Proceedings of the 1st Korean-New Zealand Workshop on Advances in Computer Graphics, Computer Vision and Virtual Reality (GRAVIKON 2008)

Kyungpook National University (KNU)
Daegu, South Korea
24th April 2008

Editors: Burkhard Wünsche and Hyeyoung Park

Co-organisers:
BK21 Information Technology Manpower Development Program
School of Electrical Engineering and Computer Science, KNU
Software Technology Research Centre, KNU
Department of Computer Science, University of Auckland
Software Engineering Research Group, University of Auckland

Program:

9.00am-9.10am: Welcome

9.10am-10.30am: Session 1 – Augmented and Virtual Reality Environments
Youngho Lee, Taejin Ha, Hyeongmook Lee, Woontack Woo (Gwangju Institute of Science and Technology, Gwangju, Korea) - "3D Models and Sounds Authoring Methods for Digilog Book with Pen-type Tangible UI in augmented reality environments"

Kevin I-Kai Wang, Waleed Abdulla, Zoran Salcic, Ian Yen-Hung Chen, Neil Desouza, Sashank Ramkumar (University of Auckland, Auckland, New Zealand) – "Multi-Agent System Architecture for Controlling and Interfacing Smart Environments"

Bruce MacDonald, Toby Collett, Alex Kozlov, Ian Chen, Burkhard Wünsche (University of Auckland, Auckland, New Zealand) – "Augmented Reality Interfaces for Robot Development"

Stefan Marks, John Windsor, Burkhard Wünsche (University of Auckland, Auckland, New Zealand) – "Game Engine-Based Virtual Surgery Simulation"

10.30am-11.00am: Morning Break and Poster Session
Sungmoon Jeong, Minho Lee (Kyungpook National University, Daegu, Korea) – "Autonomous vehicle detector using spatial-temporal brightness changes based on a saliency map for a blind sport"

Qing Zhang, Minho Lee (Kyungpook National University, Daegu, Korea) - "Emotional Scene Understanding Using GIST"

Seokjun Lee, Soon Ki Jung (Kyungpook National University, Daegu, Korea) – "Planar-Object Position Estimation by using Scale & Affine Invariant Features"

Seonho Oh, Jaeseok Jang, Kyungho Jang, Soon Ki Jung (Kyungpook National University, Daegu, Korea) – "Building Modeling System on Satellite Image using Footprint and Shadow"

Min Woo Park, Kyoung Ho Jang, Soon Ki Jung (Kyungpook National University, Daegu, Korea) – "Image Registration of Side and Rear Views for Panoramic Vision System"

Seung Dae Jeong, Yoon Suk Kwak, Soon Ki Jung (Kyungpook National University, Daegu, Korea) – "Design and Implementation of Presentation Support System based on Mobile Networks"

Jae Seok Jang, Kwang Hee Won, Kyoung Ho Jang, Soon Ki Jung (Kyungpook National University, Daegu, Korea) – "Shadow Analysis of Satellite Images for 3D Building Reconstruction"

11.00am-12.00noon: Session 2 – Computer Graphics and Simulations
Kang Young-Min, Hwan-Gue Cho (Tongmyong University, Pusan National University, Busan, Korea) - "Interactive Paper Animation with Breakable Hinge Springs"

Namkyung Lee, Nakhoon Baek, Kwan Woo Ryu (Kyungpook National University, Daegu, Korea) – "An OceanWave Simulation Method Using TMA Model"

Dongsung Ryu, Hwan-Gue Cho (Pusan National University, Busan, Korea) - "A unified communication framework using geometry topology of virtual world space"

12.00noon-1.30pm: Lunch Break and Discussions (for invited participants)

1.30pm-3.10pm: Session 3 – Computer Vision and Image Processing
Sang Ok Koo, Chang Geol Yoon, Soon Ki Jung (Kyungpook National University, Daegu, Korea) – "Visual Analysis System for Pipeline Inspection Data"

Minho Lee (Kyungpook National University, Daegu, Korea) – "Biologically inspired visual attention, object perception and knowledge representation"

Woong-Jae Won (Daegu Gyeongbuk Institute of Science and Technology) - "Toward Biologically Inspired In/Out Vehicle Monitoring Model for Interactive Safety Driving Agent System"

Minkook Cho, Hyeyoung Park (Kyungpook National University, Daegu, Korea) – "A Subspace Method Based on a Data Generation Model with Class Information"

Chuljin Jang, Hwan-Gue Cho (Pusan National University, Busan, Korea) - "Management Method for concurrent digital photos using EXIF metadata"

3D Modes and Sounds Authoring Methods for Digilog Book with Pen-type Tangible UI in Augmented Reality Environments

Youngho Lee, Taejin Ha, Hyeongmook Lee, Woontack Woo

GIST U-VR Lab.
Gwangju 500-712, Korea
http://uvr.gist.ac.kr

GIST CTI

GIST U-VR Lab (since 2001)
13 Ph.D. students
6 MS students
GIST CTRC (since 2005)
GIST, KAIST, Sejong U., Catholic U.
USC ICT, CMU HCII, GATECH,
INRIA, Munchen U., EPFL,
HITLab NZ, Nagoya U.,
ETRI, KETI, KIST, etc.

GIST CTI (since 2007)
Supported by KOCCA
2Post doc, 3 Designers
2 Admin staffs
IDEG Lab
HMCI Lab
U-VR Lab

Introduction

e-Book

- Transform paper book into PDF or XML

u-Book

-text, figure, sound, movie, flash
-Mobile devices

Books with AR Technology

Convergence of Analog Book and Digital Content

Books with AR Technology

Magic Book

- HIT Lab NZ
- IEEE CG&A, 2001

Little Red MR

- Keio University
- SIGGRAPH, 2003

How can we make these books?

Vivid Encyclopedia

- Ritsumeikan Univ., Japan
- 2004

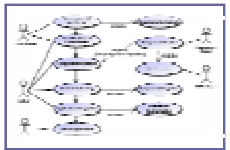
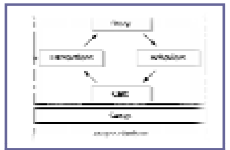
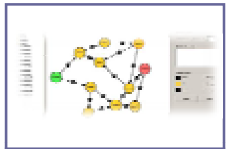
Virtual Pop-up Book

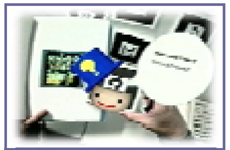
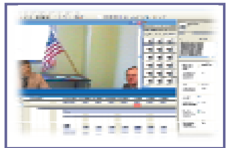
- Osaka University/NAIST
- HCII, 2007

Digilog Book

- GIST U-VR Lab.
- Korea HCI 2008

Related Works

AMIRE	APRIL	U-Create	
			Augmented Reality Authoring
The AMIRE (Authoring Mixed Reality) framework, EU IST Programme (EU)	APRIL (Augmented Presentation and Interaction Authoring Language), Graz University of Technology (Austria)	U-Create, FP6 (Research project of the Sixth Framework Programme) (EU)	

DWARF	Tiles	CATOMI	DART
			
DWARF, Technische Universität München (Germany)	Mixed Reality Authoring Interface	CATOMI, Upper Austria University of Applied Sciences (Austria)	DART (A Toolkit for Rapid Design Exploration of Augmented Reality Experiences) , Georgia Institute of Technology (USA)

Digilog Book

- Digilog Book
  - It provides additional information and interactivity to readers by converging paper book and multimedia content which stimulate human's visual, auditory, and haptic senses with AR technology in AR environment

ARtalet: Digilog Book Authoring Tool

- ARtalet: Digilog Book Authoring Tool
  - People who don't have programming skill
  - ARtalet means…
    - { AR(augmented reality) + tale(story) }
    - Art + -let(booklet)
    - Atelier
- Visual Authoring with ARtalet
  - **Ideas from Desktop UI (Windows program)**
  - **3D bar-type wireless input device**
  - **Object selection UI (2D + 3D UI)**
  - **Objects copy from folder to workspace**
  - **Translation, rotation, scaling, coloring**

Overall workflow

A user buy a book → Install Authoring tool → Search related content from DB → Download → Load content → Register pages → Authoring (Visual, Sound, Hpatic) → Preview test → Save → Viewer (reading)

Authoring process

- Scenario


<Download content>


<Positioning>


<Manuplation>


<Evn. Setup>


<Animation>


<digilog book viewer>

Digilog Book Authoring

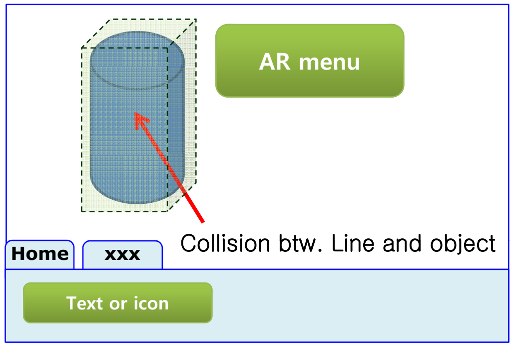
Video

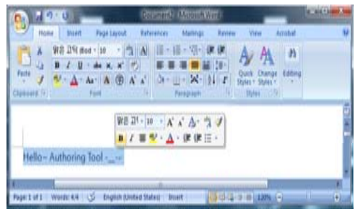
ARTalet: 3D bar-type input device

- Input system
  - Wireless mouse
  - Multi-marker in front
  - Ray-casting
- Degrees of freedom
  - 6DOF
- Input type
  - Discrete input: event trigger
  - Continuous input: 3D Pose tracking, object property changing

ARtalet: UI

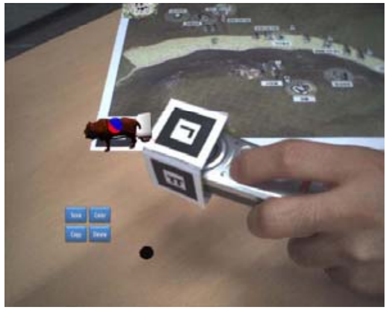
- User Interface
  - 2D GUI + 3D AR menu
  - Composing advantages of both 2D and 3D UI





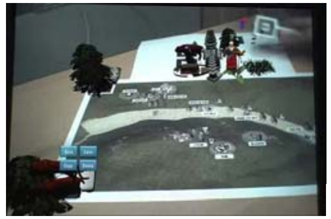
ARtalet: Object selection

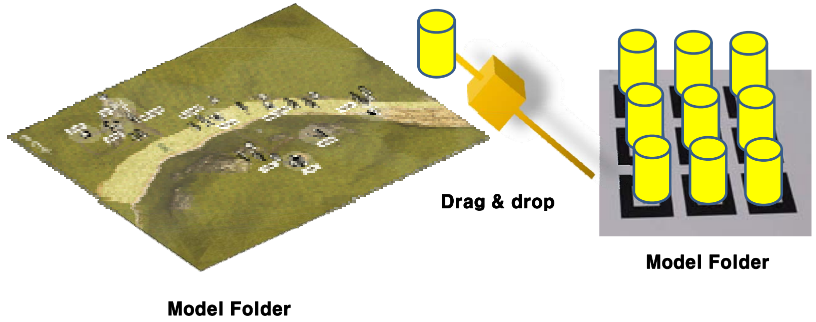
- Selection
  - Ray casting with simple line
  - Feedback: edge lines, menu, sound
  - collision: distance between object and end point of line and mouse click



ARtalet: Object Copy

- **Objects copy from folder to workspace (book)**
  - File folder metaphor
  - Drag & drop



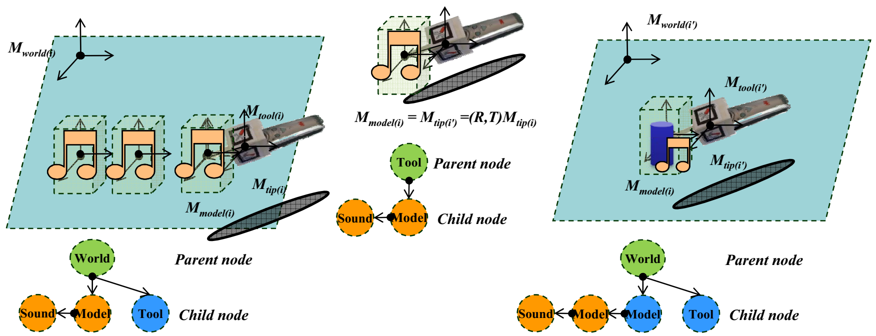


ARtalet: trans/rotation, scaling

- Translation/rotation
  - Select an object → press button→ move it → release button
- Scaling
  - Select an object → press button→ move it while pressing → release button
- Coloring
  - Select an object → press button → color box → move device → release button

ARtalet: Sounds

- Sound Copy
  - Drag and Drop



Conclusion & Future works

- Conclusion
  - ARtalet: Digilog Authoring Tool
  - Visual Authoring with ARtalet
  - Intuitive and Natural 3D user interface

- Future works
  - Natural Feature Tracking
  - Combination with 2D UI
  - Character animation and events authroing

Thank you!!

Discussions & More information

- GIST CTI, Gwangju 500-712, S. Korea
- Tel. (062) 970-3157
- Fax. (062) 970-2204
- mailto: ylee@gist.ac.kr
- Web: http://uvr.gist.ac.kr, http://cti.gist.ac.kr

Multi-Agent System Architecture for Controlling and Interfacing Smart Environments

1st Korean-New Zealand Workshop on Advances in Computer Graphics, Computer Vision and Virtual Reality

Kevin I-Kai Wang, Waleed H. Abdulla, Zoran Salcic, Ian Yen-Hung Chen, Neil Desouza, Sashank Ramkumar

Electrical & Computer Engineering Department
The University of Auckland
April-2008

Overview

- Background
- Control system architecture
  - System requirements
  - Technologies & communications
  - Control modes
- Physical ubiquitous platform – DEIR
- 3D virtual environment
  - Design process and technologies
  - User interactions
- Achievements
- Future works

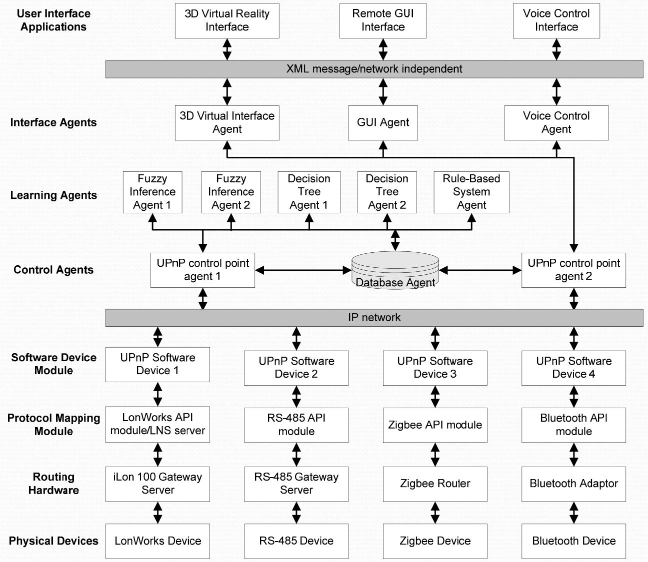
Background

- An Intelligent Environment (IE) basically should have the following capabilities:
  - User interaction
  - Context-awareness
  - Learning and adaptation
- To construct an IE application, we need:
  - A central control system
  - An ubiquitous physical environment which provides the require infrastructure for the control system
  - Various man-machine interfaces

Control system criteria

- An IE application generally has
  - Distributed devices
  - Multiple communication protocols
  - Multiple users and user interfaces
- A decentralized control system serves the IE applications better, due to the distributive, multiple access nature of IE applications
- Multi-agent system (MAS) provides standardized runtime platform and communication protocol for handling distributed environments, which shortens the design and development process of a control system
- Control system of an IE application should
  - Allow fast integration of new physical gadgets
  - Provide standard communication for user interfaces
  - Allow easy employment of different learning modules and control routines

Control system architecture



Technologies used - JADE

- JADE (Java Agent DEvelopment framework) is a Java-based, FIPA-compliant agent development tool, it consists of:
  - Agent runtime platform
  - Java libraries for agent and MAS development
- The nature of MAS allows group and society formation. Multiple algorithms can be applied for different device groups within the same control system for learning and controlling different tasks.
- Within the multi-agent control system, different learning algorithms and control routines can be encapsulated as agents easily
- Agents can be started up dynamically without shutting down the whole control system

Technologies used - UPnP

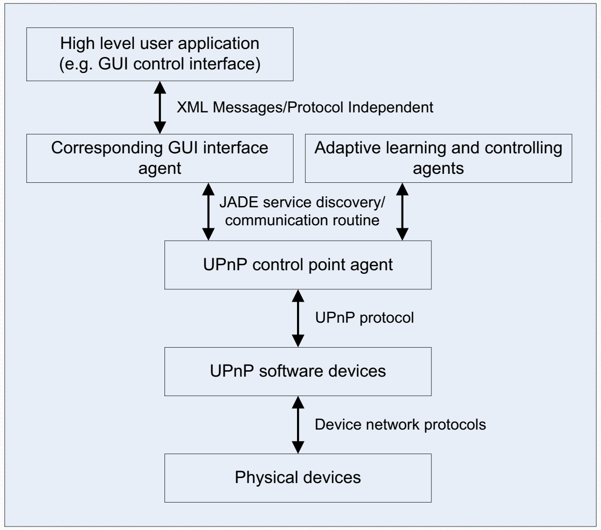
- UPnP is an IP-based service discovery protocol which enables automatic device discovery, configuration and management
- Universal Plug and Play (UPnP) is used as an middleware layer between the device control software and high level MAS
- UPnP protocol consists of 2 parts:
  - UPnP control point is encapsulated as agent, which provides an unique communication portal between the control system and physical devices
  - UPnP software devices are the software replica of the attached physical devices, which communicate with the physical devices using specific protocols.

Technologies used - XML

- XML is used to design the content language for the communication between user interfaces and the multi-agent control system
- Standardized content language
  - Allows different protocols to be used by different interfaces
  - Isolates the design of user interfaces and control system

```
<? xml version="1.0" encoding="utf-8"?>
<devicelist xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="file://c:/fileLocation /devicelist.xsd">
<device>
<name>Device name</name>
<currentState>Device status </currentState>
</device>
</devicelist>
```

Communication stack



Control modes

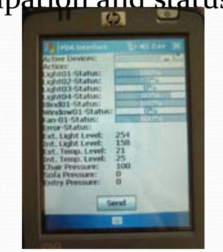
- User control mode
  - User data is collected
  - Automatic control is disabled
  - Typically for less than 3 days
  - System learns user preferences
- Auto control mode
  - User behaviour modelled and rule base constructed
  - Auto control applied based on rule base
  - User has overwrite priority
  - System adapt to user overwrite

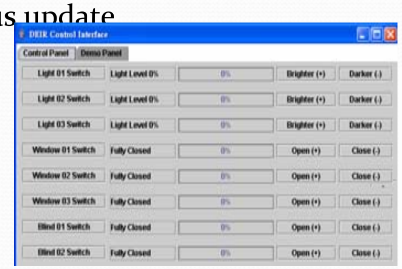
DEIR

- A physical ubiquitous testbed named Distributed Embedded Intelligence Room (DEIR) is constructed.
- It contains a number of embedded devices including light, temperature, energy, pressure, smoke, air quality, and occupancy sensors; and actuators for automatic blinds, windows, projector and dimmable lights
- In the current prototype, LonWorks network, RS-485 network, IP network, Bluetooth and Zigbee are used for interconnecting all the physical devices

DEIR

- Different user interfaces are provided to hide the existence of complicated physical gadgets and to enable friendly controls
- Interfaces from traditional switch (digitized), GUI, PDA interfaces to 3D virtual environment have been designed and developed
- Each interface has its own communication agent, handling command dissipation and status update





3D virtual environment - motives

- Physical environment is difficult and time consuming to construct, which makes short research project infeasible
- Physical environment may not be the ideal testbed during the development process
- Realistic 3D environment can act as a real-time simulator which provides idealised testbed for development and debug process
- 3D interface makes remote accessing and monitoring of the physical room possible without cameras

THE UNIVERSITY OF AUCKLAND
FACULTY OF ENGINEERING

3D virtual environment -Design

- A 3D model is constructed in 3dsMax with real textures and dimensions of the room
- Ogre3D graphical engine dynamically renders the 3D model in run time –
  - Pros:
    - 1) 3D model can be easily modified according to the change of physical room (or even different physical room)
    - 2) 3D model can be designed by someone with no knowledge on the system architecture
  - Cons: Client side requires more computational power
- Physics properties are provided by additional physics engine and network libraries.

GRAVIKON 2008

THE UNIVERSITY OF AUCKLAND
FACULTY OF ENGINEERING

3D virtual environment

Physical DEIR:
obtaining accurate dimensions and texture of the room and furniture

Modelling:
specifying shape, colour, lighting and material properties

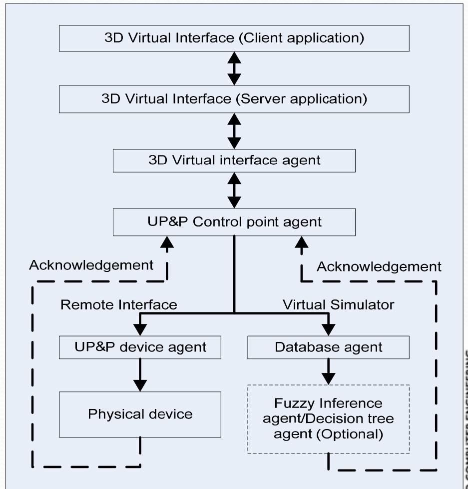
Virtual Environment:
real time graphics rendering, physics calculation, networking, and handling of all user interactions

GRAVIKON 2008

THE UNIVERSITY OF AUCKLAND
FACULTY OF ENGINEERING

Operation modes

- The current 3D virtual environment has two operation modes:
  - 1) The virtual simulator mode is designed to simulate the behaviour of DEIR in absence of the actual physical environment
  - 2) The virtual interface mode acts as a 3D remote control interface which allows for teleoperating the devices installed in DEIR



GRAVIKON 2008

User interactions

- Currently, interaction with the virtual environment is performed through mouse and keyboard
- A controllable human character in the virtual environment can be moved around using keyboard and mouse
  - The virtual human character allows user to see the 3D environment through different camera modes

Camera modes

- 1) Third Person Chasing
  - Camera follows the virtual character from behind
- 2) Third Person Fixed
  - Camera attached to top right corner looking at the virtual character

1) Third Person Chasing

2) Third Person Fixed

Camera modes

- 3) First Person
  - Looking out from the virtual character's perspective
- 4) Free Look
  - Capable of moving arbitrarily around DEIR and not affected by the laws of physics
- Free look mode is for navigation purpose, control of device is not possible under this mode

3) First Person

4) Free Look

Achievements

- UPnP middleware allows interoperation of physical devices over different physical networks
- XML content language provides standard communication and allows fast integration of different user interfaces
- Multi-agent control system architecture provides excellent handles for distributed physical gadgets and multiple user interfaces
- Physical ubiquitous environment is constructed to allow real-time simulation and experiments
- 3D virtual environment is developed to act both like a virtual simulator and remote user interface

Achievements

- The designed communication stack has been proved to provide a real-time control and status feedback loop between the user interface and physical devices
- The multi-agent control system tied the physical environment and user interfaces as a complete IE application
- Different learning algorithms, user interfaces, and physical gadgets can be integrated into the control system architecture with minimum effort which encourages collaboration

Future Work

- Control system:
  - Other machine learning algorithms can be applied and tested
  - Longer term experiments should be carried out
- DEIR:
  - Different physical network protocols should be incorporated
  - Different household appliances and mobile gadgets should be involved in the experiments
- Interfaces
  - More man-machine interfaces should be implemented
  - Internet accessibility should be provided to achieve a true remote interface

Questions?

