Computer Engineering Department

Robotics Laboratory

Professor Mayez Al-Mouhamed

Research Theme

A Multi-Threaded Distributed Framework for Telerobotics

Motivation

- A man-machine interface to extend eye-hand motion coordination to arbitrary distance and scale.
- Pervasive access anywhere and anytime through Internet or private network
- Electro-mechanical man-machine to extend human manipulative capabilities through a computer network
- Extend Eye-Hand motion coordination through arbitrary distance and scale
- Depth perception using stereo vision and augmented reality
- Extend force and haptic feeling by streaming force information as a reflected force feedback
- Real-time client-server to support Computer-Aided Teleoperation
- KFUPM Project AT-20-80

Approach

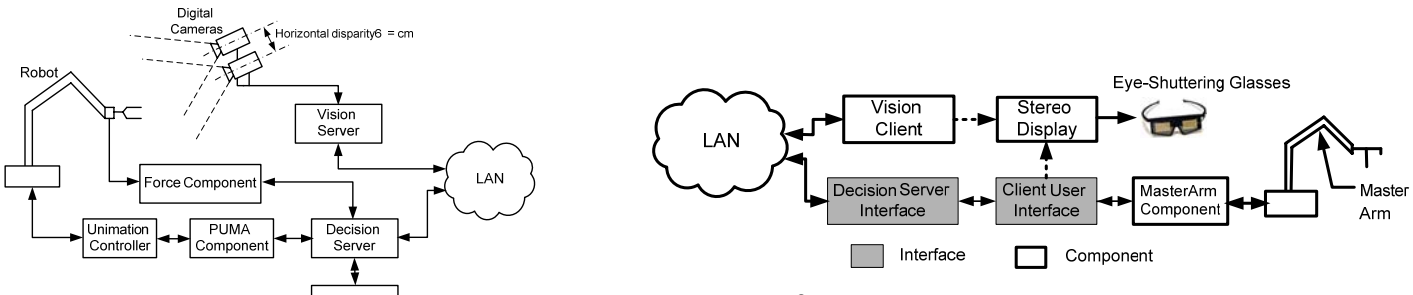
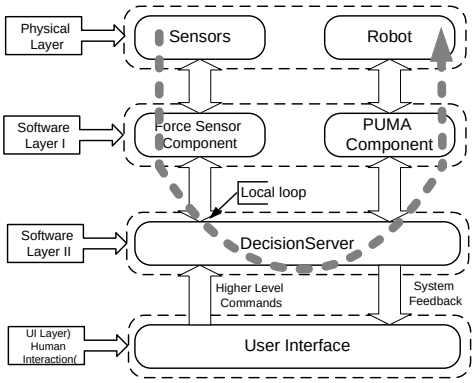


Figure 1 buted Component Framework for Telerobotics

1. ***Software Architecture***
   - Client-Server Telerobotics and Video Server: Object-Oriented Distributed Component System using Visual C#, .NET Remoting, DirectX, and Windows sockets
   - Client (master) and server (slave) interconnected to Internet
   - An integrated scheme of client-server components
   - Network programming for real-time communication
   - Real-time streaming of video data from server to client
   - DirectX, COM interfaces for graphic functionalities like DirectShow
   - Client support to Augmented Reality (AR) for superimposing graphics
   - Support to CAT operability, tool, active compliance at server, indexing, and scalability.

