Proposal for the R. Links Brain

Executive Summary

Digital products in an open source format are becoming more of a driving force in the evolution of software; specifically in robotics. The proposal is to develop an open source library that will act as a framework software for a quadruped robot called R. Links. This C++ library, or "brain", can be easily applied to a wide variety of both public and private projects. The R. Links Brain will greatly reduce the knowledge gap between an average user and an advanced user by abstracting the most common code shared among similar robotics projects. The common code that makes up the R. Links Brain includes – but is not limited to – sensor integration, kinematics algorithms, resource contention prevention, and scalability considerations for multi-core systems. With an easy-to-use API and robust documentation, the R. Links Brain can solve many of the problems faced with robotics aficionados today.

Project overview

Development and research teams from all over the world are starting to set their sights on new ways to incorporate interactive hardware into our every-day lives. This can be seen with Near Field Communications (NFC), GPS tracking for our taxi drivers, and even the Roomba robotic vacuum. But many times these devices are not so easily made and often their initial development has no clear starting point. Do you begin with the Bluetooth connection or the in-app controller to open a communication with the hardware of the device? This is an example of a potential road block facing many of these entrepreneurs and engineers. Moreover, most robotics-based projects would require a deep understanding of kinematics, or motion of an object in mechanics, along with a strong software development background in order to programmatically coordinate the moving parts in any robot. Learning to manipulate the hardware and consistently designing efficient software takes a large amount of time. Many developers may even have to reuse or redevelop nearly identical software across similar projects. But what if there were a starting point made for you that was both easy to use and easy to understand?

We are proposing an open source library as a solution to all of these problems. The R. Links Brain will abstract all of the software required to control our robotic platform allowing the user to focus on their own design or research without having to waste time developing the non-trivial software framework of their project. Advanced users will be able to adapt the code to their own needs.

Sam Zapolsky at the George Washington University has created a physical frame and a base-level set of instructions for the R. Links quadruped. A motion capture device and Inertial Measurement Unit (IMU) has been purchased and will provide the eyes and ears to the quadruped device. All that's need is the R. Links Brain library to pull all of these pieces together in a usable interface that can be used on projects for years to come.

Business Plan

It's become a standard that for a quality robot the price can range anywhere from around $20,000 to $200,000. Quality here can be defined as a measure of reliability, longevity, and efficiency. In other words, it does what you want it to for as long as you need it to at the lowest resource cost possible. Across this spectrum exist robotic platforms that are only manipulated through proprietary software. This software may even cost an extra fee to customize so that it does exactly what you need it to. At a price of less than $5000, the R. Links and the R. Links Brain open source package will become the premier robotics solution in the areas of affordability, customizability, and usability. A full project plan has been developed for the final step of creating the R. Links Brain proposed in this document.

The development of the R. Links Brain will last over the course of a year. During this time, a small development team of a project manager, a robotics and positronics specialist, and a developer are working together to test, prototype, and implement the library. After assembling the team, milestones were set for the project including 30%, 40%, and 70% status checks. The project manager keeps the project on track and the robotics specialist holds a consultation role. The core software development is almost entirely done by the C++ programmer who works around 20-30 hours per week during the one year development period. Once this timeline is complete, the code-base will be maintained by the students from the GWU positronics lab. The maintenance will include anything from adjusting the code to user requests to completely refactoring sections of library for optimizations or updating the sensors. During this maintenance and growth, the developers will adhere to established coding standards in an effort to uphold the highest level of readability. The specific conventions used in the library are built explicitly for the R. Links target audience.

The R. Links Brain is a tool that will be most useful to those with an existing interest in robotics. The intention is for these users to be inside the academic sphere with a focus around systems, robotics, or both. When the R. Links Brain gets the attention of influential researchers in the academic realm, the library will be improved on, reviewed, and shared among peers. As an open source code library, the R.

Links Brain has the potential to reach each and every member of this populace, but it will not happen automatically. In order to accomplish our goals, we will submit research papers to IEEE and suggest the use of the library to the GWU community exclusively at first. Given that this project is open source and lacking any significant marketing, there are some feasible pitfalls involved with its release.

In the project outline, several scenarios have been identified that could possibly occur. Most, if not all, of which revolve around a poor adoption of our software. If this is the case, there is a negligible level of loss due to the low cost of production and development.

Of the top few scenarios, we will first consider if the R. Links Brain library is never used. If the R. Links project is not accepted in the academic community – for whatever reason – we have still accomplished a major milestone in the evolution of robotics. We can analyze the failure and create a new product based on our findings. But it is probably to consider the library receiving some level of traffic.

In scenario number two, it is possible that the R. Links Brain is only mildly successful, or possibly only implemented in small pieces instead of holistically. Even though we don't reach the ideal state, we will have succeeded in a few of our initial goals. The users will be able to provide feedback, and we can improve in our next iteration. So even this outcome is not as bad as it may be in similar projects. Even a worst-case scenario will still allow for a smooth project recovery.

In what we could consider the worst-case, the project never reaches what we would consider a full level of completion. In this scenario, a development milestone is not reached in time. If this repeatedly occurs and the project never reaches completion, the R. Links Brain project plan states that whatever software is completed will still be released to the public. Since at the time of release we will still have a percentage of our complete product, we will only be subject to the first two scenarios mentioned above.

Despite the risks, we have all the tools we need to provide for a quality product with a high adoption rate, but given the nature of the R. Links Brain its impact relies on the library being used in projects, improved on, and talked about.

Social Impact

In regards to the R. Links Brain's effect on society, it is possible that the R. Links Brain changes how people everywhere perceive robotics. On the other hand, it may only slightly adjust the perspective of a few academics. In both of these scenarios we do see a change. One may have more of an immediate effect than the other, but the result is the same.

Just as the personal computer began as a fairly complicated assembly of disjoint software components, robotics has also seen the same sort of neglect of grace and organization in software. It is common to find expensive components with very little documentation and limited support software. It is assumed that the user already understands the device before they've ever used it. In the world of PC's this disregard for the user was overcome when the software abstracted the low-level and focused on the importance of the user experience. The goal of the R. Links Brain is similar. The library proposed here will open the world of robotics to entirely new groups of less advanced users. By providing all the tools necessary to get started with many different projects, cutting-edge robotics becomes something that is within the reach of the layman rather than only those with a doctorate. Therefore the social impact of this product may be equally widespread as the early computers.

It is probable that robotics will become more and more common as the technology becomes increasingly affordable and accessible. As this trend continues, more and more development will be done in the field. This will lead to more powerful and user-friendly devices. Therefore, the development of the R. Links Brain – or some similar project – is simply inevitable.

Investing in the R. Links Brian is an opportunity to be a part of the future. The future of robotics, the future of embedded systems, and the future of technology in our society.