

## **Aquila: Massively Parallelised Developmental Robotics Framework**

*Martin Peniak, Anthony Morse, Christopher Larcombe,  
Salomon Ramirez-Contla and Angelo Cangelosi*

Centre for Robotics and Neural Systems  
School of Computing and Mathematics  
University of Plymouth

The developmental robotic approach often relies on complex and computationally intensive algorithms running in parallel while controlling real-time systems. The processing requirements are increasing with every added feature and it is not uncommon that at the end of the software development stage a particular system is unable to cope with real-time robot-control tasks.

In 2003, NVidia released a new architecture called CUDA (Compute Unified Device Architecture), which allows programming massively parallel GPU (Graphics Processing Unit) devices, which are in most cases significantly outperforming standard processors on parallel tasks (e.g. Jang et al. (2008), Jung (2009)). Preliminary benchmarking tests of Aquila showed significant speedups, in some cases over 1000x faster.

Aquila is an open-source software inspired by the recent advancements in supercomputing and was developed as a part of the iTalk project aiming for integration of actions and language in humanoid robots. Software that would facilitate such integration and allow developers to share and reuse their work within one unifying framework was required but not present and therefore we started the Aquila project.

Aquila not only aims to provide an easy access to often used many modules, tools and applications but also aims to bring different researchers together to work on Aquila in a collaborative way, which will bring about the high-level integration of multiple subsystems. Aquila is primarily designed for use with the iCub humanoid robot or its simulator (Tikhanoff et al., 2008) but also provides several stand-alone tools that can be used by any one who does not have access to the iCub robot or its simulator such as self-organising maps, multiple timescale recurrent neural networks and training algorithms and others.