Thank You

Augmented Reality Interfaces for Robot Development

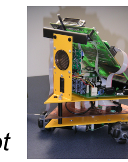
Gravikon 2008
Kyungpook National University
Daegu, South Korea
24 April 2008

Bruce MacDonald
Toby Collett, Alex Kozlov, Ian Chen, Burkhard Wünsche
Robotics and Intelligent Systems Laboratory
Department of Electrical and Computer Engineering
b.macdonald@auckland.ac.nz
http://robotics.ece.auckland.ac.nz

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND
Te Whare Wānanga o Tāmaki Makaurau

Outline

- UA Robotics and Intelligent Systems Lab
- Robotic software development
- Augmented Reality for:
  - Robot sensory data
  - Navigation algorithms (SLAM)
  - Mixed Reality simulation
  - Applications



"Shuriken"
Teaching robot

Robotics and Intelligent Systems Lab

- Long term goal: robot assistants for people
- Three main areas:
  - Robot Programming Systems
  - Human Robot Interaction
  - Applications: Healthcare and Agriculture
- Healthcare: NZ-Korean Centre for Aged Care
  - Link with South Korea: joint project with ETRI
  - UA researchers in robotics, IT, healthcare, health IT
  - NZ Health IT companies, US companies
  - ETRI, Korean Robot Companies

Multidisciplinary team

Bruce MacDonald,ECE	Robotics and Intelligent Systems
George Coghill,ECE	Artificial neural networks
Catherine Watson,ECE	Robotic Speech
Waleed Abdulla,ECE	Speech recognition
Karl Stol,Mech	Robot Navigation
Burkhard Wünsche,CS	Graphics and Visualisation
Liz Broadbent,Psych Med	Psychology in healthcare
Jim Warren,NIHI	Health Informatics
Karen Day,NIHI	Health Informatics
Martin Orr,NIHI	Health Informatics
Martin Connolly,Ger Med	Gerontology
Ngaire Kerse,Gen Practice	Gerontology
Mark Fisher,Middlemore	Geriatric Psychology
Gary Putt,UniServices	Business development
John Corey/John Hosking,CSI	Business development
Sarah Haydon/?,UniServices	Business development
Malcolm Pollock,NIHI	Business development

Vacuuming robot

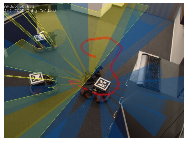
Capabilities

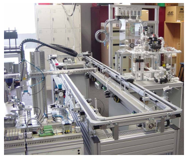
Robot face

- Mobile robot programming and control
- Robot programming systems
  - development environments
  - distributed programming
  - programming languages
- Programming by demonstration
- Emotional dimension of robotics (for speech and face)
- Perception augmentation using AR
- Visualisation
- Speech
- Navigation and coverage algorithms
- How are people's thoughts and feelings about robots influenced by the robot's behaviour? (Psychological studies)
- Evaluating robots in healthcare scenarios
- Applications in healthcare and agriculture

Facilities

- 2 large mobile robots
- 7 indoor pioneer robots
- 1 outdoor pioneer robot
- Helicopter robot
- Fiducial tracking system
- Debugging space
- Mechatronics testbed (Valeriy)





Robotic Software Development

- Currently developer tools are ad hoc
- Mobile robot environments:
  - Uncontrolled, dynamic, real world, unexpected variation
- Mobile robots
  - Change pose & relationship with environment and users
  - Large number of inputs and outputs
  - Variations in hardware and interfaces
- Tasks
  - Emphasize 3D geometry, complex data types, high dimension spaces and paths
  - May not be interruptible, multiple simultaneous activity

Robotic Software Development

- Programming languages
- Middleware & Libraries (Player)
- Tools

Programming languages for robotics (Geoff Biggs)

```
from time import sleep

event NearWall (sonar):
  for range in sonar.ranges:
    if range < 0.25~m:
      returnVal = range.index
      trigger

event HitWall (bumpers):
  for bumper in bumpers:
    if bumper == 1:
      trigger

```

Reactive programming

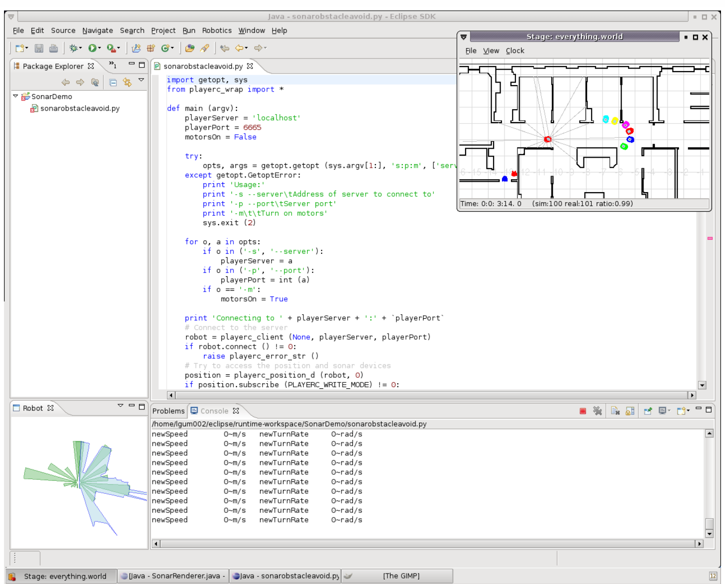
Dimensional analysis

```
response UpdatePlayer (setSpeedFunc, speed):
  while True:
    setSpeedFunc (speed.getval ()[0], \
                  speed.getval ()[1])
    sleep (0.05~s)

response Drive (speed):
  speed.setval (0.5~m/s, 0~rad/s)
  while True:
    sleep (0.5~s)
    interrupt      # Check for interrupt @ 2Hz

...............
```

Eclipse based robotic software IDE (Luke Gumbley, Steve Hsiao)



Augmented Reality (Toby Collett)

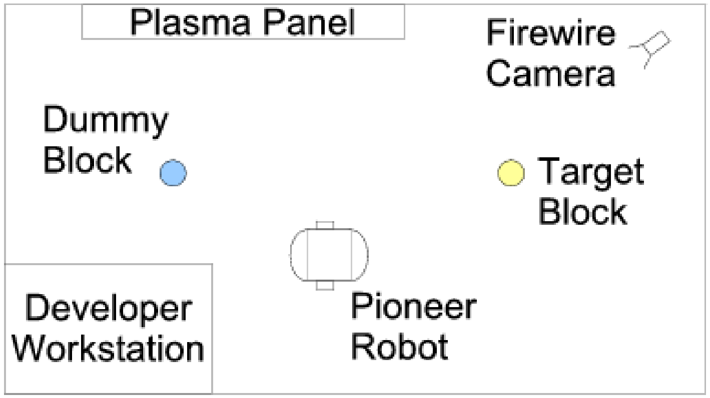
- Debugging is a key process:

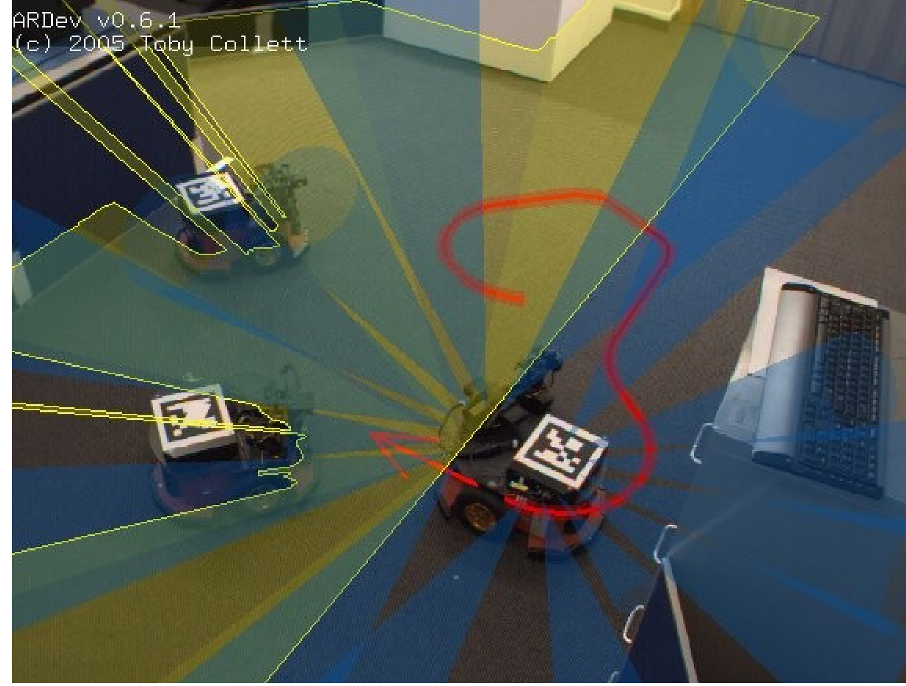
The programmer needs a clear understanding of the robot's world view

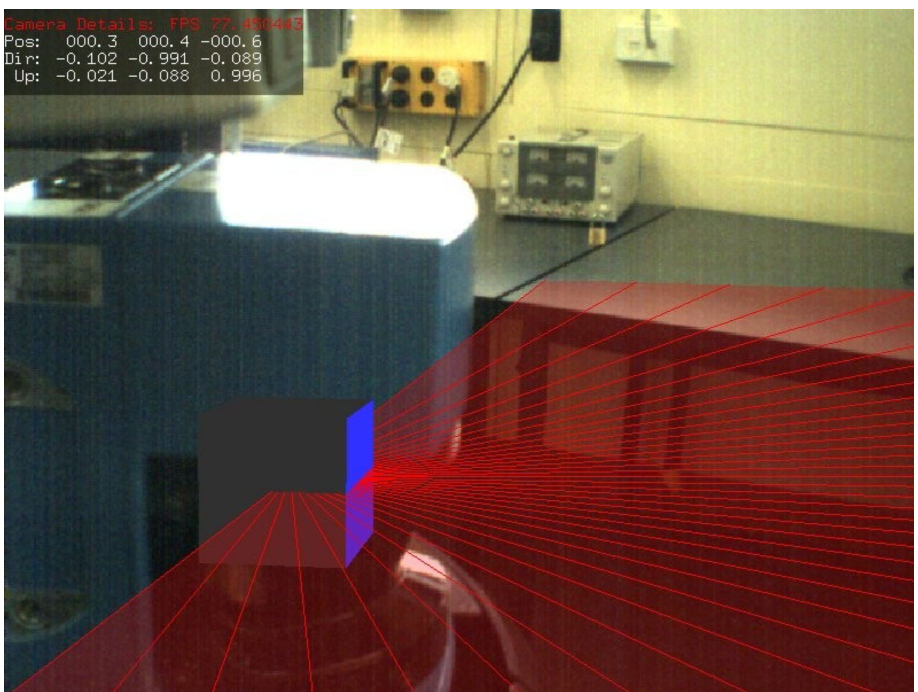
- Augmented reality for interacting with robots
- Targeted at developers
- Targeted at showing robot sensory data
- Increasing “perceptual overlap”
- Head mounted display OR large plasma display

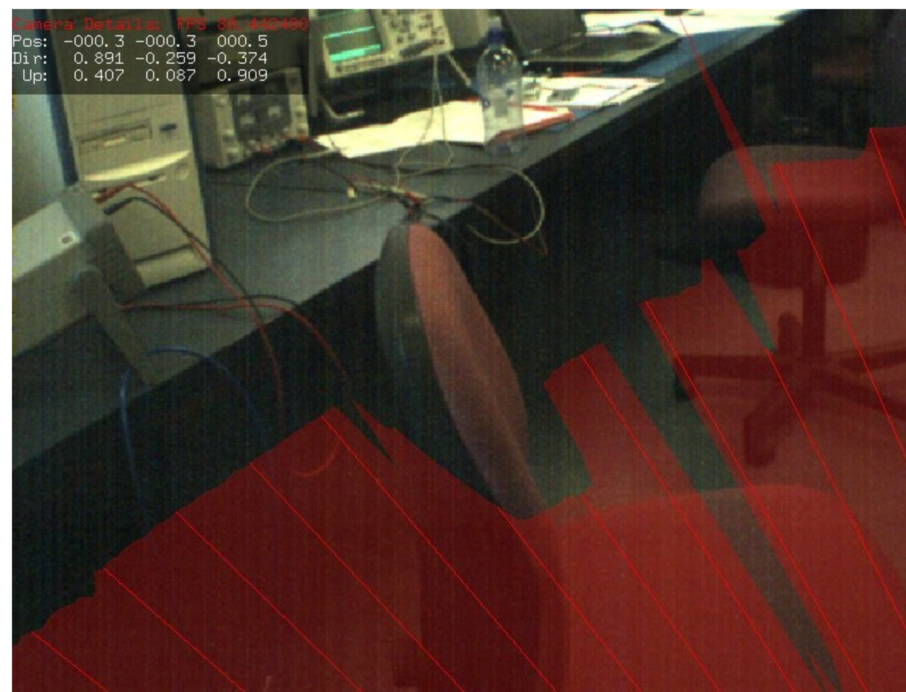
Presented at HRI2006, ICRA2006

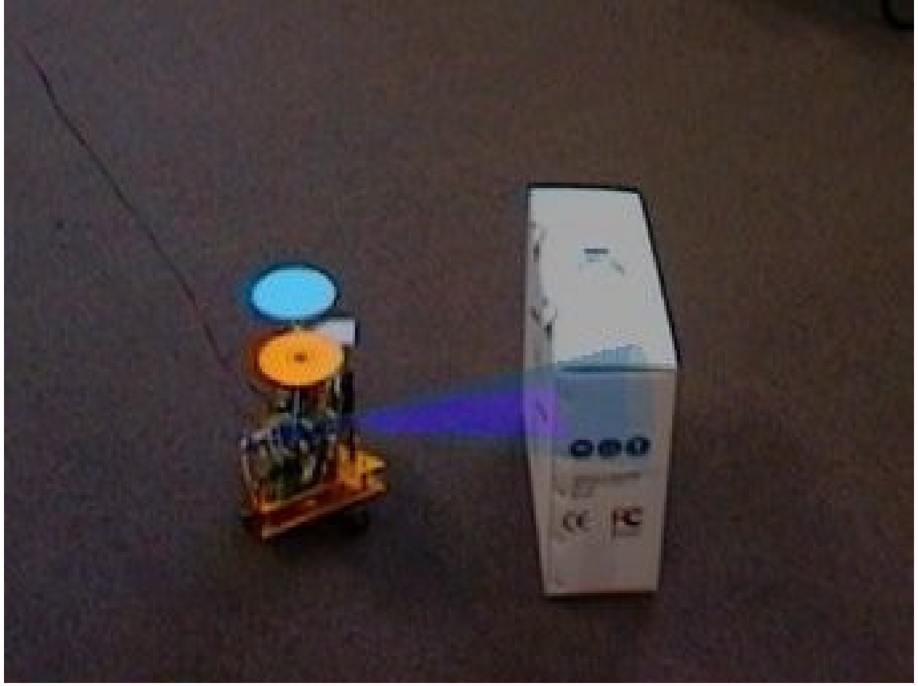
AR Debugging Space

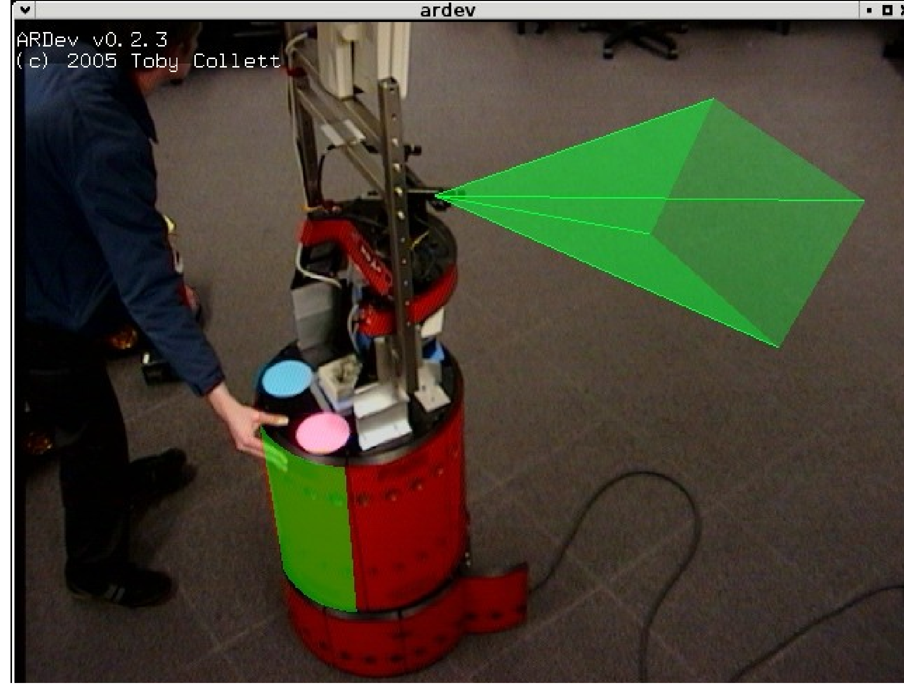


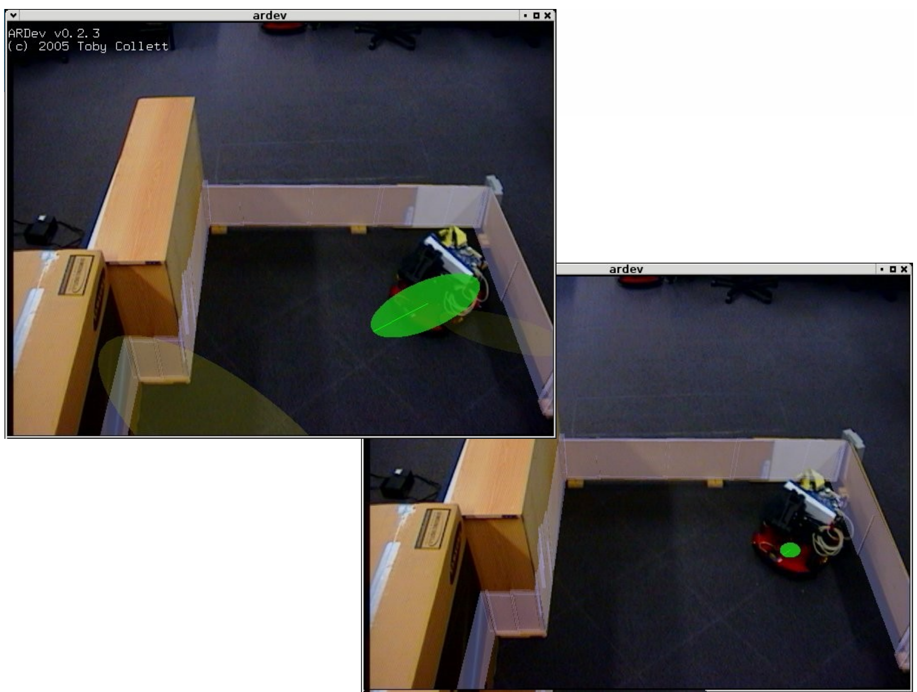












8 Stages for AR software

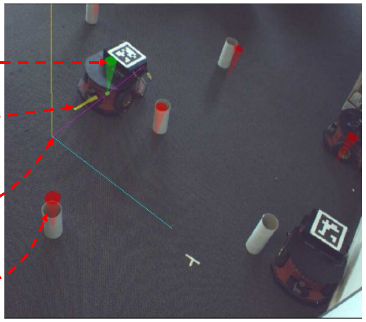
- *Capture*: background frame, orientation and camera position
- *Pre-processing*: e.g. blob tracking for robot registration
- *Render - Transformation*: view transforms are applied to the render object
- *Render - Base*: invisible models of known 3D objects are rendered into the depth buffer. Eg so the robot obstructs the virtual data behind it
- *Render - Solid*: the solid virtual elements are drawn
- *Render - Transparent*: transparent render objects are drawn while writing to the depth buffer is disabled
- *Ray trace*: to aid stereo convergence, the distance to the virtual element in the centre of the view is estimated using ray tracing.
- *Post-processing*: eg encoding to a movie stream.

Augmented Reality for Simultaneous Localisation and Mapping (Alex Kozlov)

- We can also use AR to view the internal state of a robot
- For example, of a navigation algorithm
- Analysing SLAM estimation:
  - State vector: robot pose and feature map
  - Covariance matrix: uncertainties and correlations
  - Data association: sensed feature matches with map
  - Predicted maps, dynamic objects, pose trajectories, scan matching data

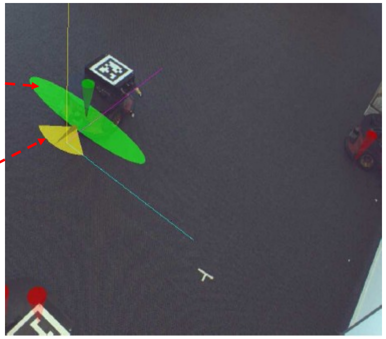
SLAM implementation: Alan Yang Alex's Preliminary Results – *Prototype*

- Operation
  - Green marker – robot position
  - Yellow marker – robot orientation
  - The axes - origin of the map
  - Red marker – feature position



Preliminary Results - *Prototype*

- Expected behaviour
  - Green ellipse – robot position covariance
  - Yellow sector – robot orientation covariance
  - No features
    - High uncertainty

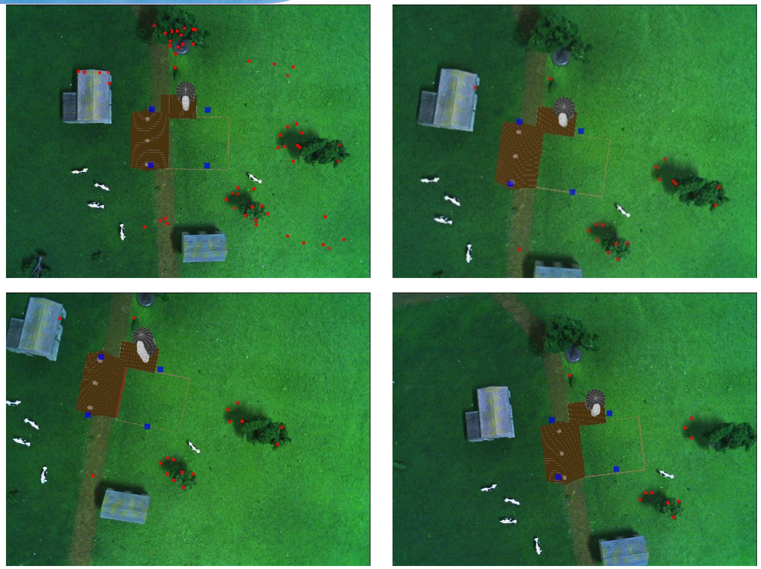


Presented at ACRA 2007, submitted to IROS 2008

Mixed Reality Simulation for UAVs in Agriculture (Ian Chen)

- Objectives
  - To assist the current UAVs research by providing a 3D environment for real time visualisation and simulation
  - Effectively communicate useful feedback to robot application developers
  - Initially: Using AR to enable markerless tracking
    - KLT feature tracking
    - Projective reconstruction
    - Testing on a mockup farm

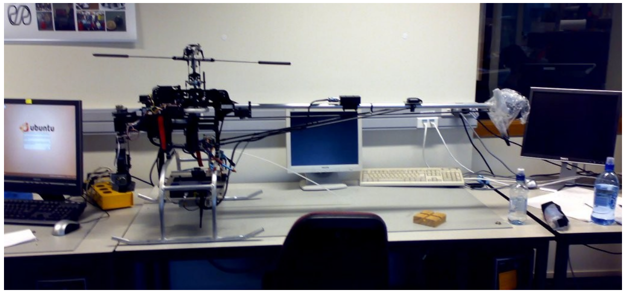
Prototype – Registration



Presented at Robot Vision 2008,LNCS, Springer

Application: Robotics in agriculture (Rick Chen)

- Helicopter project
- Tracking animals
- Monitoring fields and animals
- Interest from NZ IT companies in agriculture
- Mixed reality simulation and programming by demonstration



Rotomotion SR20 unmanned helicopter

UoA Healthcare Robotics Project (Tony Kuo)

- With Dr Liz Broadbent in Psychological Medicine
- Human reactions to good/bad robots: IROS 2007
- Student project in 2007, now a new PhD project
- Initially:
  - Taking blood pressure
  - Taking pulse
  - Taking temperature
  - Reminder service for medication
  - Networked communications to health services

- Shortly: taking blood samples, psychological evaluation
- AR for augmenting the interaction with carers and patients

Summary

- Robotic software development poses some difficult problems for developers
- Augmented Reality techniques can improve the developers' perception of the robot and its environment, and improve the programming process

Game Engine-Based Virtual Surgery Simulation

Stefan Marks
Department of Computer Science
Division for Biomedical
Imaging and Visualization
University of Auckland
New Zealand
Email: dev.stefan.marks@gmx.net

John Windsor
Faculty of Medicine
and Health Sciences
Department of Surgery
Advanced Clinical Skills Center
University of Auckland
New Zealand
Email: j.windsor@auckland.ac.nz

Burkhard Wünsche
Department of Computer Science
Division for Biomedical
Imaging and Visualization
University of Auckland, New Zealand
and School of EECS
Kyungpook National University
Daegu, South Korea
Email: burkhard@cs.auckland.ac.nz

***Abstract*—The increasing complexity and costs of clinical training and the constant development of new procedures has made virtual reality based training an essential tool in medical education. Unfortunately, commercial training tools are very expensive and have a small support base. Game engines offer unique advantages for the creation of highly interactive and collaborative environments.**

This paper summarises research on the suitability of currently available game engines for developing applications for clinical education and training. We formally evaluate a list of available game engines for stability, availability, the possibility of custom content creation, the interaction of multiple users via a network and capabilities for simulating soft tissue deformations. Based on these criteria, three of the highest ranked engines are used for further case studies.

