below.

CRITICAL PATH

Robot Soccer playing field with markers identifying the corners and midpoint of the field.

Motivation

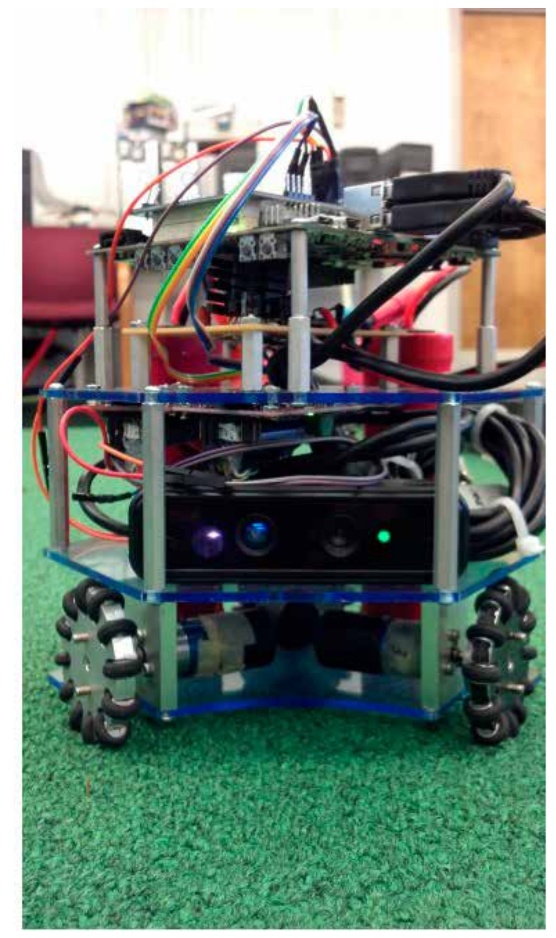
In order to have effective, fully autonomous robots, it is necessary to have a computer board that is capable of performing vision processing and other computational tasks fast enough to control the robot smoothly. The MinnowBoards used for the project in 2014 were critically unable to process more than a few frames per second of video, making them unable to perform effectively.

This year, the robot soccer senior project has temporarily reverted to using an overhead camera and off board processing to provide computer vision input for the robots. This is an important step in rebooting the project because it simplifies the vision task and allows other aspects of the project to be more fully developed. The project in 2014 suffered in part by being too broad in scope to effectively solve all the challenges. By focusing on solving a smaller subset of problems this year, such as motion control and strategy, future projects can focus more on more challenging aspects of the project, such as onboard vision processing and visual localization.

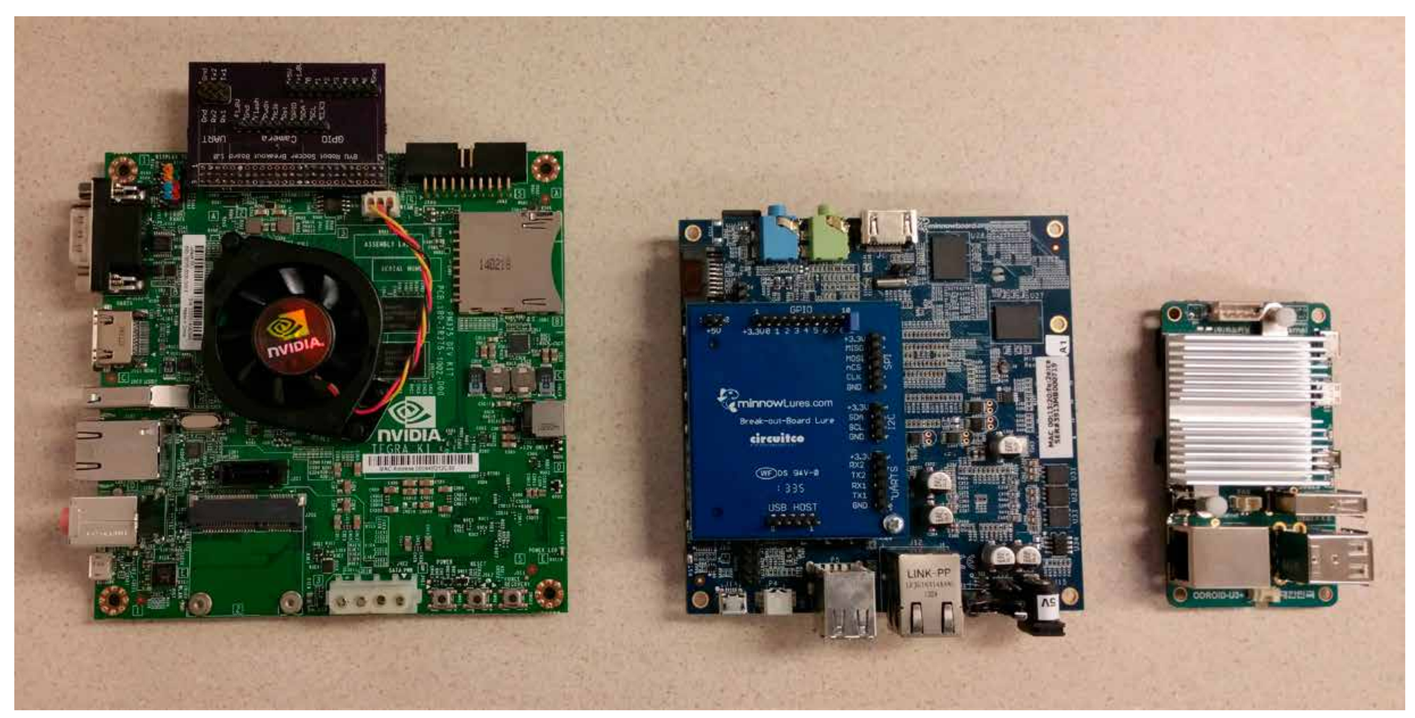
The other critical aspect of using an onboard computer for all processing is having a computer that is capable of performing the computational tasks. Image processing is the most intense task required by the project, and as such, a board needs to be identified that would be capable. Knowing that the Jetson board has the CUDA cores that can parallelize image processing, we decided to benchmark it on vision tasks that we had designed for the 2014 project.

