VIP: The Video Image Processing Framework Based on the MIRO Middleware

Hans Utz\textsuperscript{1,2}, Gerd Mayer\textsuperscript{1}, Ulrich Kaufmann\textsuperscript{1}, and Gerhard Kraetzschmar\textsuperscript{3,4}

\textsuperscript{1} University of Ulm, Neuroinformatics, Ulm, Germany
\{gerd.mayer,ulrich.kaufmann\}@uni-ulm.de
\textsuperscript{2} now at Ames Research Center, Mountain View, CA, USA
hutz@mail.arc.nasa.gov
\textsuperscript{3} Fraunhofer Institute for Autonomous Intelligent Systems, Sankt Augustin, Germany gerhard.kraetzschmar@ais.fraunhofer.de
\textsuperscript{4} University of Applied Sciences Bonn-Rhein-Sieg, Sankt Augustin, Germany gerhard.kraetzschmar@fh-bonn-rhein-sieg.de

1 Introduction

Application frameworks play a major role in fostering reusability of robotic software solutions. Frameworks foster the reuse of code and design and offer a convenient model of object-oriented extensibility. They provide a powerful concept for providing generic components of well-understood, modular solutions for robotics. Nevertheless, few such frameworks exist in robotics, especially at the application level.

Autonomous mobile robotics is a rapidly evolving field of research, and so far widely agreed upon standard solutions for subproblems have just begun to emerge. Furthermore, the substantial development effort required for developing and maintaining such an application framework has been avoided by the vast majority of research-driven projects, which tend to focus on the development of new ideas and concepts and the implementation of new algorithms.

When analyzing the situation in software development for autonomous mobile robots and the lack of reusable application-level infrastructure, one can identify two additional major obstacles on top of the issues already mentioned: Firstly, the hardware platforms used in autonomous mobile robots integrate a wide variety of different hardware, sensor, and actuator components, resulting in extreme heterogeneity. Secondly, mainly for efficiency reasons one can usually observe a tight coupling of software solutions to the low-level properties of sensory and actuator devices. This coupling is likely to jeopardize, if not destroy, all high-level portability of robotic software solutions. Thus, as described in Chapter Trends in Robotic Software Frameworks it is much more difficult to make the higher development effort for a framework-based
solution pay off in the long run by reusing the framework and its components for other scenarios or other robot platforms.

2 Middleware for Robots

Our research on software development in robotics addresses the intrinsic problems of this application domain and resulted in the design and implementation of MIRO, the “Middleware for Robots” [USEK02]. The basic idea of MIRO is to provide generic support for the management of the domain-specific difficulties by addressing them in form of a middleware-oriented architecture. For this purpose, MIRO provides several layers of functionality that reside between the bare operating systems and the robotics applications. The highest layer provides application frameworks for common tasks in robotics. The video image processing framework (VIP), which is the main subject of the discussion in this chapter, is part of this framework layer.

The frameworks make extensive use of the infrastructure provided by the middleware to ensure their portability, reusability, and scalability in long-term robotics projects. For these purposes MIRO provides a standardized distributed systems infrastructure, which was carefully configured and transparently extended in order to meet the requirements of multirobot teams, an advanced toolkit for configuration and parameter management, and generalized abstractions for sensor and actuator devices, which are modeled as network transparent services. So, before we discuss with VIP a particular framework in more detail, we will first survey some aspects of the lower layers of MIRO, in order to provide a clearer picture of their impact on framework-based development in robotics.

2.1 Communications Infrastructure

Robotics applications are inherently distributed. While this property is obvious in multirobot scenarios, single robots are also distributed systems. Many robot platforms are even equipped with multiple computational devices, encompassing embedded microcontrollers and PCs. They are often operated and monitored from remote workstations, or parts of the processing is offloaded to a powerful compute server. Therefore, we consider a distributed communication environment as an essential part of any robotics middleware.

The communication environment provided by MIRO is based on the CORBA standard. It provides a carefully configured, high performance communication environment [SGHP97] to the upper layers. System entities are therefore remotely accessible through object-oriented interfaces which are specified in the CORBA interface definition language (IDL). The design goal of the communications infrastructure provided by MIRO was to encapsulate