We found that in general it is possible to easily create scenarios with custom medical models that can be collaboratively viewed and interacted with, though limitations in physical simulation capabilities make some engines less suitable for fully interactive applications. We show that overall game engines represent a good foundation for low cost clinical training applications and we discuss technologies which can be used to further extend their physical simulation capabilities.

I. Introduction

The rising complexity and costs of clinical training and the development of new procedures has increased the importance of clinical simulators for education and training purposes. *Surgical simulators* represent a major part of the large variety of applications. Most commercially available simulators cover the area of endoscopic respectively laparoscopic procedures (e.g. Procedius MIST [29], LapSim [34], LAP Mentor [33], VEST System One [31], LapVR [24], EndoTower [41]). This kind of procedure requires well developed skills of the surgeon with respect to coordination of the camera and the surgical instruments that are not in direct view and are represented only on a 2D screen, sometimes with changed orientation due to camera rotation. The above mentioned systems are all able to train basic procedures in camera and instrument handling, before training the medical and surgical aspects.

Nevertheless, those technical skills are not the only necessity for a surgeon. The AGCME Outcome project [1] lists six general competencies which include other skills like patient care, medical knowledge, the ability to continuously learn and improve by practice, interpersonal and communication skills, professionalism, and the awareness of the health care system with its resources and demands as a whole.

Most of the mentioned simulator systems only train the technical and procedural skills of a surgical procedure, but lack the other aspects of the above list. Few physical simulators with mannequins (e.g. MedSim-Eagle [11]) enable small groups of residents to practice the collaborative aspects (i.e. interpersonal and communication skills) of medical or surgical procedures, but are very cost-intensive and thus likely to be unaffordable for most institutions.

One factor that is responsible for high costs of surgical simulators is the fact that certain parts of them are repeatedly reinvented. All simulators need at least graphical output capable of displaying 3-dimensional models with a high level of realism and user interfaces for operating and configuration of the simulator, an underlying physical simulation model, and event handling for input devices (see Figure 1). Some simulators are capable of adding the audible aspect of a surgical procedure and thus need a module for sound generation.

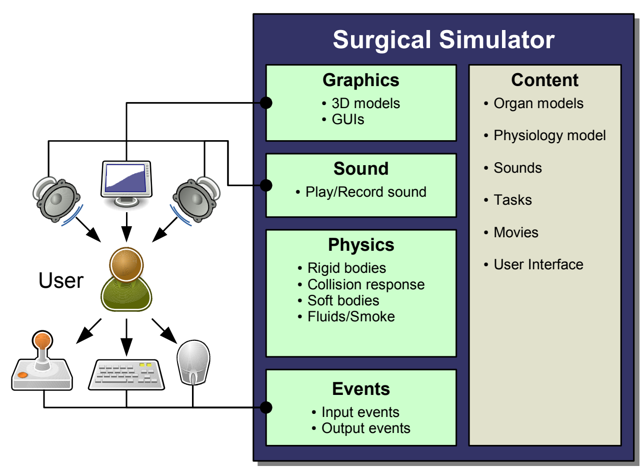


Figure 1. Functional blocks of a surgical simulator.

There have been attempts to create extensible frameworks for building surgical simulators upon (e.g. SPRING [30], GiPSi [10], SOFA [3]). They all incorporate the above mentioned modules and a variety of mathematical models for the physical simulation and interaction. But except for SPRING (ironically the oldest project in the list) they all lack the capability of networking with other simulators to build collaborative scenarios.

Teamwork is an overall important factor not only for surgeons but for all clinical personnel. Therefore, simulation systems that neglect the collaboration and communication of users can only deliver a part of the overall education.

This paper evaluates the potential of gaming engines for developing surgical simulators, particularly in relation to the above mentioned features and in their ability to support collaboration and communication between multiple users.

Section 2 summarizes the design and technical features of game engines relevant to our research. Section 3 introduces the methodology used to identify suitable game engines. Section 4 presents three promising game engines for surgical simulator development and discusses their advantages and disadvantages using simple case studies and proof-of-concept implementations. Section 5 and 6 investigate techniques for simulating soft tissue deformation and communication within a surgical team, respectively. Section 7 concludes our paper and section 8 presents future work.

II. Game Engines

The use of games or game engines for medical education has not been extensively explored with many aspects still to be investigated. One reason for this might be the lack of concordance between the seriousness of medicine and the playful, sometimes violent nature of computer games. Nevertheless, game engines offer a vast pool of useful concepts and resources in both technical and educational aspects.

Projects like the "Serious Game Initiative" focus on offering help to "organize and accelerate the adoption of computer games for a variety of challenges facing the world today." A subproject founded by this initiative is "Games for Health" [32], mainly focusing on games used in various health care sectors.

Previous authors have so far concentrated on applications where the game content was about learning facts, rather than tasks, procedures and teamwork. For example, Wünsche et al. [45] have examined how game engines can be used for visualising medical datasets, and Mackenzie et al. [27] utilise a game engine for anatomical education.

"Pulse!!" [36] is a recently developed, major project of the Texas A&M University utilising a game engine for teaching single users the procedures and systems of a health-care facility. Even more recently, the simulation "3DiTeams" [35] has extended this principle to include multiple users who collaborate in an emergency room setting over a network.

A. Game Engine Design

A game engine is a complex software system necessary for developing and playing games. Two different games with the same underlying engine differ by the *game content*, i.e. graphics, sounds, storyline. Game engines build a bridge between this content and the underlying hardware. With the help of an operating system abstraction layer, the same game content can be run on many platforms (e.g. Windows, Linux, XBox) without change.

Modern game engines consist of all or a subset of functional blocks depicted in Figure 2.

The *Graphics Engine* loads, displays, manipulates and manages all the data related to graphical content and visual effects. 3D models of objects, landscapes, buildings, objects, animals, and players can be loaded, textured, lit, and animated. Additional effects (e.g. blurring, lens distortion, depth of field) can be added to enhance the visual realism. Particle systems are utilised to simulate fire, smoke, bubbles, blood, etc.

All audible content like sound effects, ambient noise, and music is handled by the *Audio Engine*. In connection with modern soundcards it is possible to simulate acoustic obstruction by objects, environments other than air, reverb, Doppler effect and the spatial position of sound sources.

The total memory usage of game content is often higher than the memory provided by the gaming platform. Because not all of this data is needed simultaneously, the *Memory Management* is responsible for purging unused content from memory and in turn providing and managing requested memory for new content. Modern engines parallelise tasks like graphics, sound, physics, and AI. To balance the workload of all these tasks efficiently, especially on multiprocessor platforms, the *Process Management* is utilised.

Another essential part is the *Event handling*. Input devices, like joysticks, mice, keyboards, and gamepads generate events as well as network, timers, other components of the gaming hardware, game scripts and many other sources. These events are handled, filtered and distributed by one central event loop.

Some media like music or video does not need to be loaded into memory before being played back, but can instead be streamed to save precious memory resources. The *Streaming* mechanism is also capable of loading resources via the network from other servers.

Realistic behaviour of game objects has become more and more important in the recent years. The *Physics Engine* implements advanced mathematical models for calculating rigid body simulations of arbitrarily shaped and articulated objects (e.g. vehicles, machines). With the development of highly sophisticated physics engines like Havok Physics™[22] or PhysX™[2], the simulation of soft bodies, cloth, fluids, and smoke has become possible, accelerated either by specialised hardware or the computational power of the graphics hardware [21].

Artificial intelligence, provided by the *AI Engine*, is needed for controlling Non-Player Characters (NPC), the computer controlled antagonists in games. NPCs have to make decisions about how to follow, avoid or attack the player, and how to react to aggressive or defensive actions in a realistic and effective manner. AI Engines incorporate Computer Science topics like neural networks, state machines, A* search, and much more.

The flexibility of game engines is their greatest strength in creating manifold content. This is achieved by *Scripting*

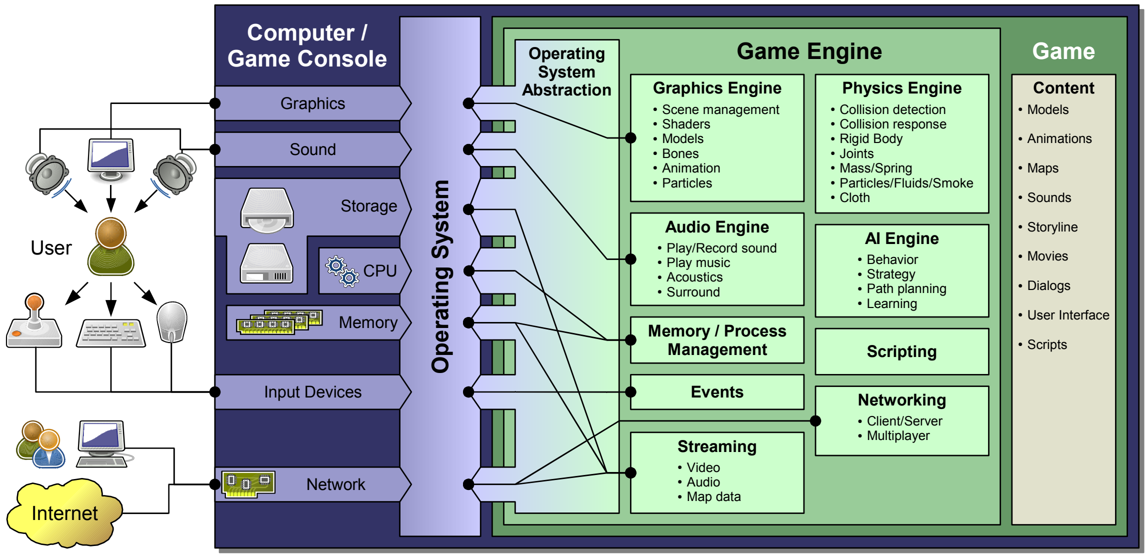


Figure 2. Functional Blocks of a Game Engine.

languages that allow an immediate access to the functions of the game engine. By scripts, the game content gets its typical "fingerprint", how a game is to be played and controlled, how the story develops, and how interactive and immersive the game environment appears.

The final important functional block of a game engine is *Networking*. By sending the state, movement and other information of each player and NPC over the network, other connected human players can collaborate in a game, because they all see the identical state of the game world at the same moment. The networking functions cope with network problems like packet loss or different runtimes of data packets from clients to servers and vice versa.

B. Technical Advantages

Since the game market is incredible competitive and incorporates both large established and innovative new vendors, game engines are constantly updated and utilise the latest graphics hardware and graphics algorithms. In addition, since most users are unable to constantly update their machines, game engines are designed for handling the same game content on hardware with different speed, memory size and features.

Playing computer games is no longer an action for individuals but has evolved into multiplayer gaming, bringing together several thousand players at the same time. Consequently, many game engines incorporate properly constructed and tested network support that serves well in connecting multiple users for collaboratively accomplishing tasks, even worldwide, unrestricted by location and physical boundaries. Built-in support for recording and playing sound over the network enables the players to communicate in a natural way to coordinate their actions. Textual input of messages serves as an alternative way.

Due to the fact that games are marketed internationally, game engines are able to deal with different languages. User interfaces, sound support and input devices can be customised accordingly.

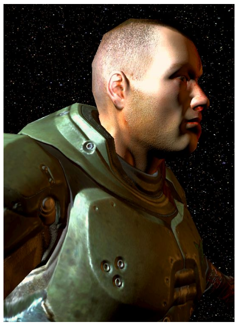
After games have been introduced to the market, they are constantly and intensively used by customers which results in massive feedback about errors and flaws. After some months and sometimes only weeks, patches are available to fix these issues. During this time, a large number of developers will have built up, who have gathered experience in modifying the game content. They form an international support community, often willing to help others when problems arise in building custom content for a game engine.

C. Important Features

One major important aspect for medical simulation is visual realism. With game engines supporting the most modern graphics hardware, this issue can easily be addressed. To enhance the realism in their hystheroscopy simulator, Bachofen et al. introduces bump mapping, spotlights, shadows, lens distortion, depth of field, bubbles, and floating tissue [7]. This list is only a subset of effects used in modern games, as the screenshots in Figure 3 illustrate.

The acoustic environment of a medical procedure is also an important part of a simulation [43] and can also be easily and accurately simulated with the features available in game engines. It may provide audible feedback of instruments used during medical procedures as well as reactions of the simulated patient, like pain or relief.

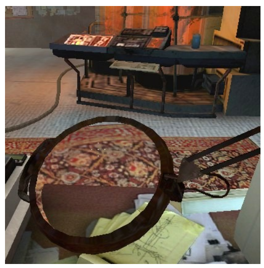
The physical simulation of objects in games is a relatively young area and thus does not yet cover the mathematically demanding aspects of soft tissue simulation (e.g. [14], [12], [25]) or cutting (e.g. [17]). This drawback can be compensated for either by playback of animations, skeletal animation, simple mass-spring systems or, if possible, extension of the physics engine. The manufacturers of physics engines are currently working on introducing new features like fluids and



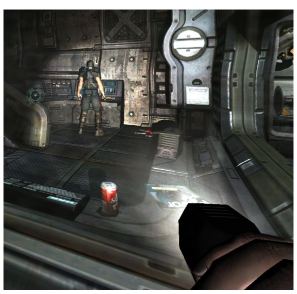
(a) Human skin

(b) Human tissue

(c) Reflection



(d) Distortion



(e) Spotlights

(f) User interfaces

Figure 3. Graphical capabilities of modern game engines. The screenshots are taken from the games Doom 3, Half-Life 2 and Quake 4.

soft body simulation, so the features missing right now may be available in the near future.

Networking (multiplayer) capabilities of games offer a great chance of building collaborative training scenarios. Players can see the position and state of their team members, and can communicate with each other by microphones and headphones or textual messages.

All of these features are useful for clinical simulations. Advanced graphics makes the simulation more realistic and limited hardware requirements allow users in development countries and smaller clinics to employ the software. The network support and GUI customisation can bring together multiple users for training of collaborative tasks, unrestricted by classroom walls and country boundaries and, when designed carefully, even independent of language barriers.

III. Methodology

A. Engine Selection

We started our selection of suitable game engines with an evaluation of an internet game engine database [15]. At the time of this evaluation (July 2007), this database contained 278 engines. We disregarded engines still in an early development state or those that were not developed or maintained any more. Engines without sound or other essential components were also removed from the list. Of the remaining engines, we selected those with in-built means of creating new game environments (maps). This last criterion is an important aspect for reducing the complexity of the editing process as there is no need for purchasing, installing and setting up external editors and necessary conversion tools, assumed the latter exist at all.

After the reduction of the original list by this selection process, we chose three inexpensive and in our opinion popular game engines for further evaluation.

- Unreal Engine 2 [18]
- id Tech 4 [44]
- Source Engine [38]

B. Evaluation

All engines were tested for their suitability for collaborative simulated surgical training applications by examining the following aspects:

Editing: Is everything that is necessary for creating and manipulating custom content included in the software? How is the editing process for a map started? Are the construction principles that are used during the process of building a map intuitive?

Content: How easy is the process of including game content as well as external models into the map? Which restrictions have to be considered when importing custom models?

Gameplay: How well can two or more users interact within the map and with the custom model? Are there any restrictions in the physical interaction?

Editing of models was performed with the 3D editor Blender [8]. This software is free, in contrast to commercial and expensive 3D editors such as 3D Studio Max [4] or Maya [5], and has import and export filters for all important 3D formats that were necessary for including custom models into the maps created for each game engine.

IV. Results

A. Unreal Engine 2

The following results are based on the game “Unreal Tournament 2004.”

Editing: The Unreal Engine 2 map editor “UnrealEd 3” is started as a standalone program. It incorporates model viewer, texture browser, script editor and other components necessary for a map (see Figure 6(a)). In contrast to the editors of the other two discussed engines, the editing process is of

(a) Unreal Engine 2 (b) id Tech 4 (c) Source Engine

Figure 4. Screenshots of our simulation scenarios implemented with different game engines.

subtractive nature. Volumes where players are supposed to move in have to be "carved out" from the originally solid game world.

An editing concept common to all editors of the three evaluated game engines is the "brush." It is used for selecting, for example, the areas that will be subtracted from the game world. But it can also be used for adding walls, spheres, stairs or other simple geometries.

Geometrically complex objects like shelves or engines are selected from a list, added into the map, and can then be moved, rotated, and changed in their behaviour or attributes. The same principle applies for physical objects like rigid bodies or joints.

Content: We constructed a room with a metal shelf (game content) and a custom skeleton model on a table (custom content). The skeleton[1] was split into parts (torso, skull, legs, and arms), which were then inserted as physical objects and connected by ball joints (see Figure 5). The file format for inserting custom models can be one of `.LWO` (Lightwave Object File), or `.ASE` (ASCII Scene Exporter). We used the latter due to having an `.ASE` export filter in Blender,

Gameplay: We started the map in multiplayer mode and interacted with the static and dynamic objects.

Figure 5. Asynchronous state of the skeleton on the server (left) and the client (right).

Non-physical actions and states are well synchronised between the server and the client. Player positions, orientations and states and also optical effects like decals (e.g. for scorch-marks) appear equally on both sides.

The articulated skeleton can be moved by applying forces. This works well in single player mode and on the server side in multiplayer mode. However, the multiplayer client shows unexpected behaviour. When force is applied, the graphical representation of the skeleton stays in place, whereas its physical representation moves (see Figure 5).

This asynchronism of the physics engine is not considered an error, as at 2003, the time of the release of the game, the physical simulation of game objects was not yet an important aspect of gameplay. Nevertheless, users wanted to create multiplayer maps with synchronised physical objects and thus developed a modification of the physics engine [46]. Due to the age of the Unreal Engine 2, this project has undergone no further improvement since 2005 and is now no longer available on servers.

B. id Tech 4

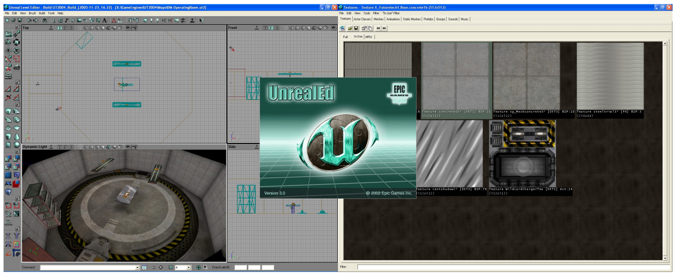
The following results are based on the game "Quake 4." This game uses a more recent version of the id Tech 4 engine than the game "Doom 3".

Editing: The id Tech 4 engine incorporates a set of editors necessary for building maps and inserting custom content (see Figure 6(b)). All of them can be started separately to edit, for example, maps, articulated figures, effects, materials, and scripts.

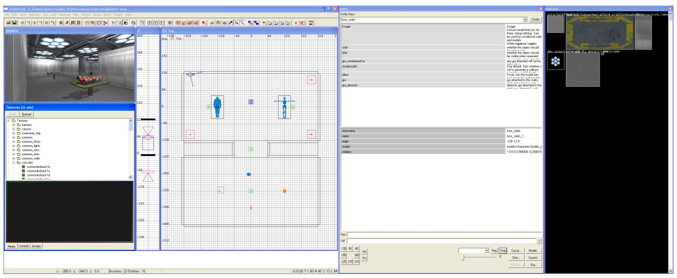
The map editor "Radiant" has a simple user interface, including a world view and a texture and model browser. Like the editors of the other two discussed engines, it also uses the brush concept for adding simple geometries and a selection list for more complex objects. In contrast to the other two editors which use four windows for the top, front, side, and 3D view of the scene, this editor is restricted to a single window with the top view in conjunction with a simplified tall window for adjusting the height of placed objects.

Content: With the map editor we created a simple room with two tables, on which we placed a game content model of a dissected body and a static custom content skeleton

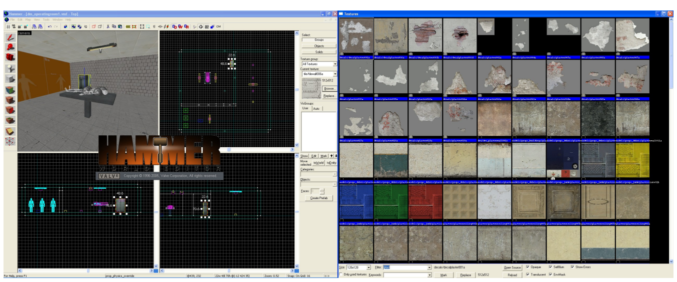
[1]Skeleton model source: http://artist-3d.com/free_3d_models/dnm/model_disp.php?uid=637



(a) "UnrealEd 3" (Unreal Engine 2)



(b) "Radiant" (id Tech 4)



(c) "Hammer" (Source Engine)

Figure 6. Screenshots of the map editors of the three evaluated game engines.

model, imported from an `.ASE` file (see Figure 4(b)). We placed additional objects (fire extinguisher, book, gas bottle) in the scene to evaluate collaborative interactions with physical objects.

When we tried to articulate the skeleton by connecting the limbs and skull to the torso, we discovered that the physical support is limited to simple rigid bodies. This limitation was unexpected, due to the fact that we also worked with the id Tech 4 engine based game "Doom 3." In this game, a movable crane with heavy, swinging load appears at least in one map. Its movements can be controlled by the user and the animation of the load is handled by the physics engine. Further investigation revealed that the physics engine used in the game "Doom 3" is part of the game content, but not of the basic id Tech 4 engine [23].

Gameplay: The map was loaded in multiplayer mode and entered by two users. Player positions, orientations and states as well as optical effects are synchronised well between server and client.

Physical objects can be manipulated by both, although the refresh rate of the position and orientation of physical objects on the client is slow and results in a jerking movement. This problem could not be solved by manipulations of the server settings. Additionally, some physical items also showed the asynchronous behaviour of their graphical and physical representations as for the Unreal Engine 2. It is yet unknown for which kind of objects this applies and if there are possible countermeasures.

C. Source Engine

The SDK of the Source Engine includes editors and helper programs and thus enables the construction of new maps and even modification of the source code of the engine. Permission to download it is obtained by purchasing a game of the "Half-Life 2" series.

