

MIMAS:
Modular Intelligent Multimedia
Analysis System
-introduction-
Jelena Tešić



Outline

- Introduction
- Related work
- Introducing MAMAS
- System Framework
- System Architecture
- **Decision Module**
- Implementation Example
 - Face detection algorithm
- Work plan
- Conclusion

Introduction

- Collaborative work with Yining Deng.
- A system that would integrate existing image classification algorithm and process the image data online.
- Objectives and possible application of the system required intelligent control and decision center.
- Learning component.
 - Automatic classification.
 - Efficient use and development.

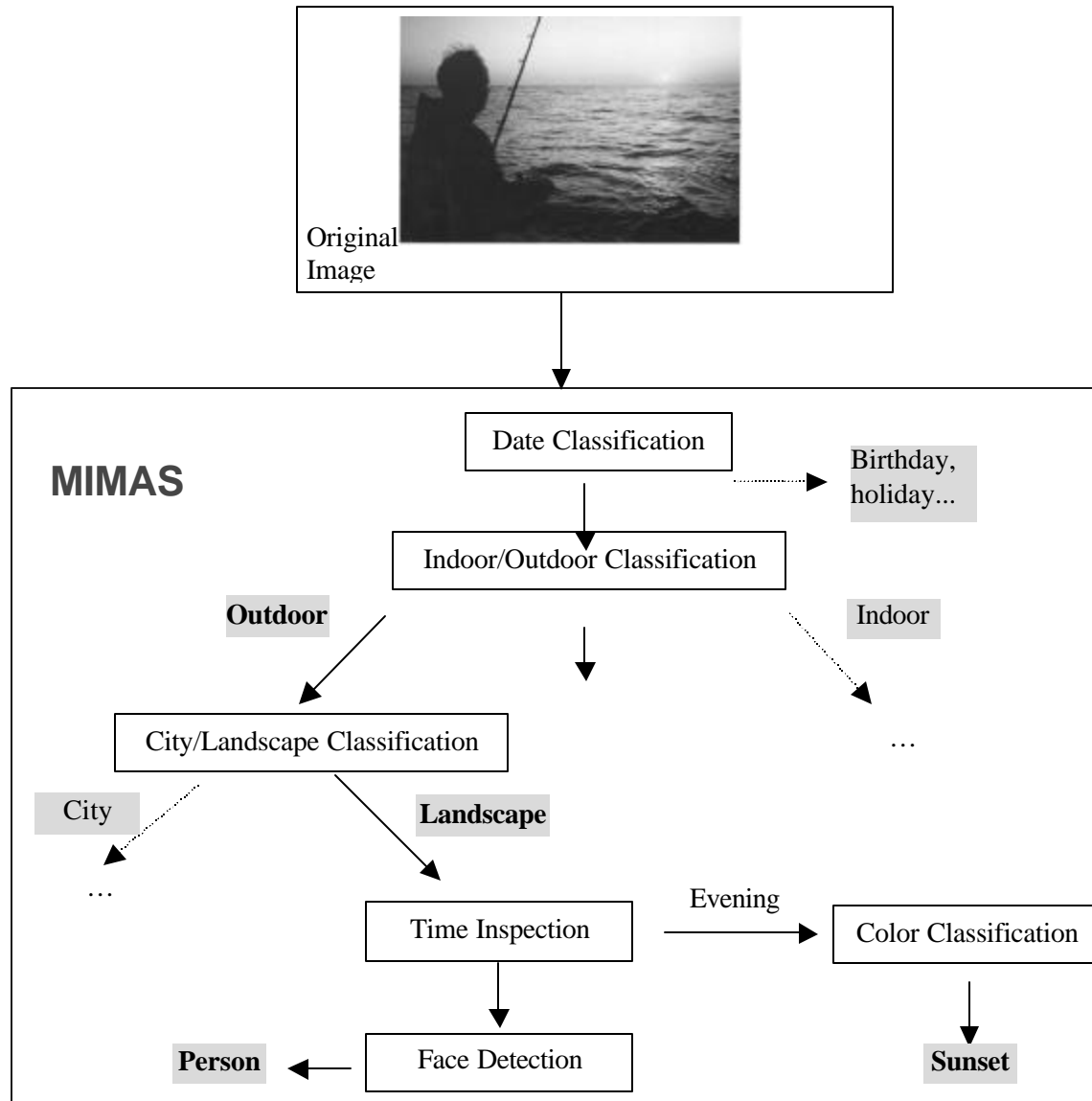
Related Work

- Multi-Media Miner (Simon Fraser University) :
 - Image and video databases
 - includes the construction of multimedia data cubes
 - facilitate multiple dimensional analysis of multimedia
 - mining of multiple kinds of knowledge
- QBIC (IBM Almaden):
 - image indexing and search engine
- Quest (IBM Almaden):
 - Intelligent alphanumeric miner

