





Prof. Dr. Wilhelm Schäfer

Dipl.-Oec. Ahmet Mehić

Prof. Dr. Gitta Domik

INTELIGENT AGENT FRAMEWORK (IAF) FOR SOFTWARE **APPLICATIONS DEVELOPMENT**

DAAD – BiHSP – Scholarship program 2006/2007

Prof. Dr. Zikrija Avdagić, mentor Mr.Sci Dušanka Bošković, assistant Aida Delić, student Amir Smajević, student Dženan Zukić, student



Application: VolumeStudio software

- Scan types: CT, MRI, PET and their combination.
- Neural network trained to recognize



Project timetable:

O 2006/2007/Oct-April Sarajevo Theoretical research :

- Field: Artificial Intelligence
- Field: Software Engineering

O2007/May-July, Paderborn Practical implementation :

- Field: Artificial Intelligence
- Field: Software Engineering
- Field: Medical Visualization
- Field: Computer Graphics

🗖 VolumeStudio			VolumeStudio	
File Volumes Add New Volume	✓ Load Volumes Shaded	x, UInt1 gument		
Remove Selected Volume	Gradient Stretch 1.0			
pat18_low.nfo	Animate Interpolate			
	Still Quality			

- histogram shapes of heart CT (Computed Tomography) scans, and based on that position filters that make up 2D transfer function used in medical visualization.
- Histogram is scaled down 4 times vertically and horizontally, and it's top is cut off in order to lower resource usage and improve generalization capabilities.

Application: Automatic track layout

- Track-or road layout in a given geographical area (which could be, for example, a satellite scan) to plan new or improve existing public and/or private transportation systems is a complex problem which implies an explosion of search-space states.
- The development and implementation of

a special genetic algorithm to solve the mentioned track-layout problem forms the core part of this project.

- **Application: Program evaluation using** fuzzy logic with bad smells as inputs
- Development and application of a fuzzybased system to automatically evaluate the maintability and "smeliness" of classes based on bad smells.
- Code evaluation is based on identifying code pieces defined as bad smells. These are, for example, duplicated code, large classes defined and implemented by a large number of instance variables and methods, or have a high cyclomatic complexity, etc.