Editing: Maps are created and modified with the map editor "Hammer" (see Figure 6(c)). The producer of the engine, Valve, also refers to the free XSI ModTool from Softimage [6], which, in conjunction with a special plug-in that is available on the website of Valve, can be used for creating static and animated models.

The editor can be switched into different modes, like constructing solid objects, placing complex objects, moving objects, and texturing them. Complex objects (entities) are not only geometrically complex models, but also physical objects, physical constraints, light sources, etc.

Content: We created a test room with some game content objects (e.g. locker, lamps) and a table with the custom content skeleton. In contrast to the Unreal Engine 2 and the id Tech 4 engine, a model imported into the Source Engine may only consist of a maximum of 32768 vertices. This also limits the number of triangles to a maximum of about 11000. These limits are coded into the engine and may not be changed. For performance reasons, Valve suggests to reduce the complexity of models to below 10000 triangles [39]. We discovered that this limitation can be overcome by splitting the model into parts that are assembled into one object when converting the model data into the game engine's internal format.

The conversion programs for models and textures are purely command line based. To convert a single model into the engine format, one has to drag the compilation file onto the converter in the explorer view, or start the process by entering a command line instruction. These command line based programs could be utilised easily to automate a complex process of creating maps and models from medical datasets.

Gameplay: Compared to the other two engines, the Source Engine performed best. The position, orientation and state of the users character as well as the physical simulation synchronised well and fluently on server and client (see Figures 7 and 4(c)).

V. Soft Tissue Simulation

Soft tissue simulation is an important technique for simulating surgical procedures. Existing solutions in the field

Figure 7. Screenshots of an interactive simulation scenario implemented with the Half-Life 2 engine. The user can interactively bend the vessels of the heart model with a tool.

of biomedical modelling can be divided into pre-animations, kinematic methods, geometric methods, and physically-based methods. Currently available game engines offer only very limited support for these techniques.

Pre-animated simulations are provided for by most game engines and are achieved by modelling a limited range of interactions using a human animator or more complex physically-based techniques. The simulations are stored in movie or 3D animation formats and are triggered when the user performs certain predefined operations such as cutting in a specified region. This type of simulation does not allow arbitrary interactions, but is fast and can be achieved using game engines, flash animations and other widely available tools. Such simple simulate-ions can be useful in virtual surgery tools teaching process rather than motor skills [16].

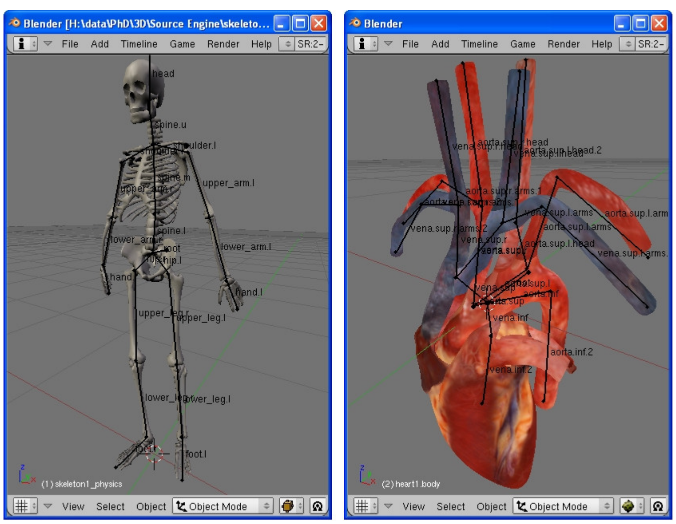


Figure 8. The animation skeleton and surface mesh of a rigid object (skeleton) and a deformable object (heart).

Physically-based methods implemented within game engines are usually restricted to rigid-body physics. A limited simulation of soft objects is possible by using skinning where objects are defined by a jointed skeleton and a surface mesh. The mesh's vertices are defined as coordinates of one or several bones of the skeleton. Hence if the skeleton moves, the surface mesh, or *skin* moves with it. Figure 8 shows a rigid and a soft object defined by an animation skeleton and a surface mesh. For a rigid object every surface point is defined with respect to exactly one skeleton bone, whereas for a soft object surface mesh coordinates are weighted averages of bone coordinates. As a result the mesh deforms smoothly if the joint angle between two bones changes.

From the engines we investigated in detail the source engine was the only one offering mesh-spring systems. We created an object consisting of 4×4 partially intersecting spheres connected in a rectangular pattern by 24 springs (see figure 9) into the test map. This physical construct can be grabbed by two or more users with a "Gravity Gun" and being pushed and pulled (see figure 10). The simulation of the mass spring system performed stable and well synchronised on server and client. With the custom texture and the shiny surface properties, it resembled the handling of a colon or a similar structure.

Figure 9. Multiplayer test map for the Source Engine.

Figure 10. Collaborative manipulation of the mass-spring system.

VI. Simulation of Inter-Team Communications

An important aspect neglected by many surgical simulators is that surgeries are performed by a team. Synchronisation and efficient communication of team members is crucial and should be incorporated into the training. Lingard et al. examined 90 hours of 48 recorded surgical procedures, classified communication into 421 events and found 129 communication errors. From these errors 36.4% lead to immediate effects such as team tension, delays and ineffciency [26]. Several authors have investigated the communication within surgical teams and it has been found that all channels of communication (e.g. voice, gesture, gaze, orientation) and types (e.g. command, offer, question, statement) are important and not observing once channel can lead to misunderstandings [42].

We investigated common interactions within games and virtual environments and identified the following ones as important: Spatial behaviour (position of team members to each other and other objects), changes in avatar appearance (stains on clothes, adornment, equipment), Language based communication (speech, chat, phrases, commands), nonverbal audio (e.g. signals of monitoring equipment), kinesics (posture, head and body movement), facial expressions, and occluesics (eye movement and eye contact [20]).

The Source Engines provides limited control over head movements, e.g. the head follows mouse movement and in multiplayer mode players turn the heads to each other when they come close. Note, however, that this feature is undesirable in our research and should be turned off. In addition the Source engine (and soon to be released new game engines) has good support for facial animation and control of eye movement. There exists a tool for synchronising spoken text of in-game characters to their facial and body expressions.

VII. Conclusion

Modern game engines contain many features that would be necessary for building a clinical training application. Graphics, audio and network capabilities are highly developed and allow the creator of applications to focus on content rather than details of the implementation. The underlying hardware is optimally used. Multiuser interaction is possible by multiplayer scenarios and allows the training of teamwork and cooperation.

In contrast, highly mathematical physics models for simulation of soft tissue are (not yet) possible with game engines. Basic soft tissue interaction can therefore either be simulated by the use of simple mass-spring models [28] or by the extension of the physics of a game engine to mathematically more sophisticated models, if the engine allows for these changes (e.g. Source Engine).

Predefined file formats can pose difficulties when converting medical images and models. These formats may be limited in their number of vertices or faces and thus would need preprocessing to reduce the amount of information without loss of optical detail. Another possibility for overcoming these limits is to split complex objects into parts that are kept together (e.g. by constraints [45]). On the positive side, the necessary file formats for the examined three engines are well documented (`.ASE`: [37], `.SMD`: [40]).

An interesting aspect of the Source Engine is the fact that not only the material and model compilation files are text based, but also the file format for maps. In conjunction with the command line based tools, this could lead to the development of a fully automated tool that reads patient related medical data and constructs a map including the custom patient models for interaction and training.

VIII. Future Work

Game engines allow for a relatively quick creation of interactive scenarios for simulated clinical training. Nevertheless, there are still restrictions in the physical modelling of soft tissue, which prevent their use for training surgeons in the area of technical skills, e.g. laparoscopic procedures. The true strength of game engines lies in their networking capabilities. The ability to interactively bring together several users in a simulation scenario is ideal for the training and assessment of *teamwork*, which is an aspect of healthcare that has often been overlooked, but is now beginning to receive due attention [9].

Our future work will emphasise the development of training frameworks, scenarios and assessment methods that are appropriate for the evaluations and training of teams in clinical settings. The existent physical simulation capabilities of game engines could be utilised to enhance the realism of these training scenarios, e.g moving instrument trays, cables connected to the patient. Tools for automatically converting real patient data to game engine specific files (e.g. based on the Source Engine command line tools) could be used to create a variety of training scenarios which are based on real-life cases.

We will also continue to monitor the latest development of new game engines, e.g. CryEngine 2 [13] and Unreal Engine 3 [19]. Among their features are enhanced graphics and physical modelling techniques. With these it is possible to blend animations while maintaining a set of constraints. Geometric models can be deformed arbitrarily by displacement textures. Physically correct simulation of smoke and liquids can also be used. These features will allow even more realistic simulations and will be subject to further analysis.

References

[1] ACGME. ACGME Outcome Project – General Competencies [online]. Sept. 1999 [cited 17.08.2007]. Available from: http://www.acgme.org/outcome/comp/compMin.asp.

[2] Ageia Technologies. Ageia [online]. 2007 [cited 21.06.2007]. Available from: http://www.ageia.com/.

[3] J. Allard, S. Cotin, F. Faure, P.-J. Bensoussan, F. Poyer, C. Duriez, H. Delingette, and L. Grisoni. SOFA – an Open Source Framework for Medical Simulation. *Medicine Meets Virtual Reality (MMVR 15)*, 125:13–18, February 2007.

[4] Autodesk. Autodesk 3ds Max [online]. 2007 [cited 16.08.2007]. Available from: http://www.autodesk.com/3dsmax.

[5] Autodesk. Autodesk Maya [online]. 2007 [cited 16.08.2007]. Available from: http://www.autodesk.com/maya.

[6] Avid Technology. Welcome to Softimage – 3D Software Solutions for Games, Films, and Television Artists [online]. 2007 [cited 17.08.2007]. Available from: http://www.softimage.com/.

[7] D. Bachofen, J. Zátonyi, M. Harders, G. Székely, P. Früh, and M. Thaler. Enhancing the Visual Realism of Hysteroscopy Simulation. *Studies in Health Technology and Informatics*, 119:31–36, Jan. 2005.

[8] Blender Foundation. Blender [online]. 2007 [cited 17.08.2007]. Available from: http://www.blender.org/.

[9] C. S. Borrill, J. Carletta, A. J. Carter, J. F. Dawson, S. Garrod, A. Rees, A. Richards, D. Shapiro, and M. A. West. The Effectiveness of Health Care Teams in the National Health Service. Technical report, Department of Health, 2001. Available from: http://homepages.inf.ed.ac.uk/jeanc/DOH-final-report.pdf [cited 18.02.2008].

[10] M. C. Çavuşoğlu, T. G. Göktekin, and F. Tendick. GiPSi: A Framework for Open Source/Open Architecture Software Development for Organ Level Surgical Simulation. *IEEE Transactions on Information Technology in Biomedicine*, 10(2):312–322, Apr. 2006.

[11] CISL. The MedSim-Eagle Patient Simulator [online]. 2007 [cited 17.08.2007]. Available from: http://med.stanford.edu/VAsimulator/medsim.html.

[12] S. Cotin, H. Delingette, and N. Ayache. Real-Time Elastic Deformations of Soft Tissues for Surgery Simulation. *IEEE Transactions on Visualization and Computer Graphics*, 5(1):62–73, Mar. 1999.

[13] Crytek. CryEngine 2 Specifications [online]. 2002 [cited 17.08.2007]. Available from: http://www.crytek.com/technology/index.php?sx=eng2.

[14] H. Delingette. Toward Realistic Soft-Tissue Modeling in Medical Simulation. *Proceedings of the IEEE*, 86(3):512–523, Mar. 1998.

[15] DevMaster.net. 3D Game Engines Database [online]. 2007 [cited 17.08.2007]. Available from: http://www.devmaster.net/engines/.

[16] B. Dunkin, G. L. Adrales, K. Apelgren, and J. D. Mellinger. Surgical simulation: A current review. *Surgical Endoscopy*, 21:357–366, Mar. 2007.

[17] J. Dworzak and L. Gu. Combining progressive and non-progressive cutting for soft tissue surgery simulations. *International Journal of Computer Assisted Radiology and Surgery*, 2(Suppl 1):S163–S165, June 2007.

[18] Epic Games. Unreal Engine 2 [online]. 2004 [cited 17.08.2007]. Available from: http://www.unrealtechnology.com/html/technology/ue2.shtml.

[19] Epic Games. Unreal Engine 3 [online]. 2006 [cited 17.08.2007]. Available from: http://www.unrealtechnology.com/html/technology/ue30.shtml.

[20] M. Garau, M. Slater, S. Bee, and M. A. Sasse. The impact of eye gaze on communication using humanoid avatars. In *CHI '01: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 309–316, New York, NY, USA, 2001. ACM.

[21] D. Geer. Vendors Upgrade Their Physics Processing to Improve Gaming. *Computer*, 39(8):22–24, Aug. 2006.

[22] Havok. Havok [online]. 2007 [cited 17.08.2007]. Available from: http://www.havok.com/.

[23] id Software. id.sdk [The Code] [online]. 2007 [cited 17.08.2007]. Available from: http://www.iddevnet.com/doom3/code.php.

[24] Immersion Corporation. Immersion Medical [online]. 2007 [cited 17.08.2007]. Available from: http://www.immersion.com/medical/.

[25] Y.-J. Lim and S. De. Real time simulation of nonlinear tissue response in virtual surgery using the point collocation-based method of finite spheres. *Computer Methods in Applied Mechanics and Engineering*, 196(31-32):3011–3024, June 2007.

[26] L. Lingard, S. Espin, S. Whyte, G. Regehr, G. R. Baker, R. Reznick, J. Bohnen, B. Orser, D. Doran, and E. Grober. Communication failures in the operating room: An observational classification of recurrent types and effects. *Quality & Safety in Health Care*, 13(5):330–334, Mar. 2004.

[27] J. Mackenzie, G. Baily, M. Nitsche, and J. Rashbass. Gaming Technologies for Anatomy Education. Online, May 2003. Available from: http://www.virtools.com/news/pdf/2004/CARET.pdf [cited 17.08.2007].

[28] S. Marks, J. Windsor, and B. Wünsche. Collaborative Soft Object Manipulation for Game Engine-Based Virtual Reality Surgery Simulators. In M. J. Cree, editor, *Proceedings of Image and Vision Computing New Zealand*, page 205–210, Hamilton, New Zealand, Dec. 2007.

[29] Mentice. Mentice [online]. 2007 [cited 17.08.2007]. Available from: http://www.mentice.com/.

[30] K. Montgomery, C. Bruyns, J. Brown, S. Sorkin, F. Mazzella, G. Thonier, A. Tellier, B. Lerman, and A. Menon. Spring: A General Framework for Collaborative, Real-time Surgical Simulation. *Studies in Health Technology and Informatics*, 85:296–303, 2002.

[31] select IT VEST Systems AG. Select-IT VEST Systems AG – medical science at your fingertips [online]. 2007 [cited 17.08.2007]. Available from: http://www.select-it.de/.

[32] Serious Games Initiative. Games For Health [online]. 2007 [cited 17.08.2007]. Available from: http://www.gamesforhealth.org.

[33] Simbionix. Simbionix, medical training simulators and clinical devices for MIS (minimally invasive surgery) [online]. 2007 [cited 17.08.2007]. Available from: http://www.simbionix.com/.

[34] Surgical Science. Surgical Science - Safer surgeons faster [online]. 2007 [cited 17.08.2007]. Available from: http://www.surgical-science.com/.

[35] J. Taekman, N. Segall, E. Hobbs, and M. Wright. 3DiTeams – Healthcare Team Training in a Virtual Environment. *Anesthesiology*, 107(A2145), Oct. 2007.

[36] Texas A&M University Corpus Christi. Pulse!! – The Virtual Clinical Learning Lab [online]. 2007 [cited 05.10.2007]. Available from: http://www.sp.tamucc.edu/pulse/home.asp.

[37] UnrealWiki. UnrealWiki: ASE File Format [online]. 2007 [cited 17.08.2007]. Available from: http://www.unrealwiki.com/wiki/ASE_File_Format.

[38] Valve Corporation. Valve Source Engine Features [online]. 2004 [cited 17.08.2007]. Available from: http://www.valvesoftware.com/sourcelicense/enginefeatures.htm.

[39] Valve Developer Community. Compiling Models [online]. 2007 [cited 17.08.2007]. Available from: http://developer.valvesoftware.com/wiki/Compiling_Models_Basics.

[40] Valve Developer Community. SMD file format [online]. 2007 [cited 17.08.2007]. Available from: http://developer.valvesoftware.com/wiki/SMD_file_format.

[41] Verefi Technologies. Verefi Technologies, Inc. [online]. 2007 [cited 17.08.2007]. Available from: http://www.verefi.com/.

[42] K. Vozella. Communicating in surgery. *Macquarie University News*, pages 8–9, May 2006. Available from: http://www.pr.mq.edu.au/macnews/MUN_PDFS/200605.pdf [cited 22.02.2008].

[43] J. D. Westwood, R. S. Haluck, H. M. Hoffman, G. T. Mogel, R. Phillips, R. A. Robb, and K. G. Vosburgh. Highly-Realistic, Immersive Training Environment for Hysteroscopy. *Studies in Health Technology and Informatics*, 119:176–181, Jan. 2005.

[44] Wikipedia. id Tech 4 — Wikipedia, The Free Encyclopedia [online]. 2007 [cited 17.08.2007]. Available from: http://en.wikipedia.org/wiki/Doom_3_engine.

[45] B. C. Wünsche, B. Kot, A. Gits, R. Amor, J. Hosking, and J. Grundy. A Framework for Game Engine Based Visualisations. In *Proceedings of Image and Vision Computing New Zealand 2005*, Nov. 2005. Available from: http://www.cs.auckland.ac.nz/~burkhard/Publications/IVCNZ05_WuenscheKotEtAl.pdf.

[46] J. Zepp. GoodKarma Physics Mod Beta 4 [online]. 2005 [cited 16.08.2007]. Available from: http://www.ataricommunity.com/forums/showthread.php?t=440477.

Interactive Paper Animation with Breakable Hinge Springs

Young-Min Kang
Department of Game Engineering
Tongmyong University
Email: ymkang@tu.ac.kr

Hwan-Gue Cho
Department of Computer Engineering
Pusan National University
Email: hgcho@pusan.ac.kr

***Abstract*—There have been little efforts to animate paper objects in interactive or real-time VR environments. Paper can be represented with the well-known mass-spring model, whose animation is relatively easy compared with rigid body models or fluid simulation. However, animating paper with the straightforward mass-spring model results in a serious numerical problem. The properties of plain paper require the representing mass-spring model to be extremely stiff, and the stiffness causes numerical instability. This numerical instability has been one of major obstacles to the real-time cloth animation, and many approaches have been proposed for efficient and plausible simulation of fabric material such as cloth. However, paper is far stiffer than cloth and in addition often fragile. The paper does not recover its original shape when it is deformed with strong force. The traditional mass-spring system cannot represent this damage preserving properties. In this paper, we propose a simple and intuitive approach to animate the damage-preserving stiff mass-spring model in order to represent the behavior of paper in virtual environments. The proposed method uses an approximate implicit method to guarantee the stability of the system, and employs breakable hinge springs to show damage-preserving properties. Fractal textures represent the small scale wrinkled appearance of old paper. The proposed method can be successfully utilized in real-time or interactive games or VR environments for paper animation.**

***Index Terms*—Paper Animation, Stiff ODE, Paper Folding, Breakable Hinge Spring**

I. Introduction

Although paper is a common object in everyday life, there have been little efforts to animate paper objects in interactive or real-time VR environments. Recently computational geometry has shown some interests in the combinatorial property of folded paper in the computational origami[4]. The dynamic behavior of paper has been modeled by Chu et al. in order to simulate page turns for virtual books[2]. The authors use a mass-spring structure defined on a rectangular grid of particles. These are connected with primary mesh springs, diagonal bracing, and four-way flexural rigidity springs. Many of the springs are very stiff, leading to a stiff system of differential equations. The resulting system of differential equations is solved with a Runge-Kutta method which can take several minutes for a single page-turn, precluding real-time operation. Other problems are that the mesh is static, has a low resolution and can only be bend smoothly and with a low curvature. Hence it's not possible to model other interactions with the paper such as folding, ripping or crumpling it. Also area preservation is not ensured.

In order to represent the behavior of a paper object, the well-known mass-spring model can be easily employed. The mass-spring model is the most intuitive and simple representation of deformable soft objects. In the computer graphics society, the animation of soft objects has been intensively studied for a couple of decades. The animation of soft objects modeled with mass-spring systems is far more time-consuming than that of a rigid object because of numerical instability. One of the most common applications of mass-spring based soft object models is the efficient simulation of cloth, which is represented with stiff springs connecting mass-points.

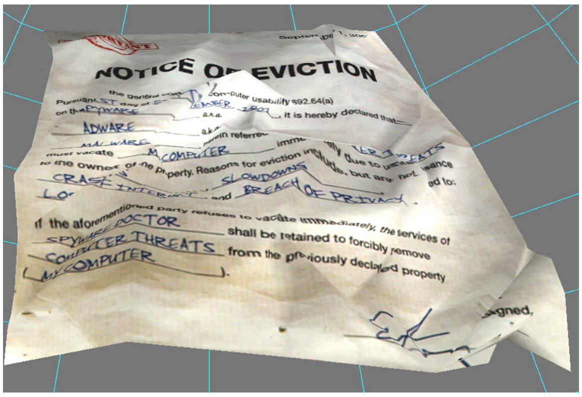


Fig. 1. Virtual paper constructed by this paper for the interactive environments.

However, animating paper with mass-spring model leads to a serious numerical problem. The properties of paper require the representing mass-spring model to be extremely stiff, and the stiffness causes numerical instability.

In the early stage of deformable object animation, geometric approaches have been widely employed because physical-based models require too heavy computation[7]. Terzopoulos *et al.* characterized the soft object animation as a deformable surface problem [8], and their work formed the foundation for the subsequent physics-based approaches.

Physical-based modeling approaches to soft object animation suffered from numerical instability which was a major obstacle to the real-time animation of stiff deformable objects such as cloth and paper. However, the use of implicit methods has resulted in many approximate approaches for the efficient animation of virtual cloth. Paper is far stiffer than cloth and also more also fragile. The paper does not recover its original shape when it is deformed with strong force. The traditional mass-spring system cannot represent this damage preserving properties. In this paper, we propose a simple and intuitive approach to animate the damage-preserving stiff mass-spring model in order to represent the behavior of paper in virtual environments.

II. Problem and Previous Work

The interactive animation of paper can be described as an efficient numerical integration of motion equation for the damage preserving stiff mass-spring model. This problem is composed of two main issues: 1) stable integration of a stiff ODE (ordinary differential equation), and 2) mesh management for damage preservation.

When modeled with a mass-spring model, paper is extremely stiff because paper hardly expands or contracts. Therefore, the animation of paper can be reduced to a numerical integration of extremely stiff ordinary differential equations. In order to animate the paper in virtual world, stable and efficient integration method has to be employed.

Although the stability is an essential issue for paper simulation, stable animation does not guarantee the realism of paper animation. If a paper object is animated with a stable mass-spring simulation method, the resulting animation would look like rubber animation rather than stiff but fragile paper. In order to increase the realism of paper animation, a damage preserving property must be taken into account.

Since Terzopoulos *et al.* represented soft objects as deformable surfaces [8], various physically-based approaches have been proposed for soft object animation. Thalmann's group has proposed many techniques for dressed virtual characters [9]. However, their major interest was the realism of the virtual cloth instead of computational efficiency. Therefore, most of their work employed explicit integration and interactive or real-time animation was impossible.

Various techniques have been proposed for efficient animation of soft objects, and one of the most important advances is the use of implicit integration which guarantees the stability of the animation [1]. However, the implicit integration involves large linear system [3]. Therefore, some efficient approximate approaches have been proposed [6], and a stable method that approximates the solution with a direct update formula was also introduced [5].

III. Interactive Paper Animation Techniques

In this section, techniques for interactive animation of paper objects will be introduced. The techniques are composed of three different approaches for stable animation, damage preservation, and surface details.