Introducing MIMAS

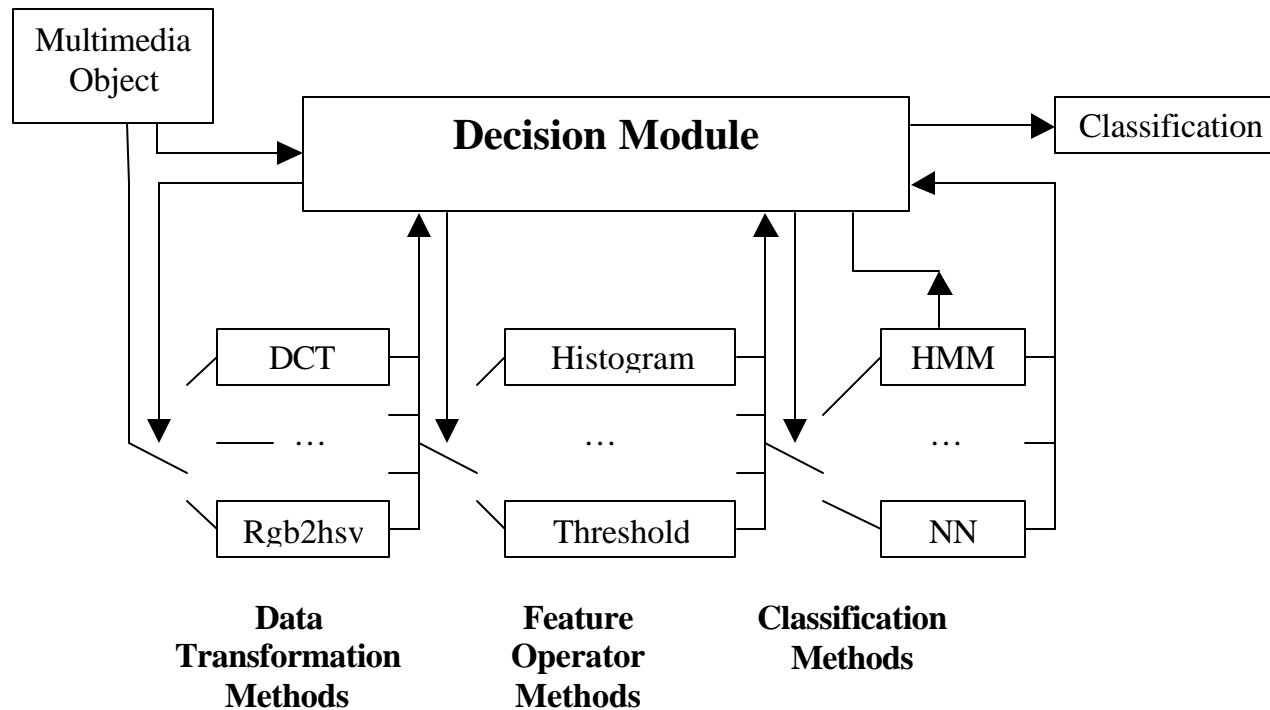
- Name origin:
 - one of Giants, son of Gaia (Earth) and Uranus (Sky).
 - born after the Olympians defeated the Titans; attacked the heaven.
 - killed by an Olympian god (either Ares or Hephaestus).
 - Heracles shot him with an arrow so he would perish (an oracle).
- A Modular Intelligent Multimedia Analysis System:
 - **Modular** – each module performs a set of methods or a task; easy to integrate new modules; distributed structure enabled.
 - **Intelligent** – optimal decision making; helpful development tool; enables extraction of higher semantics; accepts user's feedback.
 - **Multimedia** – supports image, video and audio data.
 - **Analysis** - classifies data based on its content and on any metadata associated with it.