Contest Entry

Shortly after the conclusion of the senior project in April 2014, the robots and playing field were disassembled so they could be stored for the next year. Since our vision code was developed to find these unavailable objects, a different vision task was needed to test the vision processing capabilities of the Jetson as part of the whole system. We needed to pair this task with the other functions of the system - control, tracking, and movement. In the absence of other robots and the markers on the field, we decided that a comparable vision task would be to locate the nearest face and center it horizontally in the field of vision. Using the Jetson, we were able to obtain a high enough frame rate to effectively provide the location of the nearest face to a real-time motion control algorithm. This algorithm then rotated and moved the robot laterally to center the face.



Robot Soccer player equipped with MinnowBoard, camera, batteries, and motors.



(left to right) Jetson TK1, MinnowBoard, and ODroid U3

Methodology

In order to determine the effectiveness of the Jetson board in a robot soccer application, four computing platforms were benchmarked. The CPU and the CUDA cores on the Jetson board were directly compared to the ODroid-U3 and the MinnowBoard development boards. These platforms have the small form factor that can easily be incorporated into a robot that performs all processing onboard.

As a basis for comparison, we also benchmarked a desktop i5 processor. This is the platform that is being used for this year's robot soccer project to process a video stream from an overhead camera. The desktop computer is an ideal control for the comparison as it is able to comfortably handle all 30 frames per second provided by the camera.