In order to obtain real-time or interactive performance, we employed an approximate implicit integration. The method provides stable and efficient computation of the next state of stiff mass-spring model. Our breakable hinge spring model is an approach to the representation of damage preserving paper objects. The model adaptively adjusts the mesh structure of the mass-spring model in order to preserve the damage and express various creases on the paper. The last technique is to generate the surface detail with a fractal surface texture.

A. Stable Animation of Extremely Stiff Springs

The stability is essential for the real-time or interactive animation of mass-spring model, and implicit integration is the common solution to the stability problem. The simplest implicit method is the backward Euler method, which can be described as follows:

$$\begin{pmatrix} \mathbf{v}^{t+h} \\ \mathbf{x}^{t+h} \end{pmatrix} = \begin{pmatrix} \mathbf{v}^{t} + h\mathbf{M}^{-1}\mathbf{f}^{t+h} \\ \mathbf{x}^{t} + h\mathbf{v}^{t+h} \end{pmatrix} \tag{1}$$

where h denotes the time interval, $\mathbf{v}$ the vector of velocity values of the mass-points, $\mathbf{f}$ the vector of forces, $\mathbf{x}$ the vector of locations of the mass-points, and $\mathbf{M}$ the mass matrix.

Although the implicit method described in Eq.1 seems simple, real-time animation is not an easy problem. The difficulty arises from the unknown force vector at time $t+h$. Since Hooke's law can compute only $\mathbf{f}^t$, the force at the next time step should be approximated by applying Taylor expansion as follows:

$$\mathbf{f}^{t+h} = \mathbf{f}^t + \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\Delta\mathbf{x}^{t+h} = \mathbf{f}^t + \mathbf{J}\Delta\mathbf{x}^{t+h} \tag{2}$$

where $\mathbf{J}$ denotes the Jacobian matrix of the force vector with respect to the position vector.

By using the force shown in Eq.2, an animation step of the mass spring model can be represented by a typical $\mathbf{A}x = b$ linear system:

$$(\mathbf{M} - h^2\mathbf{J})\Delta\mathbf{v}^{t+h} = h\mathbf{f}^t + h^2\mathbf{J}\mathbf{v}^t \tag{3}$$

For simplicity, let $\mathbf{W}$ denote the matrix $(\mathbf{M} - h^2\mathbf{J})$. The right side of Eq.3 is the sum of a spring force and viscosity force and can be easily computed without any computational problem. Let $\tilde{\mathbf{f}}_i^t$ denote the sum of all spring forces and viscosity force exerted on the mass-point i, and $\mathbf{f}^t$ be the vector of the forces $[\tilde{\mathbf{f}}_1^t, \tilde{\mathbf{f}}_1^t, \cdots, \tilde{\mathbf{f}}_n^t]^{\mathbf{T}}$. Then the problem shown in Eq.3 is efficiently represented as follows:

$$\mathbf{W}\Delta\mathbf{v}^{t+h} = h\tilde{\mathbf{f}}^t \tag{4}$$

By taking into account the property of the matrix $\mathbf{W}$, the velocity change of each mass-point can be described as follows:

$$\Delta\mathbf{v}_i^{t+h} \quad = \quad \mathbf{W}_{ii}^{-1}(h\tilde{\mathbf{f}}_i^t + h^2 \sum_{(i,j)\in E} \mathbf{J}_{ij}\Delta\mathbf{v}_j^{t+h}) \tag{5}$$

The update formula for mass-point i shown in Eq.5 depends on the same formula for mass-point j. Therefore, we cannot directly update the velocity change of each mass-point. Our method computes the approximate solution with an iterative update based on Jacobi iteration as follows:

$$\begin{aligned} \Delta\mathbf{v}_i^{t+h^{(0)}} &= \mathbf{W}_{ii}^{-1}h\tilde{\mathbf{f}}_i^t \\ \Delta\mathbf{v}_i^{t+h^{(k+1)}} &= \mathbf{W}_{ii}^{-1}(h\tilde{\mathbf{f}}_i^t + h^2 \sum_{(i,j)\in E} \mathbf{J}_{ij}\Delta\mathbf{v}_j^{t+h^{(k)}}) \end{aligned} \tag{6}$$

where $\Delta\mathbf{v}_i^{(k)}$ denotes the velocity change computed with k iterations.

Fig.2 shows the result of the stable integration described in this paper. Since the mass-spring is sufficiently stiff, the model does not stretch or contract severely. However, the resulting animation does not look realistic because no damage is preserved on the surface. In order to effectively represent a paper object, we must also simulate the damages which are not easily recovered by the stiff spring edges.

B. Breakable Hinge Springs

Although the stable integration method described in the previous subsection enables real-time animation of a stiff mass-spring model, the resulting images or animations produced with the method are far from the real appearance and behavior of paper.

The unsatisfactory result is caused by the static structure of the mass-spring mesh. When real paper is deformed by external forces, the paper is easily damaged and the damage is preserved on the surface. However, the stiff but static mass-spring mesh cannot represent such characteristic.

In order to represent the damage preserving property of paper object, we propose a breakable hinge spring model. The breakable hinge spring model is an adaptive mass-spring mesh structure. In this model, each spring can be broken when its length is shortened by external or internal forces. When a spring is broken, the mesh structure of the paper model should also be modified in order to maintain the mass-spring mesh to be a triangular mesh.

Fig.3 illustrates the breakable spring model. On the left side of the figure, a spring edge between two triangles is compressed. The compressed spring is broken and the adjusted mesh structure is shown on the right side. The compressed spring is divided into two distinct spring edges in order to maintain the original length of the compressed spring. The broken spring edge causes a problem in the mesh structure. Because of the broken edge, the mesh is not anymore a triangular mesh. In order to fix the problem, two additional springs

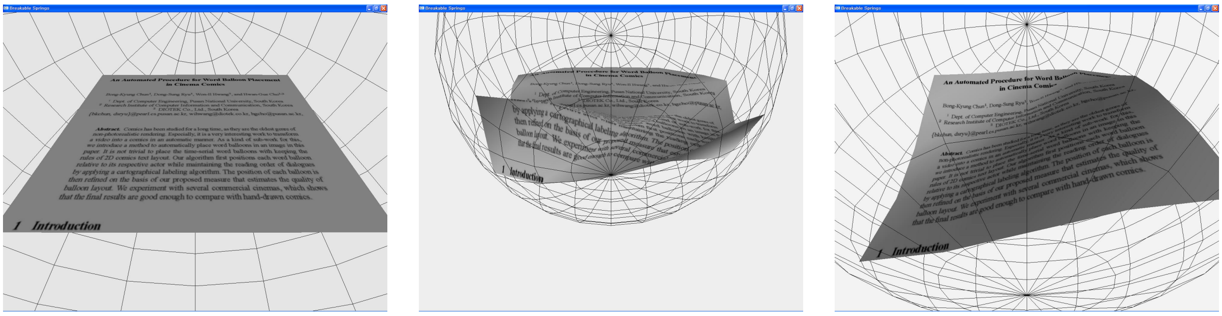


Fig. 2. Stable simulation of a stiff mass-spring model without breakable springs

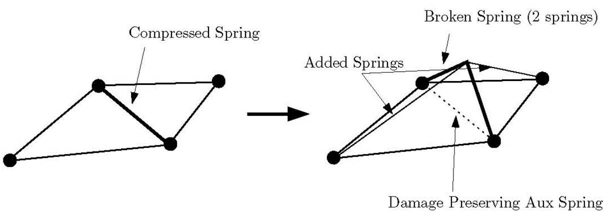


Fig. 3. Breakable spring and mesh adjustment

are used. After adjusting the mesh structure, the breakable spring model inserts an auxiliary spring to preserve the damage.

In the Fig.4, the auxiliary spring is represented with a dotted line. Because of the auxiliary spring, the mesh remembers the damage and shows the paper-like wrinkles. Fig.4 shows the mesh appearance when springs are randomly chosen and broken. The thick blue lines are breakable springs, and thin red lines are auxiliary spring edges which are not breakable. As shown in the figure the breakable spring model can efficiently represent the damages on the paper surface.

(a)

(b)

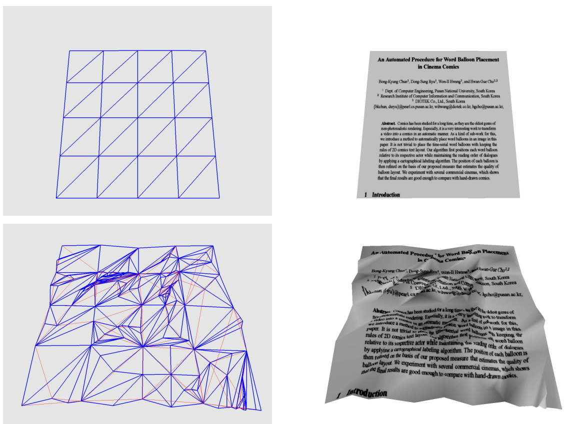


Fig. 4. Geometric appearance with breakable springs: (a) original mesh, (b) result when springs are randomly broken

Although the breakable spring model produces paper-like appearance, the method has a severe disadvantage: breakable springs can generate wrinkles only across the existing spring edges. Therefore, it is impossible to preserve the damage when the paper is folded along the existing springs. In order to represent the wrinkles along the spring edges, the springs also have to work like hinge joints. The hinge folding of a spring edge is implemented by applying an auxiliary spring across the spring edge as shown in Fig.5. In the figure $\mathbf{x}_i$ and $\mathbf{x}_j$ denote the mass-points linked with a spring, and $\mathbf{x}_r$ and $\mathbf{x}_l$ are the mass-points on the right and the left of the spring respectively. If the distance between $\mathbf{x}_r$ and $\mathbf{x}_l$ is shorter than a specified threshold, an auxiliary spring between the mass-points is inserted to the mass-spring structure.

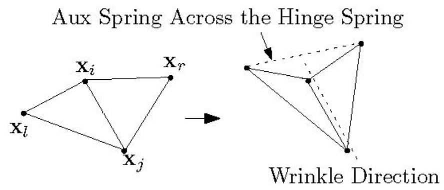


Fig. 5. Auxiliary spring across a hinge folding spring

Fig.5 shows the result when auxiliary springs for hinge springs have been inserted. It can be seen that the hinge springs make the surface of the deformable object wavy.

The method proposed in this paper utilizes both the breakable springs and hinge springs. Each spring in the mesh is either a breakable hinge spring or an unbreakable auxiliary spring. The result by randomly breaking or folding the spring edge is shown in Fig.7. As shown in the figure, breakable hinge springs make the mesh structure look like paper.

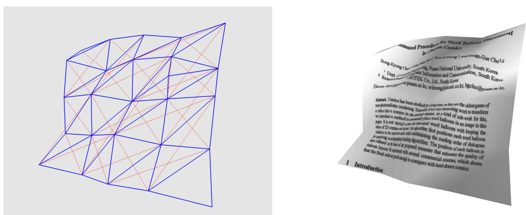


Fig. 6. Geometric appearance with hinge springs

C. Adaptive Mesh Structure

Although the mesh with randomly broken hinge springs displays plausible paper appearance, the random approach cannot be applied to paper animation. In the animation sequence, the paper model should be damaged by the forces.

In order to produce a plausible animation sequence, an automatic edge selection strategy for breaking or folding springs is necessary. Let $\mathbf{x}_i$ and $\mathbf{x}_j$ be the mass-points linked with a breakable spring. The

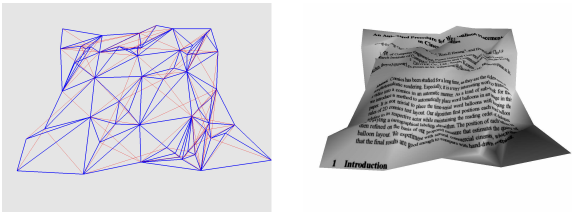


Fig. 7. Auxiliary spring across a hinge folding spring

locations of the mass-points at the original rest state are denoted as $\mathbf{x}_i^0$ and $\mathbf{x}_j^0$, and the locations at the current time t are $\mathbf{x}_i^t$ and $\mathbf{x}_j^t$. The rest length of the spring $l_{i,j}^0$ is, therefore, $|\mathbf{x}_i^0 - \mathbf{x}_j^0|$. The current length of the spring $l_{i,j}^t$ is, similarly, $|\mathbf{x}_i^t - \mathbf{x}_j^t|$. Because the spring is broken when it is compressed, the ratio $l_{i,j}^t / l_{i,j}^0$ is the most important value for determining whether the spring will be broken or not.

A simple approach is to break the spring when the ratio $l_{i,j}^t / l_{i,j}^0$ is less then a threshold. However, the damage on the paper surface actually occurs stochastically. Therefore, we used the ratio as the parameter for computing the probability of the catastrophic phenomena in the mesh structure. In other words, the ratio ranges [1,0] for a compressed spring, and the probability of breaking the spring $\mathbf{P}_{i,j}$ is computed as follows:

$$\mathbf{P}_{i,j} = (1 - l_{i,j}^t / l_{i,j}^0)^{\phi} \tag{7}$$

where ϕ is a parameter that controls the fragility of the springs.

When a spring is broken, a new mass-point must be inserted. The location of the mass-point must be determined. An easy method is to set the location of the new mass-point as the middle of the two mass-point linked with the original spring. However, this simple method recursively breaks the newly inserted broken springs. We adjust the location of the mass-point along the surface normal vector. The magnitude of the adjustment movement h can be easily computed as follows:

$$h = \frac{\psi}{2}\sqrt{(l_{i,j}^0)^2 - (l_{i,j}^t)^2} \tag{8}$$

where ψ is a parameter that controls the area conservation property of the paper model.

IV. Experiment and Results

The control parameters for the proposed method are ϕ and ψ in Eq.7 and Eq.8 respectively. According to the control parameters, the appearance and the behavior of the paper model is changed.

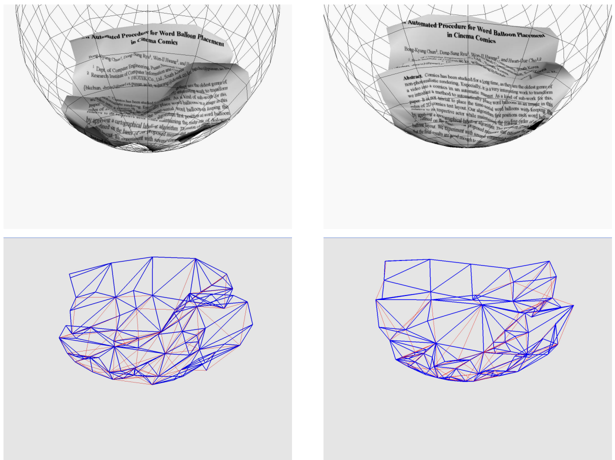
Fig.8 shows the effect of the control parameter ψ. As shown in the figure, a large ψ value generates stiff paper such as card board which you can not bend and which folds if you apply to much force. And a small ψ value generates flexible paper which does not crumble easy (e.g. a bank note).

Fig.9 shows the effect of the control parameter ϕ. The parameter controls the fragility of the paper. Therefore, a large ϕ value produces rubbery paper while a small ϕ produces fragile paper.

Fig.10 shows the animation result when breakable hinge springs are employed in interactive environments. The springs are broken or folded according to the external force exerted by the sphere. As shown in the figure, the breakable hinge spring model can produce plausible paper animation.

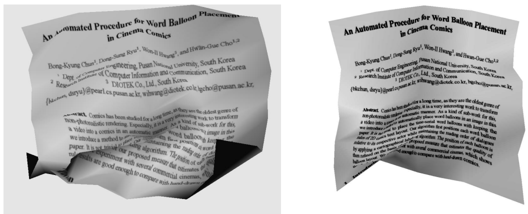
V. Conclusion

In this paper, we proposed simple and intuitive approaches to animate the damage-preserving stiff mass-spring model in order to represent the behavior of paper in virtual environments. The proposed method exploits an approximate implicit method to guarantee the stability of the system, and employs breakable hinge springs to show damage-preserving properties. A fractal model is used to create fine surface details.



(a)ψ=1.0 (b) ψ=0.1

Fig. 8. Effect of control parameter ψ



(a) ϕ=1 (b) ϕ=5

Fig. 9. Effect of control parameter ϕ

The proposed method enables real-time animation of paper objects in virtual environments or game application. The proposed method is intuitive because it is based on a simple and effective mass-spring model and uses a direct update formula instead of solving the large linear system.

The proposed method produces stable paper animations and generates natural-looking wrinkles on the paper surface. Besides the realistic appearance, the proposed method also produces paper-like moving behavior in interactive environments, and can be easily implemented. The proposed method can be successfully utilized for paper animation in real-time or interactive applications.

References

[1] Baraff, D., Witkin, A.: Large steps in cloth simulation. Proceedings of SIGGRAPH 98 pp. 43–54 (1998)

[2] Chu, Y.C., Witten, I.H., Lobb, R., Bainbridge, D.: How to turn the page. In: Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries (JCDL '03), pp. 186–188. IEEE Computer Society Press (2003)

[3] Desbrun, M., Schröder, P., Barr, A.: Interactive animation of structured deformable objects. Graphics Interface '99 pp. 1–8 (1999)

[4] Ida, T., Buchberger: Proving and solving in computational origami. In: 5th International Workshop for Symbolic and Numeric Algorithms for Scientific Computation, pp. 246–263 (2003)

[5] Kang, Y.M., Cho, H.G.: Real-time animation of complex virtual cloth with physical plausibility and numerical stability. Presence - Teleoperators and Virtual Environments **13**(6), 668–680 (2004)

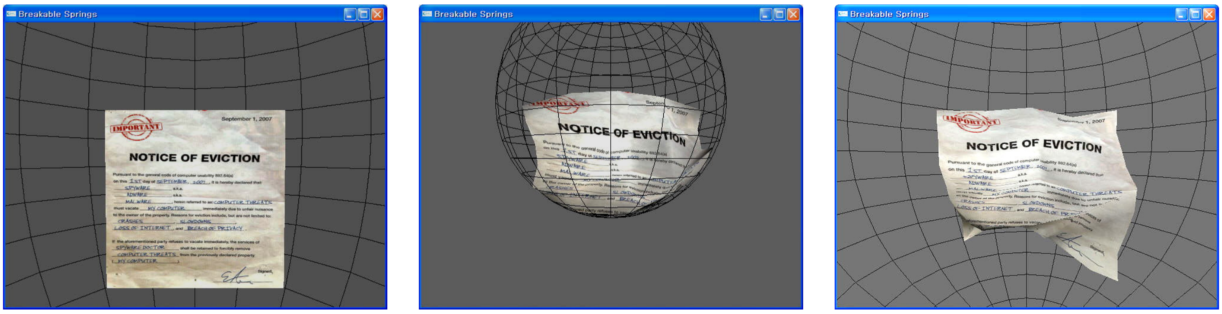


Fig. 10. Interactive animation of the paper model with breakable hinge springs

[6] Meyer, M., Debunne, G., Desbrun, M., Bar, A.H.: Interactive animation of cloth-like objects in virtual reality. The Journal of Visulalization and Computer Animation **12**, 1–12 (2001)

[7] Ng, H.N., Grimsdale, R.L.: Computer graphics techniques for modeling cloth. IEEE Computer Graphics & Applications **16**(5), 28–41 (1996)

[8] Terzopoulos, D., Platt, J., Barr, A., Fleischer, K.: Elastically deformable models. Computer Graphics (Proceedings of SIGGRAPH 87) **21**(4), 205–214 (1987)

[9] Volino, P., Courshesnes, M., Magnenat-Thalmann, N.: Versatile and efficient techniques for simulating cloth and other deformable objects. Proceedings of SIGGRAPH 95 pp. 137–144 (1995)

An Ocean Wave Simulation Method Using TMA Model

Namkyung Lee, Nakhoon Baek and Kwan Woo Ryu

Kyungpook National University

Contents

Our Goal

a new hybrid approach to get a real-time surface gravity wave simulation

- To generate large scale oceans without difficulty
- To work in real time
- To have more controllable parameters

Previous works

1. Approach based on solving Navier-Stoke's(N-S) equation
2. Approach based on the oceanography

Previous works

Approach based on solving N-S equation

- J. Stam, N. Poster, R. Fedkiw
- Heavy computation
- Not suitable for large scene

Previous works

Approach based on the oceanography

- Methods based on the Gerstner swell model
  - ✓ *Fouriner, gonzato*
  - ✓ *Parametric equation are too complex*

Previous works

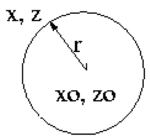
- Methods based on empirical wind driven sea spectrum model
  - ✓ *Mastin, Tessendorf*
  - ✓ *filtering white noise image*
  - ✓ *Height field y=f(x,z)*
  - ✓ *Can not express wave refraction, wave breaking*

Previous works

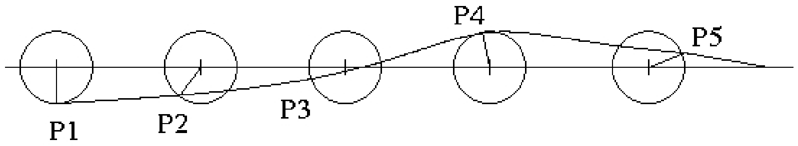
- Hybrid methods
  - ✓ *Hinsinger*
  - ✓ *Spectrum approach + Gestner model*

The Gerstner swell model

Particles move in circular orbits around the rest position



(x_0,z_0) - Rest position



The Gerstner swell model

$$x = x_0 + r\sin(\omega t - kx_0)$$

$$z = z_0 + r\cos(\omega t - kz_0)$$

$k = 2\pi / \lambda$ - Wave number

$\omega = 2\pi f$ - Angular velocity

The surface of an ocean

- **a finite sum of these simple waves**
- **height of the water surface on the grid point**

$$z(x,y,t) = \sum_{i=1}^{n} A_i \cos(k_i(x\cos\theta_i + y\sin\theta_i) - \omega_i t)$$

Spectrum based method

- **Pierson-Moskowitz(PM) model**
  - ✓ *empirically expresses a fully developed sea in terms of the wave frequency* f

$$E_{\mathrm{PM}}(f) = \frac{0.0081 g^2}{(2\pi)^4 f^5} e^{-\frac{5}{4}(\frac{f_{\mathrm{p}}}{f})^4} \qquad f_{\mathrm{p}} = 0.13 g / U_{10}$$

  - ✓ **assumes the infinite depth of the ocean**
  - ✓ **may fail to the shallow sea cases**

Spectrum based method

- **The JONSWAP model**
  - ✓ *fetch-limited seas such as North sea*

$$E_{\text{JONSWAP}}(f) = \frac{\alpha g^2}{(2\pi)^4 f^5} e^{-\frac{5}{4}(\frac{f_p}{f})^4} \cdot \gamma^{e^{-\frac{f/f_p - 1}{2\sigma^2}}}$$

$$f_p = 3.5\left(\frac{g^2 F}{U_{10}^3}\right)^{-0.33}$$

Spectrum based method

- **The TMA model**
  - ✓ *extends the JONSWAP model to include the depth of water*

$$E_{\text{TMA}}(f) = E_{\text{JONSWAP}}(f) \cdot \Phi(f^*, h)$$

$$\Phi(f^*, h) = \frac{1}{w(f^*)}\left(1 + \frac{K}{\sinh K}\right)$$

$$f^* = f\sqrt{h/g} \quad K = 2(f^*)^2 w(f^*)$$

$$w(f^*) = \tanh^{-1}[(2\pi f^*)^2]$$

The TMA model

- **can represent the waves on the surface of lake or small-size ponds, in addition to the ocean waves**
- **wind speed, fetch length, depth of water**

Implementation model

- **Extend the spectrum of TMA model to 2D**

$$E(f, \delta) = E_{\text{TMA}}(f) \cdot D(f, \delta)$$

$$D(f, \delta) = N_p^{-1} \cos^{2p}\left(\frac{\delta}{2}\right)$$

Hybrid method

$$z(x,y,t)=\sum_{i=1}^{n}A_i\cos(k_i(x\cos\theta_i+y\sin\theta_i)-\omega_i t)$$

- **We need to compute frequency, amplitude, wave direction, wave number and pulsation for implementation**
- **the peak frequency**
  - ✓ *Frequency of each wave, wave number, pulsation*

Evaluating parameters

- **According to the random linear wave theory**

$$E(f,\delta)=\Psi(k(f),\delta)\cdot\frac{32\pi^2 f^3}{g^2}$$

- **From the random linear wave, the wave number spectrum**

$$\Psi(k(f),\delta)=\frac{\beta}{4\pi^2}A(f)^2$$

$$A(f)=\sqrt{\frac{E(f,\delta)\cdot g^2}{8f^3\beta}}=\sqrt{\frac{E_{\mathrm{TMA}}(f)\cdot D(f,\delta)\cdot g^2}{8f^3\beta}}$$

Results

Simulation environment

- **PC with ntel Core2 Duo 2.6 GHz processor**
- **Visual C++ 6.0, Opengl API**
- **Not use any hardware based acceleration techniques**
- **More than 60 fps**

Results (Ocean wave)

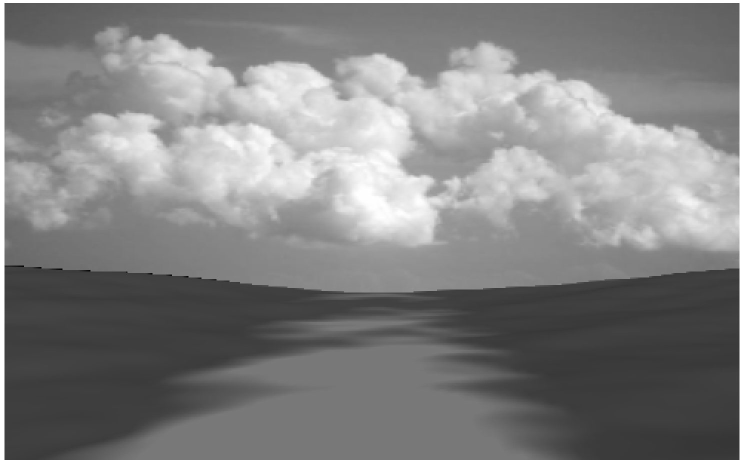
Wind speed : 3m/s , Water depth : 6m , Fetch length : 5km

Results (Ocean wave)

Wind speed : 3m/s , Water depth : 100m , Fetch length : 5km

Results (Ocean wave)

Wind speed : 3m/s , Water depth : 100m , Fetch length : 10km



Results (Ocean wave)

Wind speed : 6m/s , Water depth : 100m , Fetch length : 10km

Conclusions

- **Our ocean wave model**
  - ✓ *display visually plausible scenes even for shallow seas*
  - ✓ *Improved user controllability : more parameters such as fetch length and depth of water, in addition to the wind velocity*
  - ✓ *Can be used in computer games, virtual reality, maritime training simulator*

A unified communication framework using visibility geometry on virtual world space

Pusan National University
GA Lab
Ryu DongSung

A unified communication framework

- **Current Technology of Virtual framework**
  - The communication system provides the 3D virtual environment to communicate on the Internet.
  - Our approach is focused on 3d word balloons.
  - 3d word balloon replacement is very important.