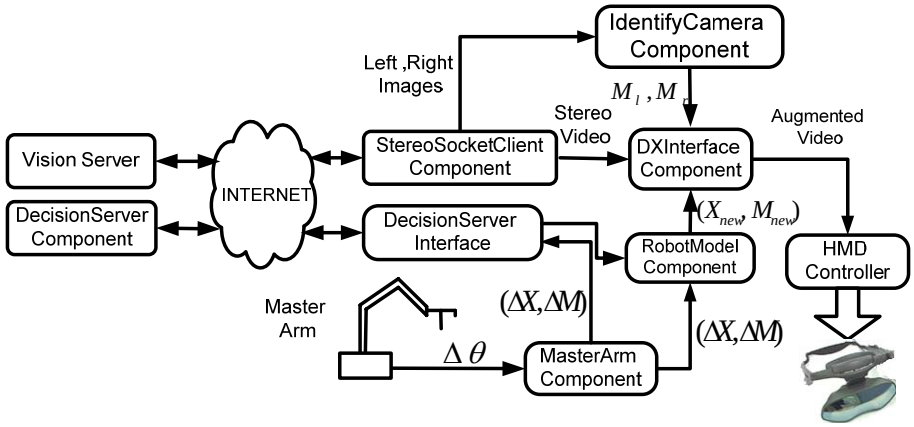
2. *Distributed Component System*

- Component (PUMA and Force): software proxy (thread) of server
- Public methods, ConnectRobot, InitializeRobot, Move, etc.
- Cartesian motion mapping at tool and world frames
- Public properties, Booleans for robot state, and Public Events
- Communication and synchronization mechanisms.



3. *Multi-threaded Execution*

- Simultaneous activation of many threads like
  - Thread Pipelining: Grabbing of two digital cameras at server
  - Communication thread: Transfer of video from Server to client
  - Force Thread: Streaming of Force from server to client operator hand
  - Command Thread: Rendering hand motion and transfer to server
- Network sharing: Video Server and the Distributed Component



4. *Multi-threaded Execution*

- DecisionServer component provides slave supervisory control
- Server Side Interfaces and .NET Remoting
- A component has a set of public methods, properties, and events
- Interfaces serve as a contract for any component which implements the interface.
- Shim Classes hide component assembly to increase security
- IProxyRobot and IForceSensor to communicate with PUMA and Force Sensor components
- IDecisionServer inherits both of the above interfaces
  .NET Remoting publishes an instance of DecisionServer component on the network

- Allows defining a unified set of methods, properties, and events within DecisionServer component
- .NET Remoting enables access to remote objects using SOAP.

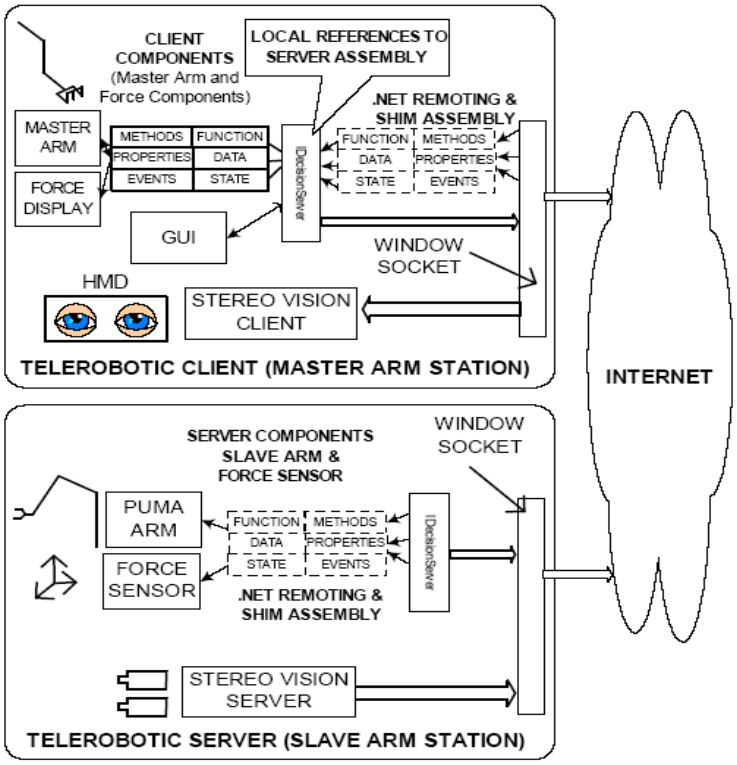


Fig. 1. Schematic diagram of our telerobotic system: real-time transfer of motion commands, force data, and live stereo video from SASS to MACS.

5. *Overall*

- A reliable and efficient man-machine interface between a Server Station (Slave) and a Client Station (Master) over the Internet
- OO distributed component framework:
  - Software reusability, ease of extensibility, debugging, and data Encapsulation
  - Automatic handling of network resources and data transfer (.NET remoting)
  - Isolation of components from network protocol issues
  - Enhances data security as well as facilitates deployment
- Stereo vision support, DirectX and Hardware Accelerated Graphics APIs
- Multi-threaded execution for multi-streaming of force, command, and for live stereo video data transfer