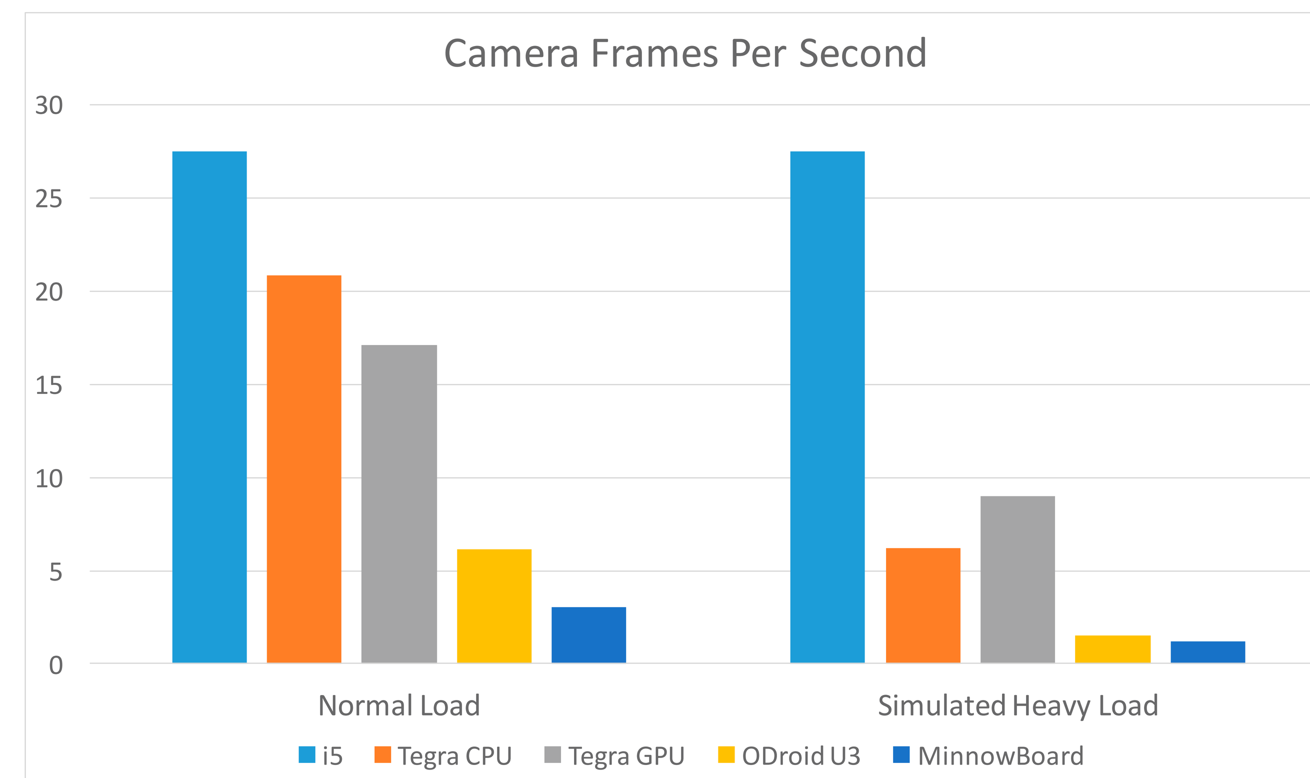
Each machine was tested under two conditions. In order to show how each processor performs under the current robot soccer rules, the vision code that was developed during the project was run. Given that we hope to grow the project into a more complex task in future years, we also tested a heavier vision processing load. This heavier load was created by adding additional basic image manipulations to the code used to test the standard load.

Critically, the CUDA cores on the Tegra were better able to handle the load of the heavier vision task than were the other small computing platforms. In future iterations, this slowdown may be mitigated by a better understanding of how to use the CUDA cores, allowing more image processing to be done in parallel.

Performance Benchmarking

Of the three embedded platforms tested the Jetson TK1 had by far the best performance. Both the MinnowBoard and the ODroid U3 were both fine at other simpler tasks such as controlling the motors and estimating the state of the game, but their ability to process images fast enough severely limited our filter's accuracy. With the ability to perform vision processing at more than 20 FPS the Jetson really shines and would pair well with our estimator and filters.

The normal load show below is an example of the FPS we encountered when running our basic vision processing code. On the Tegra we simply replaced the normal OpenCV functions with their respective OpenCV/CUDA cousins. The simulated heavy load demonstrates what future and more intense vision processing might require of each system. The ODroid U3 and MinnowBoard take a hit on this load while the Tegra's 192 CUDA cores continue to chug along.



Hardware Comparison

An embedded robotics system requires more than just raw processing power. It also needs to take into account size, power, IO support, and general community support. The first board we tried was the MinnowBoard which boasted a small form factor and little power consumption, but needed a breakout board for many of the GPIO pins and UARTs. The Minnow also lacked compatibility with most Linux distributions, due to 32-bit UEFI, which meant a small support community.

The second board was the ODroid U3 which was much smaller than the MinnowBoard, but packed a punch in performance and its long history of community support. Due to the nature of its small size many features such as the serial debug port require additional attachments.

The Jetson TK1 boasts low power, countless IO pins, a well supported Linux distribution, and a very responsive forum. Although its form factor is somewhat larger than desired, it was still able to fit into the chassis of our robot. Developing software on the Jetson was also considerably much easier than the other two boards. What took months of development on the previous two boards took a

Conclusions

Due to difficulties with the MinnowBoard and problems with the onboard cameras we've temporarily switched to using an overhead camera and are testing the ODroid U3s as our embedded platform. The ODroid U3 has shown that it can process frames almost twice as fast as the MinnowBoard, but our tests simulating a heavier load predict that it may not scale well to more intense applications and algorithms.

The Jetson TK1 proved overall to be an amazing development platform. It easily outperformed our previous efforts and at times it was difficult to decide whether it was a desktop or an embedded computer. On all fronts it demonstrated and fulfilled its design as a powerful tool for embedded vision processing.