[The 3d online game – SP1, by Nexon Corp., Korea]

Motivation

- **Communication Style on the Internet**
  - The advance of Internet and PC Hardware.
  - Simple chatting style transits 3D Virtual World
  - `Second Life'[1] and `IMVU'[2]
  - Recently, 3D Word balloon is more focused.

(a) Second Life [1] (b) IMVU [2]
[Communication system are used 3D Word Balloons]

Previous Work (1) – 2D

- **Comic Chat [3]**
  - On 2D Image plane
  - Consider Tail Generation➔ Greedy Algorithm

- **An automated procedure for word balloon replacement in cinema comics[4]**
  - On 2D image plane.
  - They solve word balloon replacement by Heuristic method based on Reading Order.

Previous Work (2) – 3D

- **Second Lifestyle[1] and IMVU[2]**
  - On 3D Image plane
  - Word balloons are disappeared by timeline
  - They consider a style of word balloons[1,2]
    - **For example , shout word balloon.**
  - The can visualize history of talks.[2]
  - ➔Just identify of time line information.

Graphics Application Lab

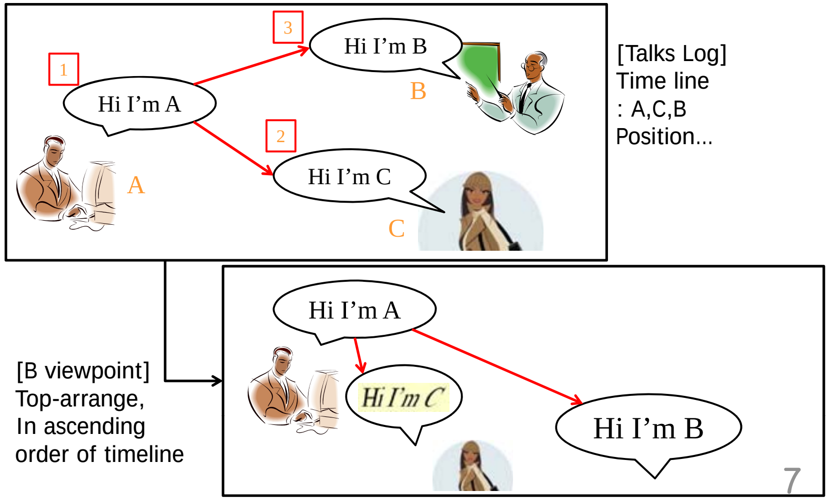
Goal(1)– Overview

- **On the Internet**
  - On the Internet, 3D Platform
  - Java 3D , Java Applet
- **Visibility**
  - Word balloon scene graph
  - Visualize the relationship and history each word balloons
- **Ordering**
  - How arrange word balloons according to time line
  - When we can see 3D scene, we want to know intuitive word balloon history.

Graphics Application Lab

Goal(2)- Visibility and Ordering

- **Examples [Visibility and Ordering]**



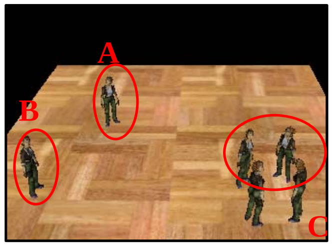
[Talks Log]
Time line
: A,C,B
Position…

[B viewpoint]
Top-arrange,
In ascending
order of timeline

Graphics Application Lab

Goal(3)

- **Partial Talk**
  - Enable group chatting.
  - Group A, B, C want to talk with each group.
  - Group A can't participate in group C's talk.



[Partial Talk Group A, B, C]

Overview of our system

Current chatting system

Position	Shape
– Character – Another character – The collision of each word balloons	– According to text length

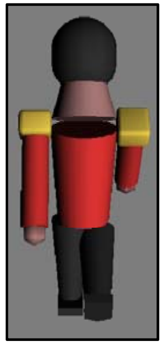
Proposed communication system for 3D virtual world

Position	Visibility	Shape
– Character – Another character – The collision of each word balloons – The order of dialog text	– Direction of view point – Distance of characters – Keyword of dialog contents	– According to text length – The effect of animation

Agent Visibility Model (1)

❑ 3D Characters

- o We used 3Ds max models referenced by 3D character textbook and DirectX 9.0 example.



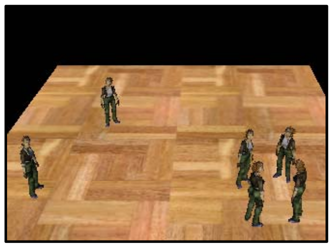
[Toy Soldier]

[Tiny Guy]

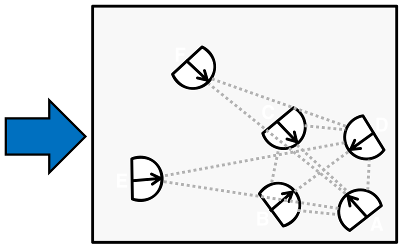
Agent Visibility Model (2)

❑ According to viewing information

- o Angle of view between participant and word balloon.
- o Distance between all participants.



[Virtual Environment]



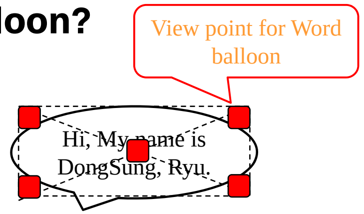
[Viewing Information]

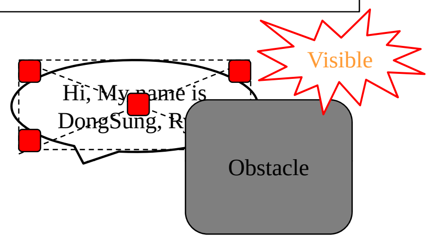
Agent Visibility Model (3)

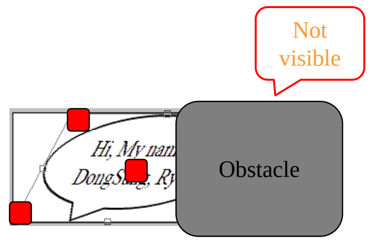
❑ Can you see a word balloon?

```
bool IsVisibleWB(WB){
        Point WBVPos[] = WB.Vis Pts();
        int count = 0;
        for (i=0;i<5;i++){
            if (IsVisiblePt(m_vPos,WBVPos[i])){
                count++;
            }

        if (count >=3)
            return true;
        return false;
}
```



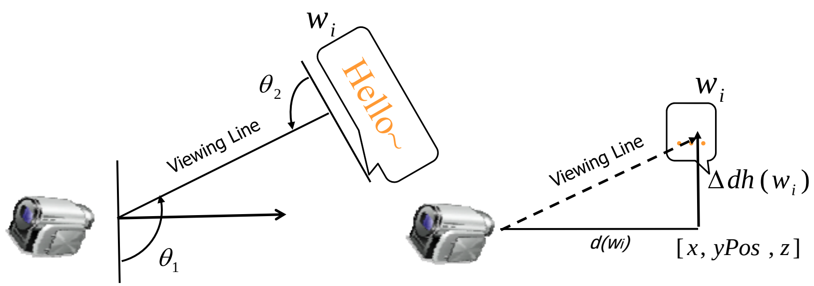




Agent Visibility Model (4)

❑ Degree of Visibility (DoV)

$$DoV(w_i) = c_2 \frac{(\sin\theta_1 \cdot \sin\theta_2)^{k2}}{d(w_i)^{k1} + \Delta dh(w_i) + c_1}$$



Agent Visibility Model (5)

❑ Solving Visibility – Simple Greedy Algorithm

- o $\Delta dh(w_i)$ changes the degree of visibility (DoV)
- o All word balloons are visible. ➔ Terminate Condition.
- o Word Balloons can move upwards.

```
bool GreedyVisibility(WB){
        WBs wbs= allVisibleWB();

        While(true){
            MWB = maxDegreeOfVisibility(wbs);
            bool bTollable = true;
            while (IsAllVisible() ){
                MWB.upPos();
                bTollable = IsTollableIteration();
                if (bTollable){
                    bTollable = false;
                }
            }
            if (!bTollable){
                wbs.replaceLocalMaximumPositions();
            }
        }
}
```

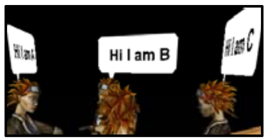

Agent Visibility Model (6)

[View point for A]

[View point for B]

[View point for C]

[View point for E]



[View point for D]


[View point for F]

Ordering and Replacement

❑ Ordering and replacement

- o Reading Order : Top and Down direction
- o Each word balloons shouldn't overlap each other.
- o The sum of Word balloon's DoV is maximum
- ➔Heuristic Solving [Futere work]
- ➔We didn't solve that.

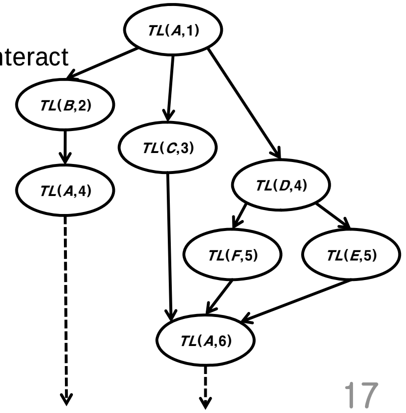
Talk Visibility Graph

❑ **Based on the timeline**

- We just render the relation between visible talks(word balloons).
- We use simple algorithm for generation of talk visibility graph.
- All participants don't need to interact the response action.

[Talk Script, TL(Participant, Timeline *Ti*)]

TL(A,1)	"Where is my book?"
TL(B,2)	"It may be on the table."
TL(C,3)	"I don't know."
TL(A,4)	"But there is not."
TL(D,4)	"Did you see her book?"
TL(F,5)	"No."
TL(F,5)	"I saw the book in the bookcase."
TL(A,6)	"Oh! I see. Thanks."



Conclusion

- ❑ We modeled 3D communication system for 3d virtual environment.
- ❑ We can visualize each word balloons according to all topology information for participants.
- ❑ We propose talk visibility graph to consider the relationship of each visible word balloons.

Future Work

- ❑ We just considered whether all visible word balloons are shown.
  - ➔ Simple Greedy Algorithm.
- ❑ We didn't considered overlap between all visible word balloons.
- ❑ We need heuristic solving algorithm when we can't find positions for all visible word balloons.

References

[1] "Second Life", http://secondlife.com.

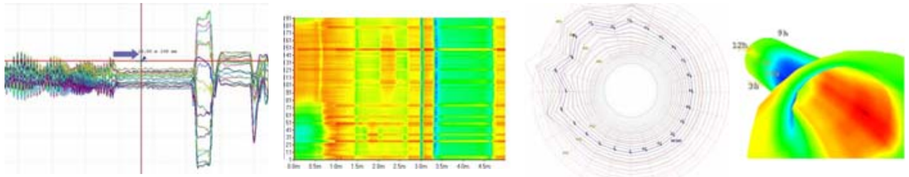
[2] "IMVU", http://imvu.com.

[3] David Kurlander, Tim Skelly, and David Salesin. "Comic Chat", *Proceedings of SIGGRAPH 1996*, pp 225-236, 1996.

[4] Bong-Kyung Chun, Dong-Sung Ryu, Won-Il Hwang, and Hwan-Gue Cho. "An automated procedure for word balloon placement in cinema comics", *LNCS(Lecture Notes in Computer Science) 2006*, pp 576-583, 2006.

[5] Freelancin' Roundtable. http://en.wikipedia.org/wiki/Freelancin'_Roundtable/

[6] Tansu Alpcan, Christian Bauckhage, Evangelos Kotsovinos. "Towards 3D Internet: Why, What, and How?", *2007 International Conference on Cyberworlds*, pp 95-99, 2007.



Visual Analysis System for Pipeline Inspection Data

Sang Ok Koo*, Chang Geol Yoon, Soon Ki Jung

Virtual Reality Laboratory,
Kyungpook National University,
Daegu, South Korea
2008

Multi-Sensor Data

- ❖ **Complex industrial (inspection) systems consist of multiple heterogeneous sensors.**
  - ✓ optical, thermal, magnetic, gyro, speed, pressure, …
- ❖ **The amount of multi-sensor data is usually considerable.**
- ❖ **Sensor data have some noises from the sensor itself or the sensing environment.**
- ❖ **It is very difficult to know**
  - ✓ the exact relationship between a real phenomenon and a sensor data
  - ✓ the dependencies between sensors

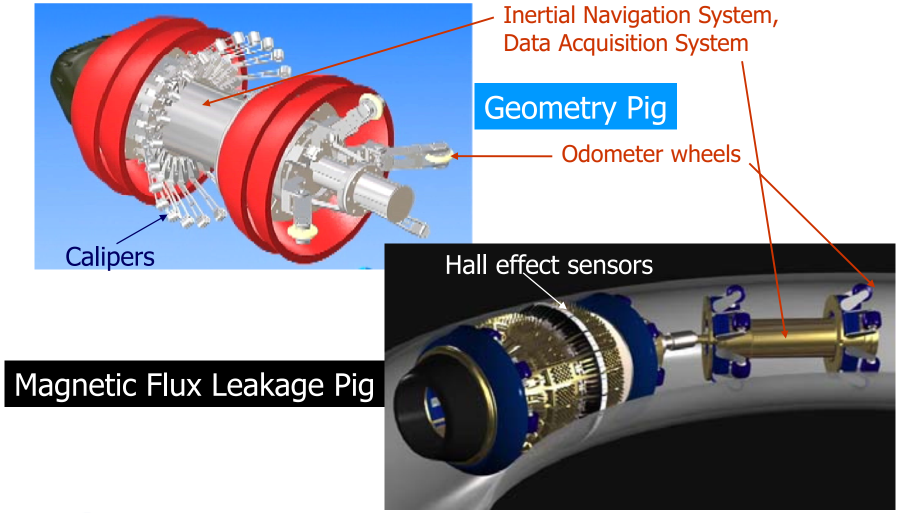
Visual Integration

- ❖ **Integrating multi-sensor data**
  - ✓ Data Integration
    - Need to be sufficiently pre-analyzed by human
  - ✓ Visual Integration
    - Help humans to understand data intuitively
- ❖ **In order to satisfy industrial requirements for analysis accuracy, feature identification and verification *by human* is essential.**

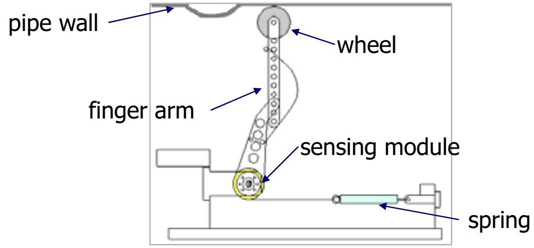
Our work

- ❖ **Development of a software to visualize and analyze multi-sensor time-series data acquired by pipeline inline inspection instruments.**
- ❖ **PIG (Pipeline Inspection Gauge)**

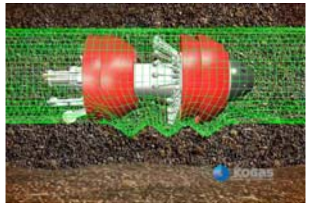
Two Types of PIGs



Geometry PIG Principle

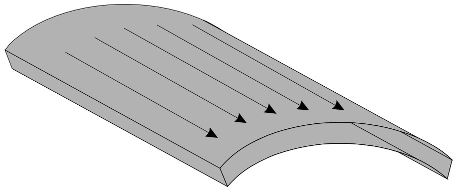



Bend

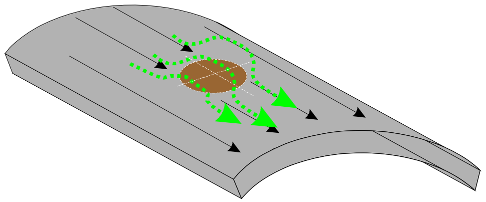

Dent

MFL Basic Principle

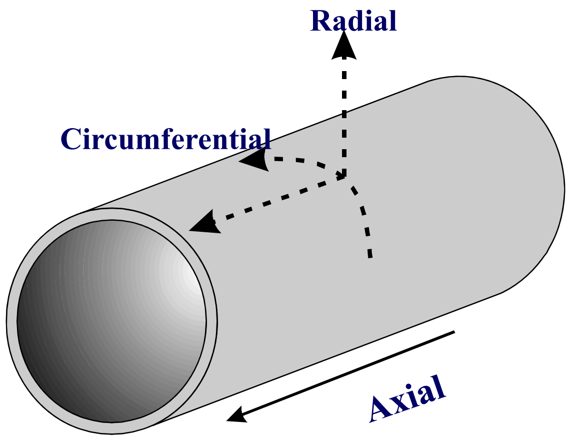
Magnetic Flux Pattern in Nominal Wall



Magnetic Flux Leakage Pattern due to metal loss



Cylindrical Coordinates

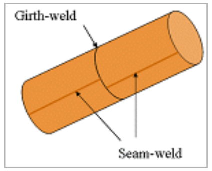


Data Size

PIG	Sensor Type	Sample Data (bytes/sec)	Sample Rates (Hz)	Total Data (Mbytes/h)
Geometry PIG	Finger	48	400(800)	65.9(131.8)
MFL PIG	Axial	384	400(800)	527.3(1054.6)
	Radial	384	400(800)	527.3(1054.6)
	Circumferential	384	400(800)	527.3(1054.6)
	Eddy	128	400(800)	175.7(351.4)
Geometry & MFL PIG	Odometer	6	100	2.1
	IMU	12	100(400)	4.1(16.4)
	Pressure	4	1(10)	0.013(0.13)
	Temperature	2	1(10)	0.006(0.06)

- Pig speed: 4-6 km/h
- Raw data size: over 3Gbytes (2 hours, 400Hz, MFL) about 1.87Gbytes (50km(12.5h), 4km/h, 800Hz, GEO)

Pipeline Features

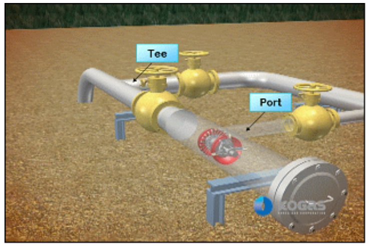


(a) Weld

(b) Bend

(c) Valve

(d) Flange



(e) Tee, Port and Valve

Metal Loss Examples

- Corrosion, crack, gauge …

Related Work

- **Visualization Interface**
  - ✓ "Design Galleries: A general approach to setting parameters for computer graphics and animation", SIGGRAPH97
- **Multivariate data visualization**
  - ✓ Volume model, scatter plot, surface plot, contour plot, parallel coordinate…

  [Pak Chung Wong and R. Daniel Bergeron.]
  "30 Years of Multidimensional Multivariate Visualization. Scientific Visualization: Overview, Methodologies, Techniques," IEEE Computer Society Press, 1997.
- **Time-series data visualization**
  - ✓ ThemeRiver, Spiral graph

  [S. Havre et al.]
  "ThemeRiver: Visualizing Thematic Changes in Large Document Collection," IEEE Transaction on Visualization and Computer Graphics 2002

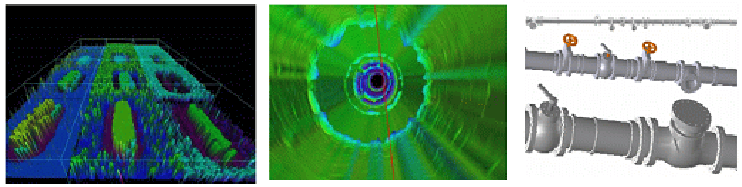
  [Marc Weber, Marc Alexa, and Wolfgang Muller]
  "Visualizing time-series on spirals," IEEE Symposium on Information Visualization 2001.

Related Work

- ❖ **Data integration based on data fusion technique**
  - ✓ MFL, UT, thermal sensor data integration
  - ✓ Integration & Visualization of Graphical, Functional and Measurement data

[Scott Papson et al., January, 2004.]