A Service Example

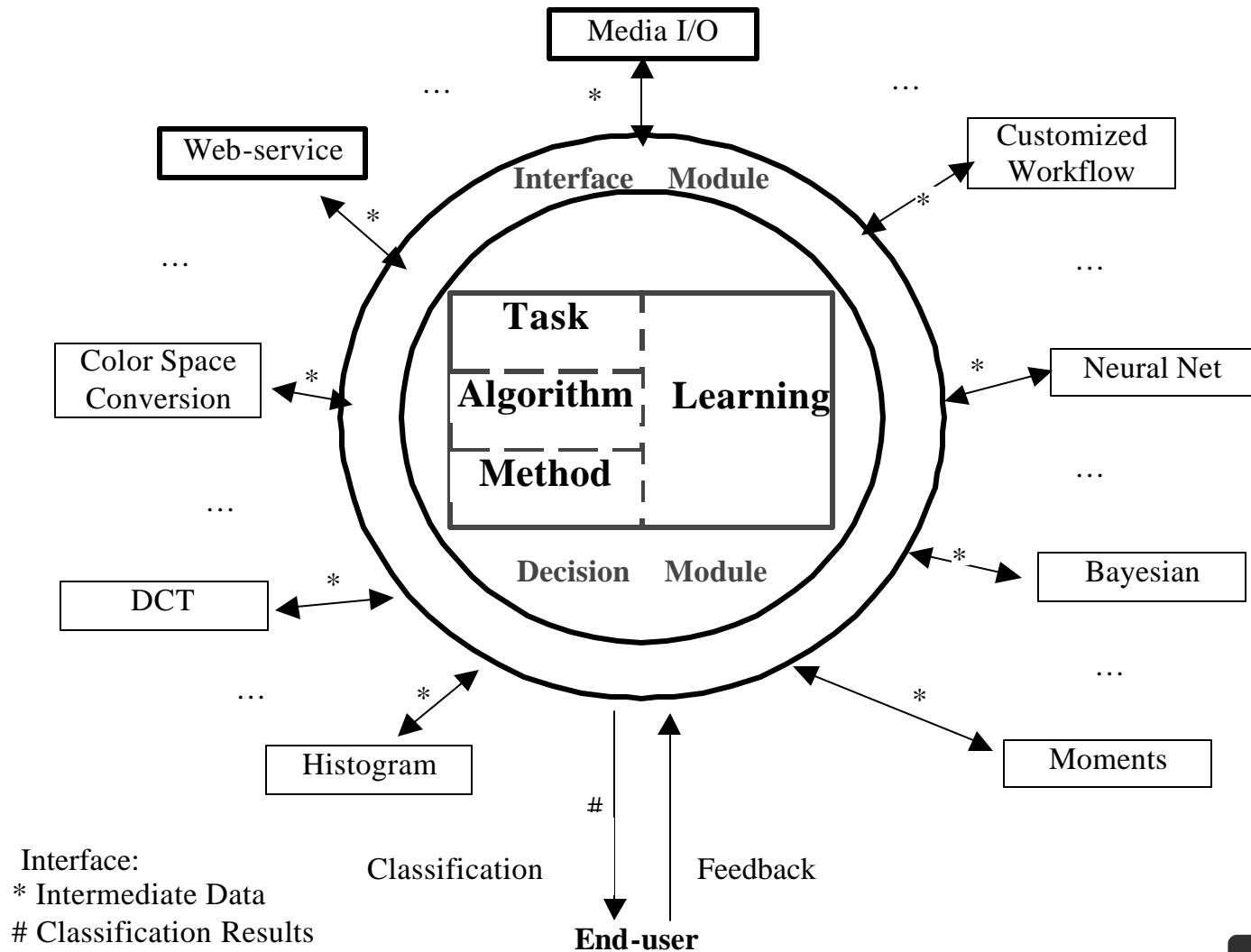


System Framework

- Centralized Decision Module
- Integrates different levels of decision
- Controls the interface between different methods



System Architecture



Interface Module

- The only way of communication among system modules and methods
- Enforces the execution of tasks, methods and algorithms in order specified by the decision module
- Collects the intermediate results and the information specified by the decision center

Decision Module

- User's perspective:
 - Accurate and/or real-time classification
- Developer's perspective:
 - Finding the best use for the new module in the system
 - Connecting it with the existing modules in optimal way
 - Manual control of the method flow based on intermediate results
- Three levels of decision:
 - Task, algorithm and method flow control

Algorithm Decision Component

Multiple algorithms for the same classification

- Decision based on requirement:

(different perspectives of user or developer)

SPEED – the fastest one

ACCURACY – parallel processing

BOTH – decision based on additional information

- Additional Information:
 - testing phase results and previous results
 - user's feedback
 - Other algorithm parameters

Method Component

- Data Transformation Methods
 - Wavelets, DCT, Gaussian, color space conversion.
- Feature Operator Methods
 - Histogram, moments, threshold, mean, etc.
- Classification Methods
 - Neural Networks, HMM, ML, Bayesian, etc.
- Lookup table:

Algorithm	Method ID			
Face Detection 1	10	270	...	1360
Face Detection 2	50	860	...	1790
Face Detection 3	90	630	...	1480

