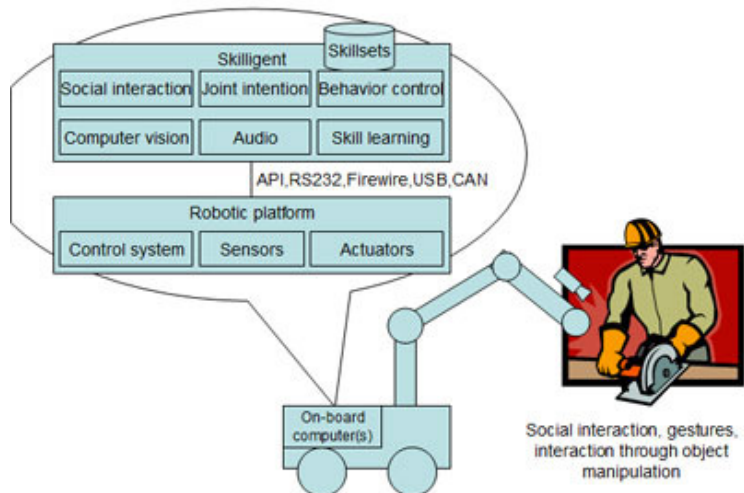


## Skilligent Robot Behavior Learning System

Skilligent Robot Behavior Learning System is a complete control system for autonomous service robots, including a behavior control and coordination system, task and skill learning functions, a powerful computer vision system, a social human-to-machine interface, a database for storing knowledge and other integrated components.

The control system enables solution providers to create autonomous robots which can learn procedures and skills directly from human users. The product is a set of software components specifically designed for straightforward integration into control loops of PC-controlled robots.



**Figure 1 A logical view of Skilligent software architecture**



**Figure 2 A robot learning a skill under control of Skilligent**

Skilligent is a set of software libraries (Figure 1) and a database. The open APIs of the robotic software library has been designed for straightforward integration into control loops of PC-controlled robots. Skilligent complements hard-wired robot control systems by playing a role of a high-level intelligence component which learns new tasks and skills (Figure 2) via interaction and collaboration with humans.

## Summary of foundation technologies

Technology	Advantage	Examples and use cases
Robot Task Learning from Demonstration	No need to program task sequences as the robot can learn those directly from humans (up to 80% reduction of the cost of ownership).	Learn how to pick up, process and deliver a part at a workshop.
Robot Skill Learning from Demonstration	No need to program complex control policies (controllers) as the robot can learn them directly from humans (up to 80% reduction of the cost of ownership).	Learn how to regulate temperature in a greenhouse by opening and closing windows according to sensed temperature.
Social Interaction with Humans	Users don't have to have a background in robotics to train or use a robot as the communication between the robot and the human is based on natural signals such as gestures (this opens new markets for robotics products such as small-and-medium size businesses)	If a human wants a robot do something, the human can simply show the robot what to do by manipulating relevant objects in the way that the robot can correctly guess the required behavior. To attract robot's attention, a human can simply wave by an object.
Hierarchical Behavior Control System	The robot is able to learn complex behaviors and skills as they are internally represented as a hierarchy of simpler behaviors and controllers.	If a robot is trained to water plants and clean floors, it can be trained to do both tasks in a sequence with optional conditional branches in the middle ("the green house" task)
Generalization of concepts based on actions	The robot can logically group and generalize learnt concepts to speed up the learning process.	If several objects were used as "watering-cans" during a training session, the robot can group those objects into a "watering-cans" category. This can help the robot execute a "find a watering-can" command or automatically suggest a replacement for a lost watering-can.
Skillssets Database	All behaviors, skills and concepts are stored on disk in a structured and interchangeable way.	If a robot is trained, the skills can be copied over to similar robots. The behaviors/skills database can be searched and changed by external programs if needed.
Built-in Visual Object Recognition	The software comes with a built-in visual object recognition sub-system; no need to buy/integrate with a third-party library.	The robot can recognize objects, their positions and orientations. This is used for visual servoing and interacting with humans.
Built-in Sound Recognition	The software comes with a built-in sound recognition and generation subsystem; no need to buy/integrate with a third-party library.	Sound recognition is used for executing behaviors and interacting with humans.

The software is capable of learning *hierarchical procedures* (also known as *tasks*), *skills*, *concepts*, *stimuli*, *social cues* and the relationships between them.

A *procedure* and a *skill* are the two kinds of *behaviors* which the software is capable of learning and executing.

A *task* is a step-by-step procedure with possible *branches*, *merges* and *hierarchies*, which leads to a concrete result. For example, "taking patient's blood pressure" is a *procedure*.

A *skill* is a low-level control rule which can be used as a step in a *task*. For example, "grabbing an object" is a *skill*.

A *concept* is uniquely recognizable and identifiable combination of sensory inputs. The *concepts* are used by *procedures* and *skills* as runtime parameters. The *concepts* are grouped together and generalized based on the way they are used by the behaviors.

A *stimulus* is a reason why an idle robot begins executing a *procedure* or a *skill*. For example, a voice command given by a user is a *stimulus*; a low battery signal is a *stimulus*. The software is capable of learning new stimuli and associating them with proper behaviors.

*Social cues* are a special kind of stimuli which provide a foundation of human-to-machine interaction.

## About Skilligent

Skilligent's flagship product is a trainable control system for autonomous robots such as mobile service robots, UGV, UAV or AUV. The software enables the robots to learn new behaviors, tasks and skills by observing how human operators accomplish the same tasks. A robot vision system, a part of the package, reliably recognizes objects, landmarks and gestures under real life conditions. The vision system enables the robots to visually navigate, control a manipulator and socially interact with the users.

<http://www.skilligent.com>