"A virtual reality environment for multi-sensor data integration," Proceedings of the Sensors for Industry Conference

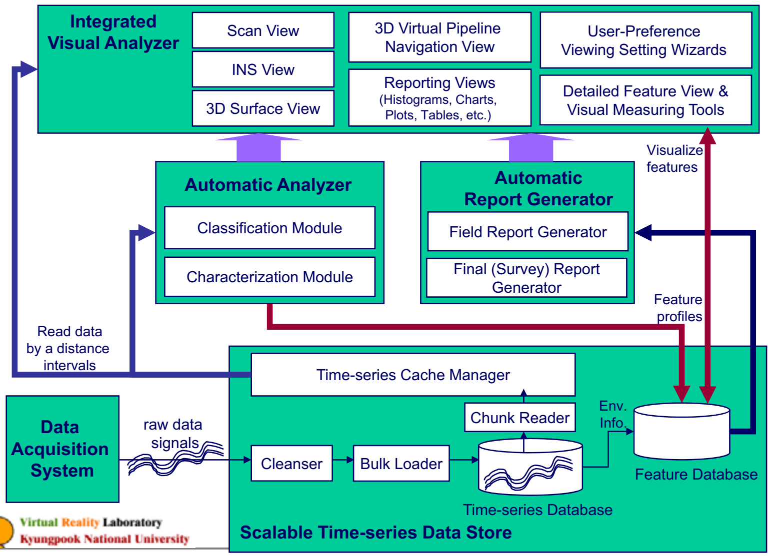


(a) Defect **(b) 3D Pipeline** **(c) 3D model mapping**

Visual Analysis System

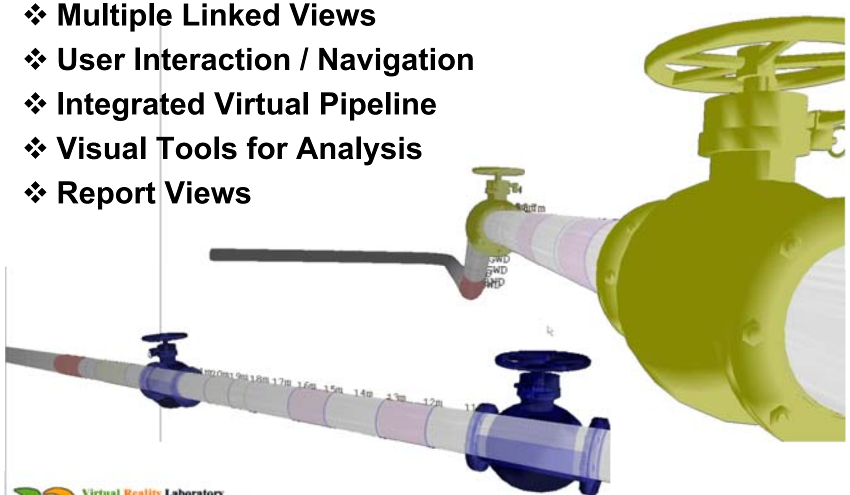
- ❖ **Requirements**
  - ✓ Scalable data management
  - ✓ Integrated Multiple Visualization Interfaces
  - ✓ Interactive Feature Analysis Environment
  - ✓ Rapid data access and free navigation
  - ✓ Automatic/manual feature classification / characterization tools
  - ✓ Automatic Survey Report Generation
  - ✓ Collaborative Analysis Environment

System Architecture

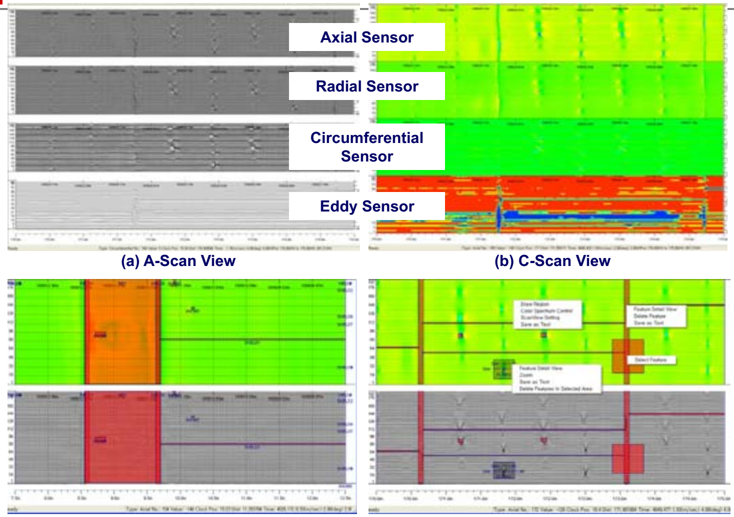


Integrated Visual Analyzer

- ❖ **Multiple Linked Views**
- ❖ **User Interaction / Navigation**
- ❖ **Integrated Virtual Pipeline**
- ❖ **Visual Tools for Analysis**
- ❖ **Report Views**



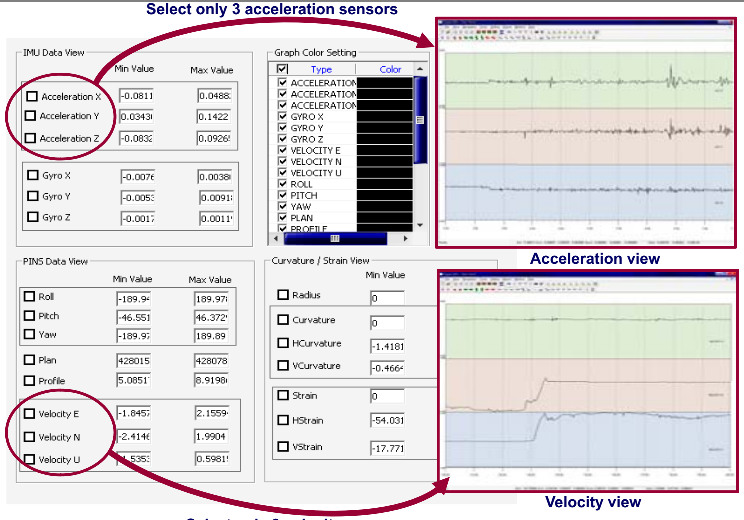
Scan Views



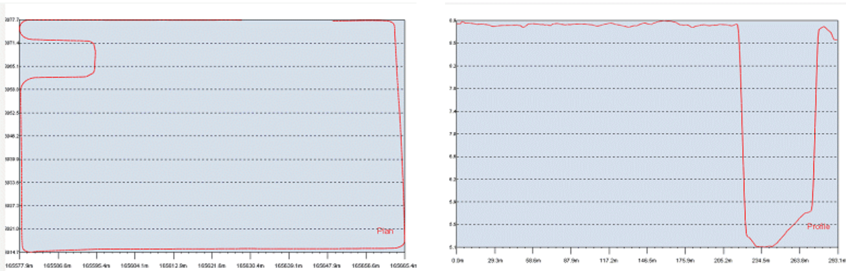
(a) A-Scan View (b) C-Scan View

(c) Features and other info. (d) Feature editing

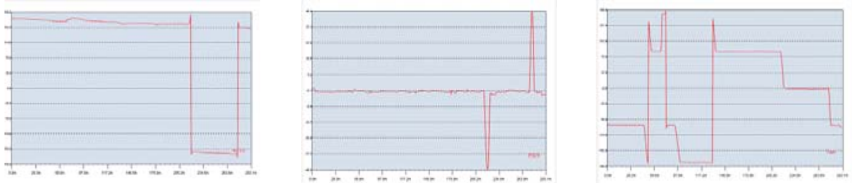
INS Views



Total INS Views

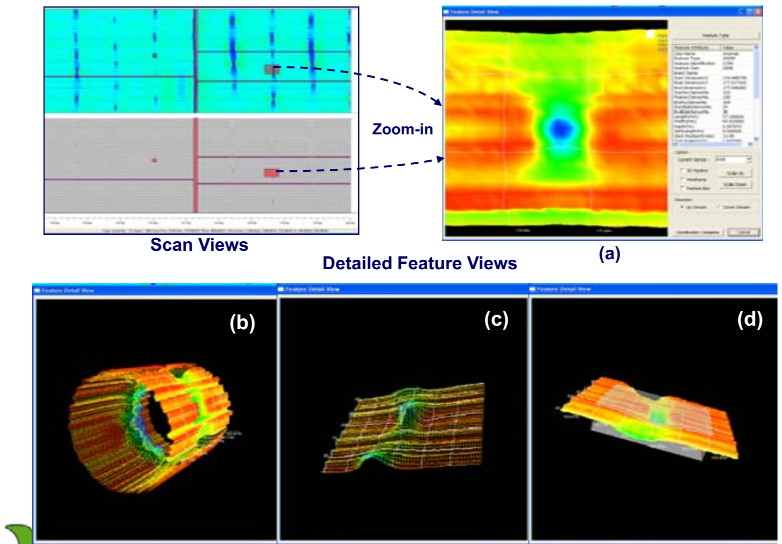


(a) Total Plan (b) Total Profile

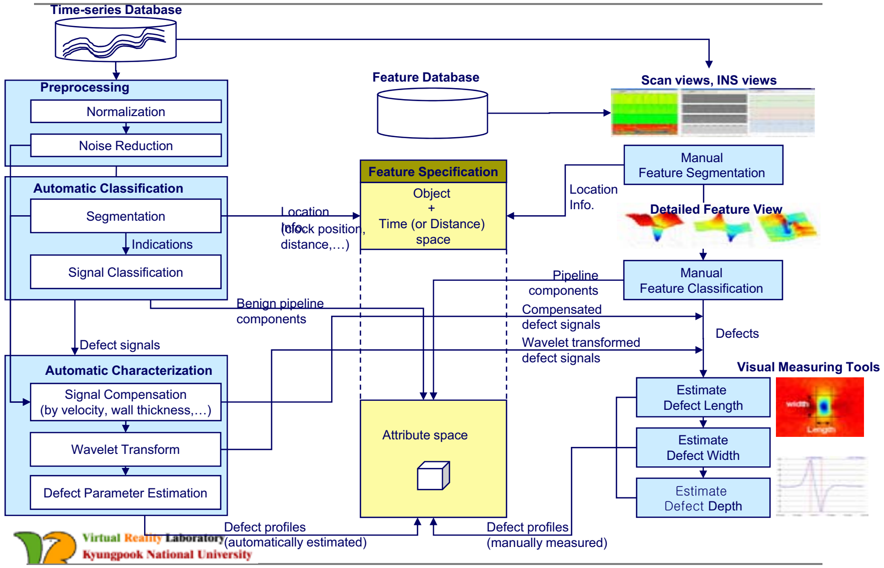


(c) Total Pitch (d) Total Roll (e) Total Yaw

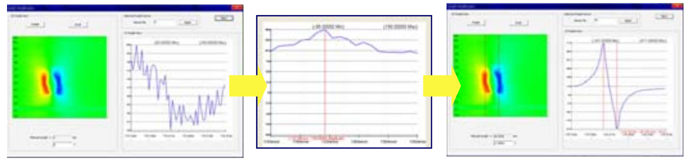
Detailed Feature Views



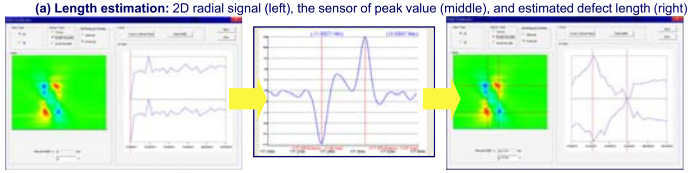
Pipeline Feature Analysis Process



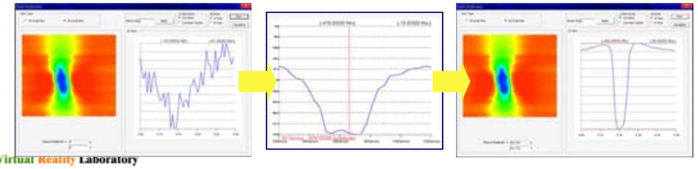
Manual Defect Characterization



(a) Length estimation: 2D radial signal (left), the sensor of peak value (middle), and estimated defect length (right)

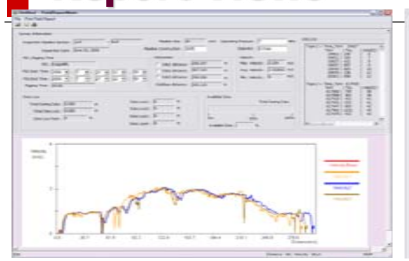


(b) Width estimation: Wavelet-transformed radial signal (left), the sensors of min/max values (middle), and estimated width (right)

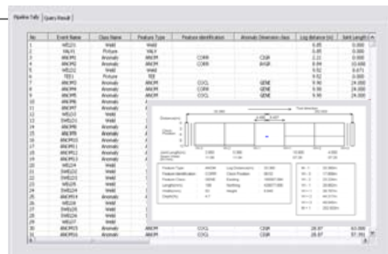


(c) Depth estimation: 2D Axial signal (left), estimation of the peak amplitude (middle), and estimated depth (right)

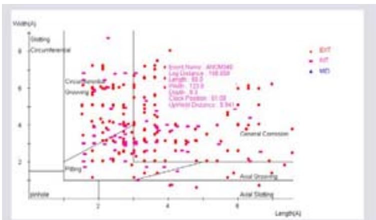
Report Views



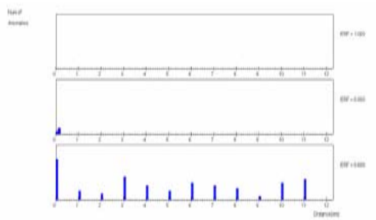
(a) Field report



(b) Pipeline tally & digging sheet

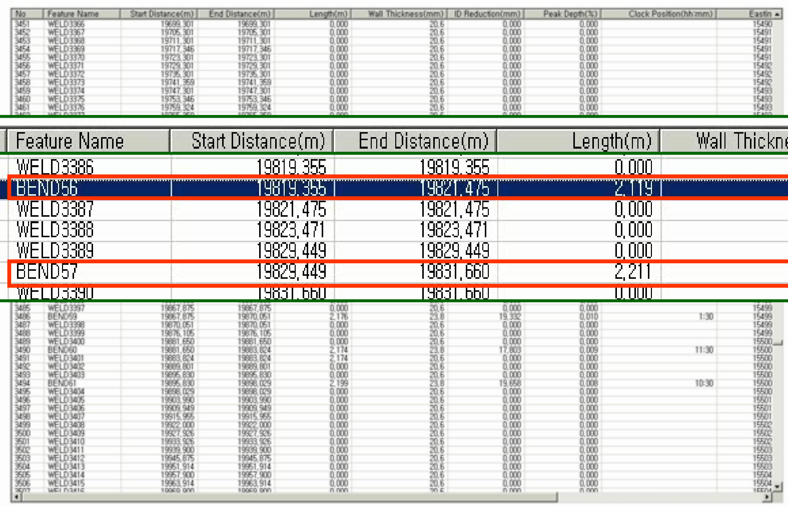


(c) Statistic chart



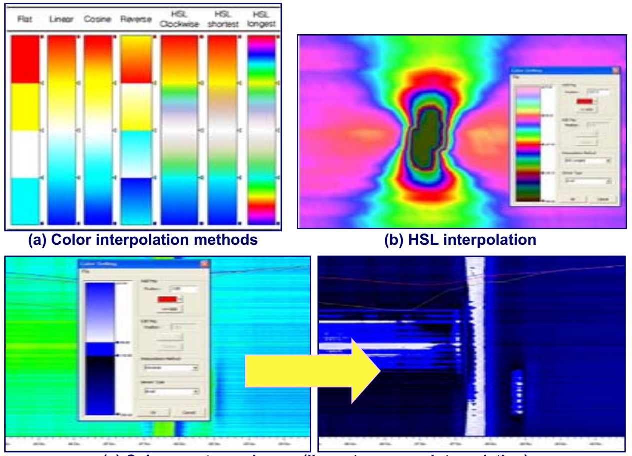
(d) Histogram

Reports - Pipeline Tally



No	Feature Name	Start Distance(m)	End Distance(m)	Length(m)	Wall Thickness(mm)
3471	WELD3386	19819.355	19819.355	0.000	20.6
3472	BEND56	19819.355	19821.475	2.119	23.8
3473	WELD3387	19821.475	19821.475	0.000	20.6
3474	WELD3388	19823.471	19823.471	0.000	20.6
3475	WELD3389	19829.449	19829.449	0.000	20.6
3476	BEND57	19829.449	19831.660	2.211	23.8
3477	WELD3390	19831.660	19831.660	0.000	20.6

Color Spectrum Setting

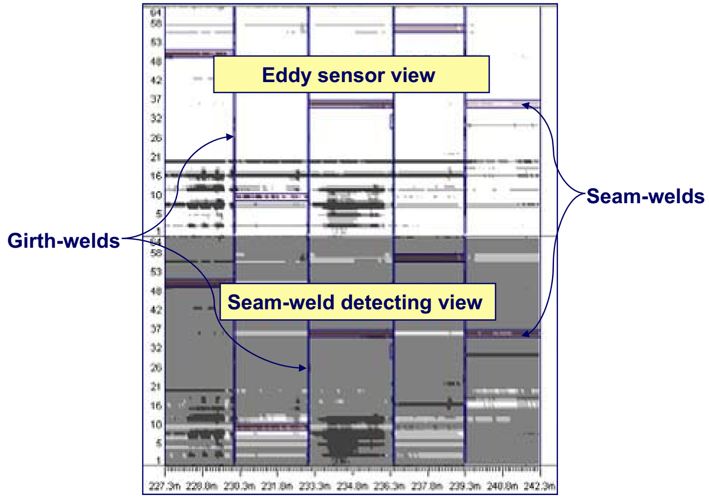


(a) Color interpolation methods

(b) HSL interpolation

(c) Color spectrum change (linear to reverse interpolation)

Visual Warping : Seam-Weld Detecting View

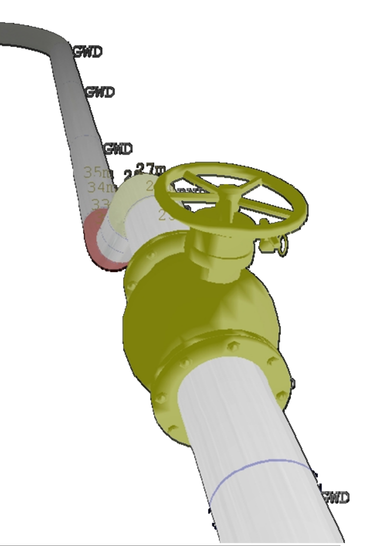


User defined multiple view layouts

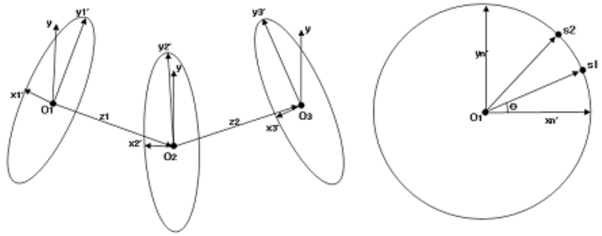
Geometry PIG

Virtual Pipeline

- ❖ **Pipeline Volume Generation**
- ❖ **Navigation Mode**
  - ✓ Outside View
  - ✓ Inside View
- ❖ **Sensor data Integration**
- ❖ **Fixture Integration**



Generation of Pipeline Volume

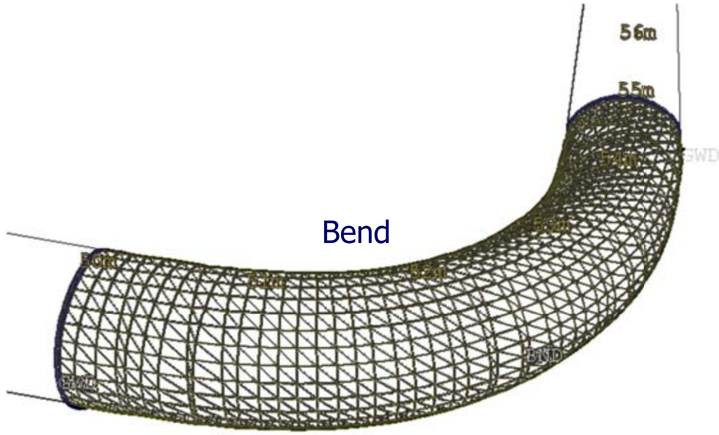


$$y = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad z_n = \overrightarrow{O_n O_{n+1}}$$

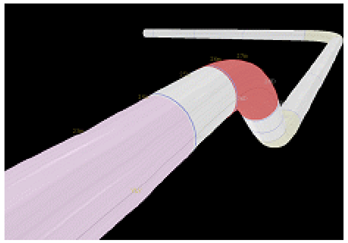
$$x_n = z_n \times y, \quad x_n' = \frac{x_n}{\| x_n \|}, \quad y_n' = x_n' \times z_n$$

$$\begin{aligned} s_1 &= x_n' \cos\theta + y_n' \cos\theta + O_n, \\ s_2 &= x_n' \cos 2\theta + y_n' \cos 2\theta + O_n, \\ &\dots \\ s_m &= x_n' \cos m\theta + y_n' \cos m\theta + O_n, \end{aligned}$$

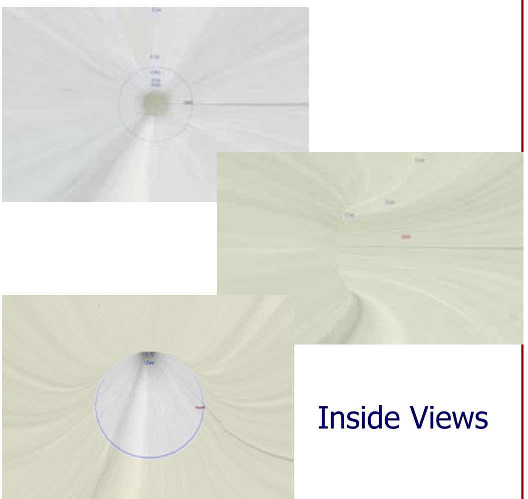
Virtual Pipeline Surface



Virtual Pipe Navigation Mode

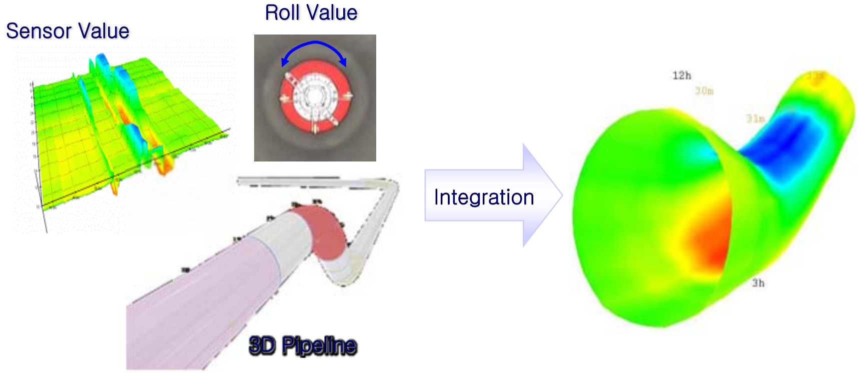


Outside View

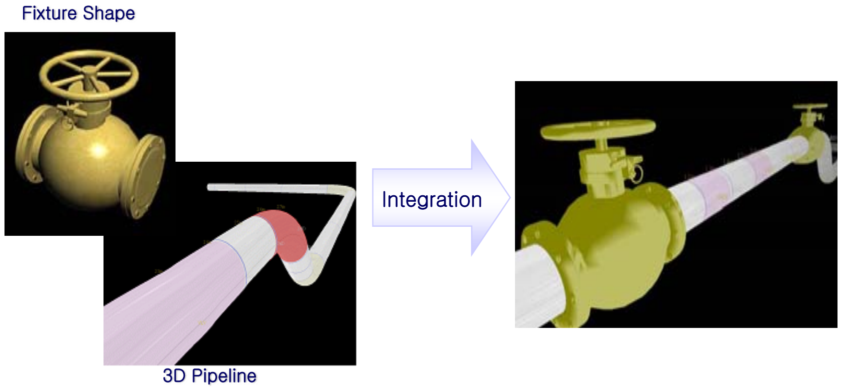


Inside Views

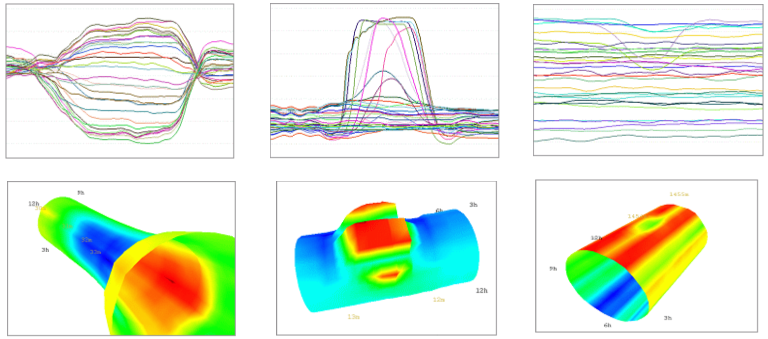
Sensor Value Integration



Fixture Integration

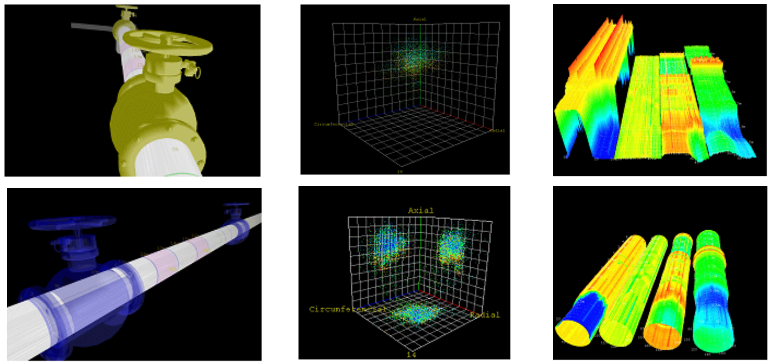


Detected Features (Geometry PIG)



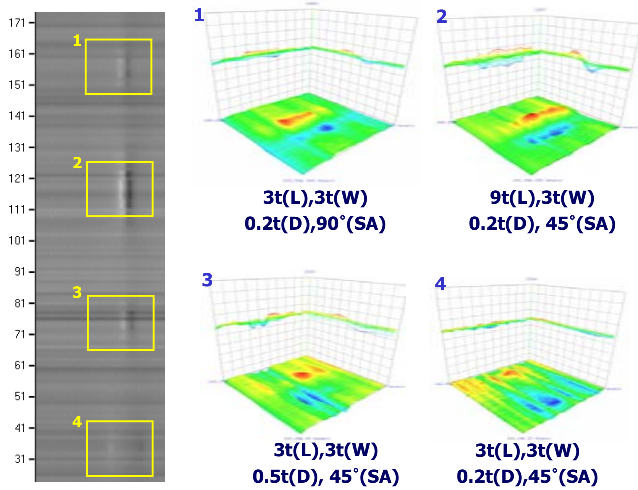
(a) Bend (b) Tee (c) Dent

Various Views (Geometry PIG)



(a) 3D Model mapping (b) Scatter View (c) Integrated 3D Surface

Feature Library (MFL PIG) Demo



Feedback Survey

	VectraView	PipeImage	LinaView	Our System
Viewing	Good	Normal	Normal	Good
Analysis Facilities	Normal	NA	NA	Excellent
Navigation	Good	Normal	NA	Good
Reporting	Poor	Good	Excellent	Excellent
Visualization Quality	Normal	Poor	Normal	Good
Software Speed	Poor	Normal	Normal	Good
Analysis Speed	Normal	NA	NA	Good
Functions	Analysis + View	View Only	View Only	Analysis + View

- VectraView, PipeImage and LinaView are well-known commercial softwares for pipeline feature analysis.