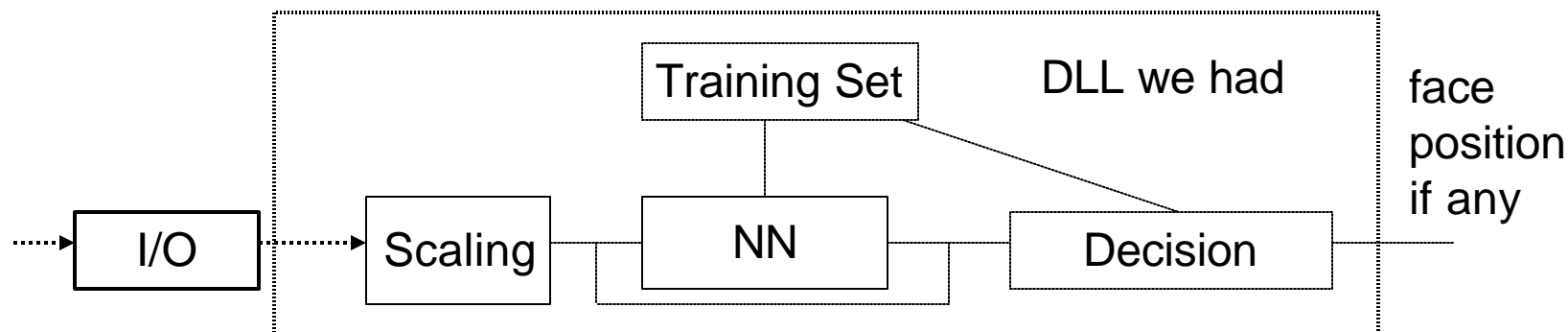
Learning Component

- User's feedback:
 - additional rules for pruning the task tree
- Intermediate results:
 - Influence the algorithm and the method component
- Developer:
 - Ability to impose decision rules
- Update on all decision levels

Task	Learning
Algorithm	
Method	

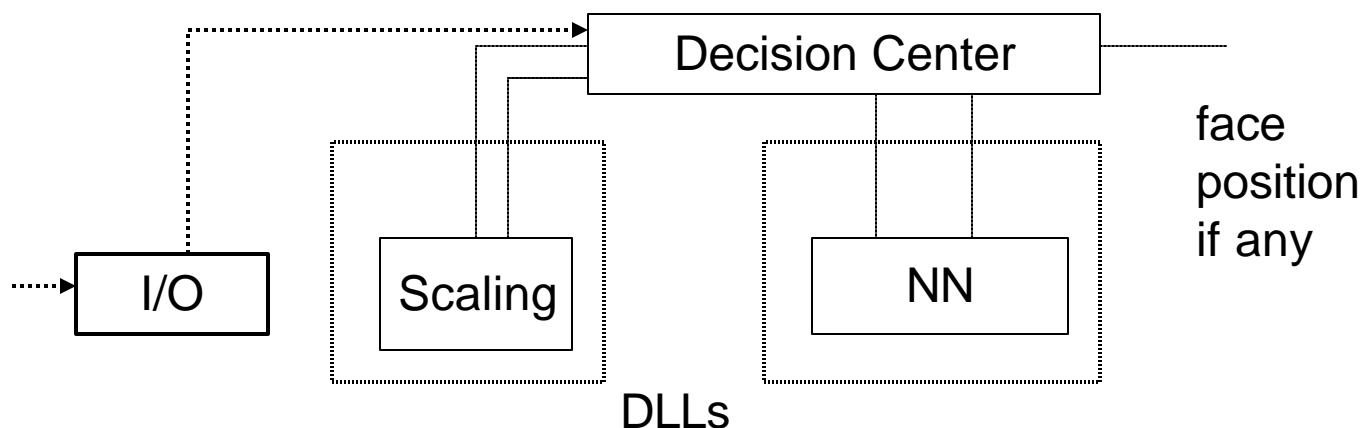
Implementation Example

- VC++ code
 - Modules – DLLs
 - Intermediate data – Data Class
- I/O module handles different types of images
- Data transformation Methods
- Face Detection Algorithm - breaking into modules



Implementation Example

- Scaling – data transformation method
 - Data class can handle the list of scaled images
- Neural network module is separated.
 - Weighted coefficients for face detection module are triggered by algorithm ID
- Decision is made in the decision center



Conclusion

- First phase of the implementation
- System has a modular architecture
- Useful tool from user's and developer's perspective
- Protected rights and benefits of service provider
- Intelligent part optimizes the process
- Lots of work to be done

Work Plan

- Finish the first version of the system prototype – interface module
- Implement the Decision module
- Test simple classification tasks
- Web-service module
- Learning module
- Distributed computing

Acknowledgments

- MAM project crew
- ITD people

Links

- Multimedia Asset Management:
<http://hplug.hpl.hp.com/mam/>
- Jelena's web page:
<http://eci.ucsb.edu/~jelena>

The End

Thank You