Conclusions

- **Scalable pipeline data processing framework using visualization techniques for analyzing multi-sensor data (specially, in an industrial domain).**
- **Integrated Multiple Visualization**
- **Interactive Feature Analysis Environment**

- **A powerful and user-friendly diagnostic tool for PIG data analysis.**

Biologically Insprired Visual Attention, Object Perception and Knowledge Representation

Minho Lee

School of Electrical Engineering and Computer Science,
Kyungpook National University, KOREA
http://abr.knu.ac.kr mholee@knu.ac.kr

Motivation

A baby mentally grows and becomes an adult… HOW ?

Efficient processing of complex input sensory information
Adapting changing environment
Incremental mechanism for extending knowledge

Human vision system

Biologically motivated approach

- **Selective visual attention**
- **Incremental object perception**
- **Knowledge representation & inference**

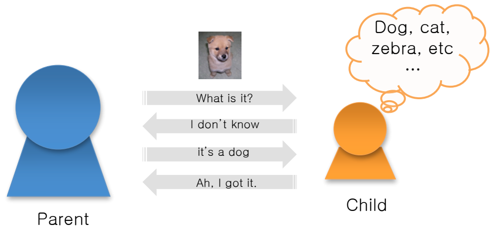
understanding

mimicking

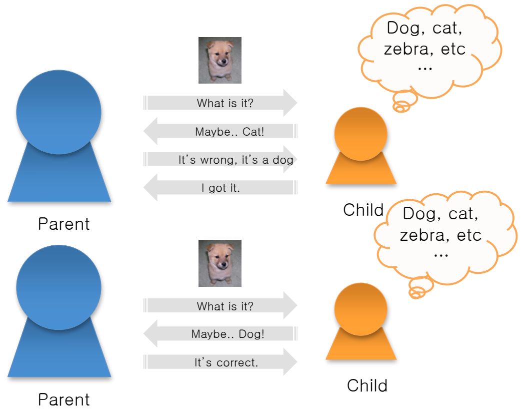
?

Getting close to human level visual intelligence based on biologically motivated approach

AMD with incrementally interactive learning, adaptation and affection



AMD with communicational learning, adaptation and affection

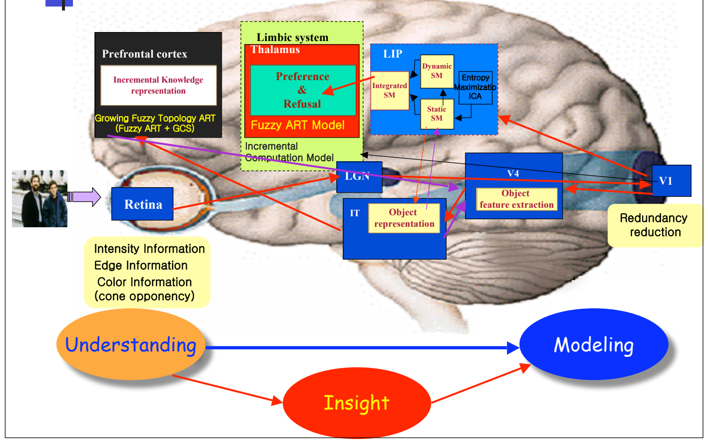


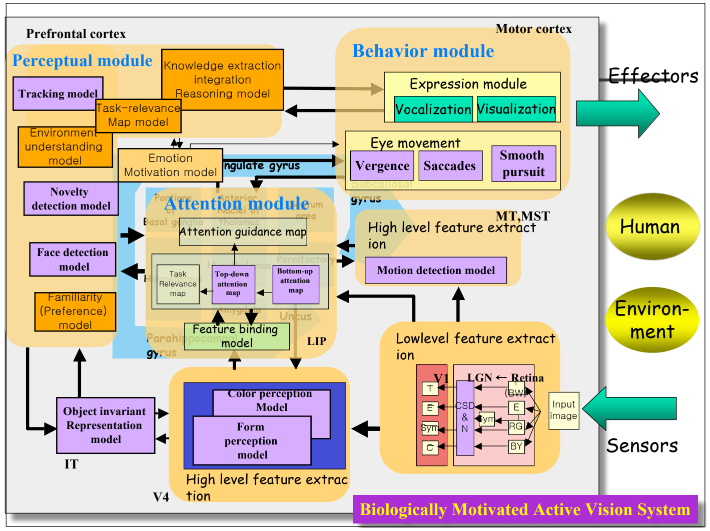
Introduction

- **Integrated visual attention model**
  - Dynamic and static saliency map model
  - Top-down attention model **with human affective factors and object specific attention**
  - Stereo-type attention based on single eye dominant hypothesis
- **Incremental object perception and knowledge representation model**
  - Object knowledge representation and inference based on object perception
  - Growing Fuzzy topology adaptive resonance theory (GFTART) network with push-pull learning
  - Incrementally growing structure with reasoning
- **Hardware implementation**
  - Self-Motivated Artificial Robot with Trainable attention (SMART v1.0)
  - Toward office-mate robotic head vision system

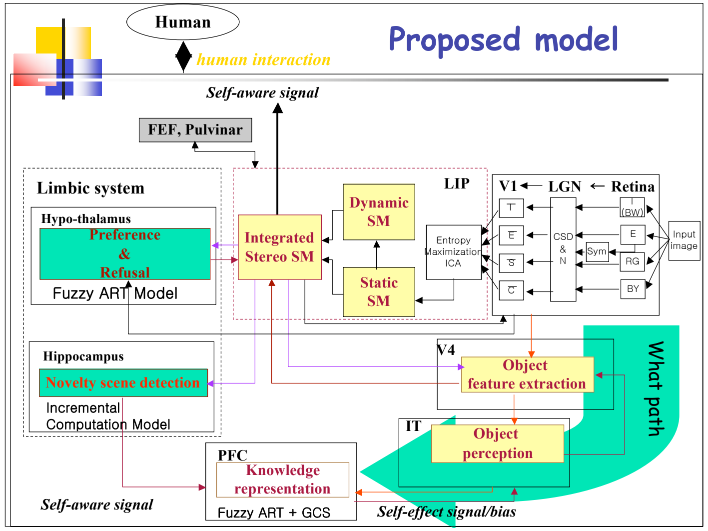
Biological understanding

Biological visual pathway





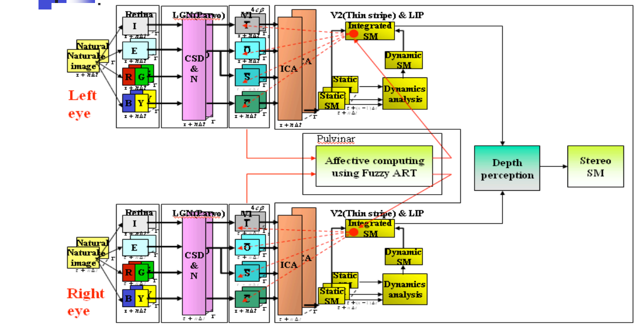
Proposed model



Visual Attention

- **What is attention?**
  - Biological visual system process complex scenes 'serially' (despite parallel computation)
  - Specific parts of the scene are "attended" by covert or overt attention (eye movements).
- **What drives attention?**
  - Bottom up saliency
    - Driven by scene features, Fast!
  - Top down saliency
    - Driven by volitional control, Slow
- **Biological Evidence**
  - Spike modulation observed in V1,V2,V4
    (Luck et al., '97; Reynolds et al '00)

The proposed Bottom-up saliency map model



I : Intensity
E : Edge
S : Symmetry
RG : Color opponency of red & green
BY : Color opponency of blue & yellow
LGN: Lateral geniculate neucleus

CSD&N : Center-Surround Difference & Normalization
ICA : Independent Component Analysis
SM : Saliency Map
LIP: Lateral-intera parietal cortex

$\bar{I}$: Intensity feature map
$\bar{O}$: Oriented feature map
$\bar{S}$: Symmetry feature map
$\bar{C}$: Color opponency feature map

Symmetry information

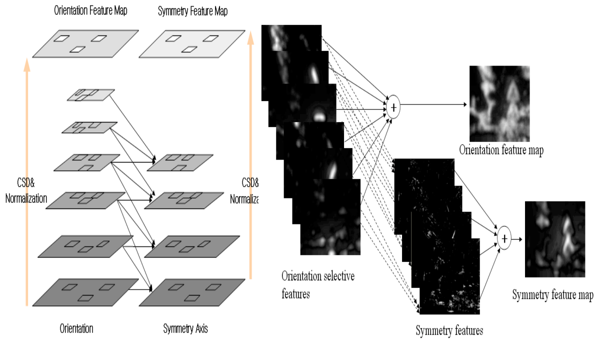


Figure 2: Symmetry feature map generation process.

Feature integration based on redundancy reduction

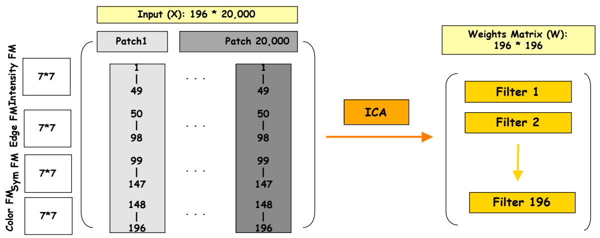
Independent Component Analysis(ICA)

Barlow

Human's feature detectors might be the end result of a redundancy reduction process

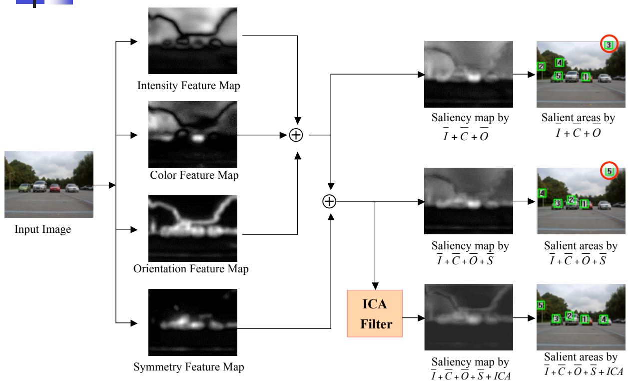
Bell and Sejnowski

ICA is the best representation for redundancy reduction process



ICA filter generation

Bottom-up selective attention comparison



Object preferable selective attention comparison

Image : 194 images / Salient areas : 500 areas			
	I+O+S+C+ICA	I+O+S+C	I+O+C
1st	165	150	143
2nd	106	104	103
3rd	68	77	64
4th	66	57	47
5th	46	32	28
Total	451	420	385
%	90.2 %	84.0 %	77.0 %

Dynamic bottom-up saliency map model

$$s(\mathbf{x}) = \arg\max_{s}\{H_D(s,\mathbf{x},\tau)\times W_D(s,\mathbf{x},\tau)\}$$: degree of saliency at location x

$$H_D(s,\mathbf{x},\tau) \equiv -\sum_{d\in D} p_{d,s,\mathbf{x},\tau}\log_2 p_{d,s,\mathbf{x},\tau}$$: the entropy

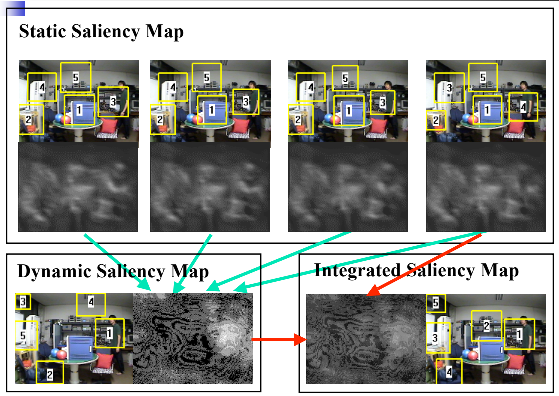
$$W_D(s,\mathbf{x},\tau) \equiv \frac{s^2}{2s-1}\sum_{d\in D}\left|p_{d,s,\mathbf{x},\tau} - p_{d,s-1,\mathbf{x},\tau}\right|$$: the inter-scale measure

$p_{d,s,\mathbf{x},\tau}$ is the probability mass function at time τ for scale s, position $\mathbf{x}$, and the descriptor value d which takes on values in D

$$T_D(s(\mathbf{x}),\mathbf{x},\tau+n\Delta t) \equiv -\sum_{d\in D} p_{H_D(s(\mathbf{x}),\mathbf{x},\tau+n\Delta t)}\log_2 p_{H_D(s(\mathbf{x}),\mathbf{x},\tau+n\Delta t)}$$

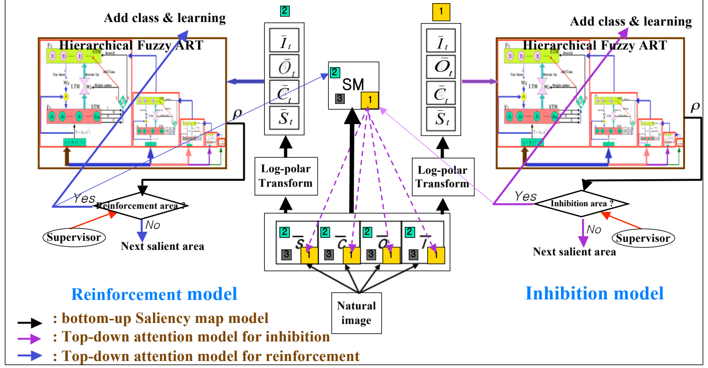
$p_{H_D(s(\mathbf{x}),\mathbf{x},\tau+k\Delta t)}$ is obtained from the histogram of the entropy values at location $\mathbf{x}$ for a sequence of frames.

Dynamic bottom-up saliency map model

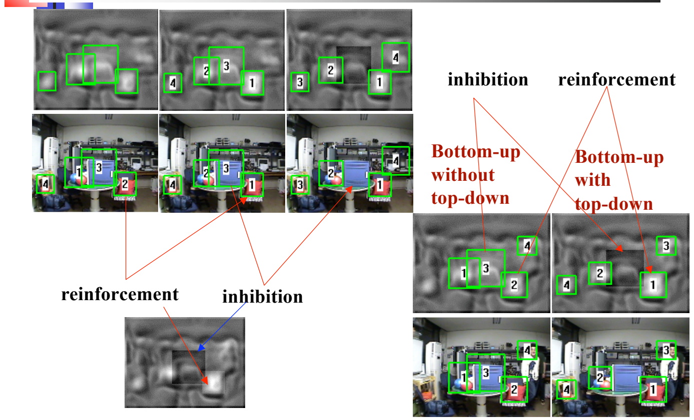


Considering affective factors

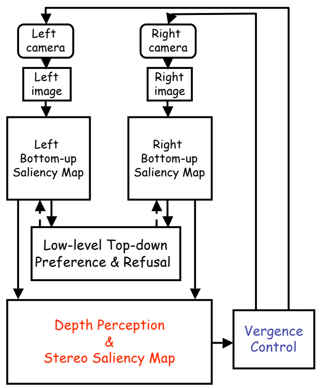
- Reflect human preference
- Generate more human plausible scan path through inhibition uninteresting areas and reinforcing interesting areas

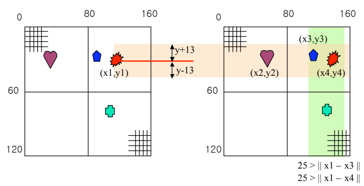


Results of affective saliency map model



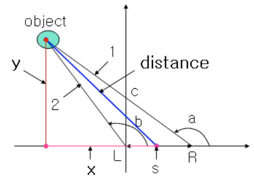
Stereo saliency map based vergence control

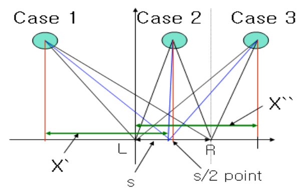




Based on single eye alignment

Depth estimation





$$\tan(a)\cdot x + b = y$$

$$\tan(b)\cdot x = y$$

$$x = \frac{\tan(a)}{\tan(a)-\tan(b)}\cdot s$$

$$\therefore\ y = \frac{\tan(a)\cdot\tan(b)}{\tan(a)-\tan(b)}\cdot s$$

Case 1 : b>90 and a>90

$x = y/\tan(b)$ $\quad x' = |\,y/\tan(b)\,| + s/2$

$d = \sqrt{x'^2 + y^2}$

Case 2 : b<=90 and a>=90

$d \cong y$

Case 3 : b<90 and a<90

$x = y/\tan(b)$ $\quad x'' = |y/\tan(b)| - s/2$

$d = \sqrt{x''^2 + y^2}$

Experimental results


(a) The first landmark selection of left and right CCD camera


(d) The second landmark selection of left and right CCD camera


(b) The first saliency map of left and right CCD camera


(e) The second saliency map of left and right CCD camera


(c) The first result of vergence control by motor control


(f) The second result of vergence control by motor control

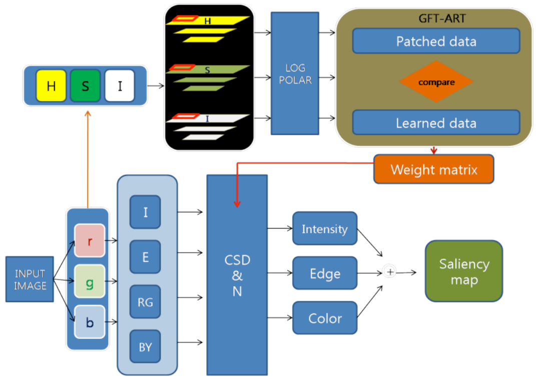
Experimental results

SALIENT POINT	DISTANCE	WEIGHT
1	296.10	71
2	175.01	100
3	354.75	48
4	291.00	73
5	X	X

Limitation of Bottom-up SM Model

- Strength
  - Explains **"what"** is there
  - Is **invariant to scale and location**
  - Shares features among all object categories
  - Based on **feed-forward** only connections
- Weakness
  - By construction, cannot explain **"where"** the object are
  - Cannot explain **"how many"** objects there are
  - Accuracy degrades in presence of **clutter/multiple objects**
  - Based on feed-forward only connections

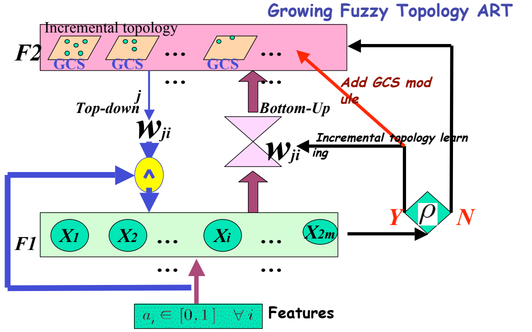
Feature-level top-down biased attention model



Growing Fuzzy Topology ART

- increases stability in conventional fuzzy ART while maintaining plasticity
- preserves topology structures in input feature spaces
- makes clusters to construct an ontology map in the color & form domains

Growing Fuzzy Topology ART



Growing Fuzzy Topology ART

Fuzzy ART

Between-class representation
No topology relation between classes

GCS

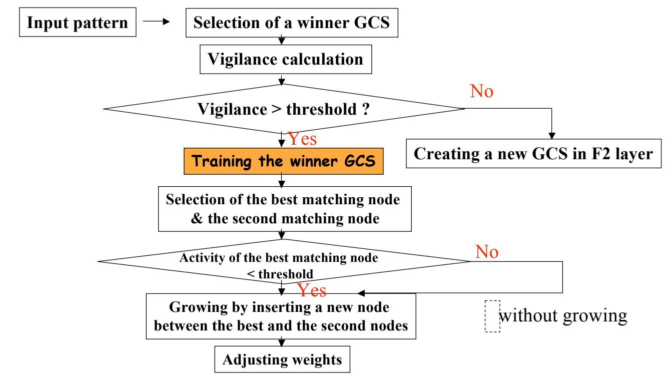
Between-class topology
Weak Within-class topology

Growing Fuzzy Topology ART

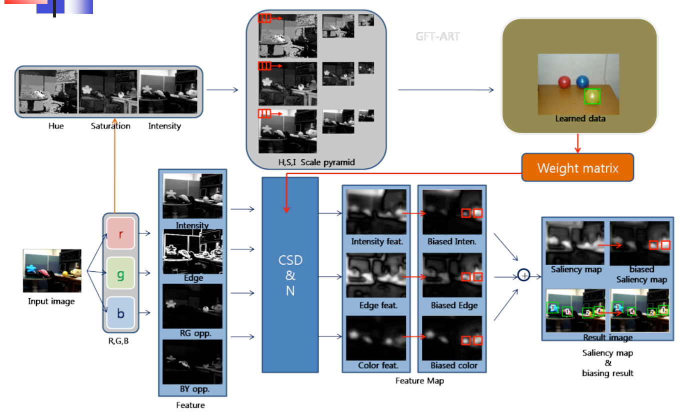
Between-class representation
Within-class topology

Growing Fuzzy Topology ART

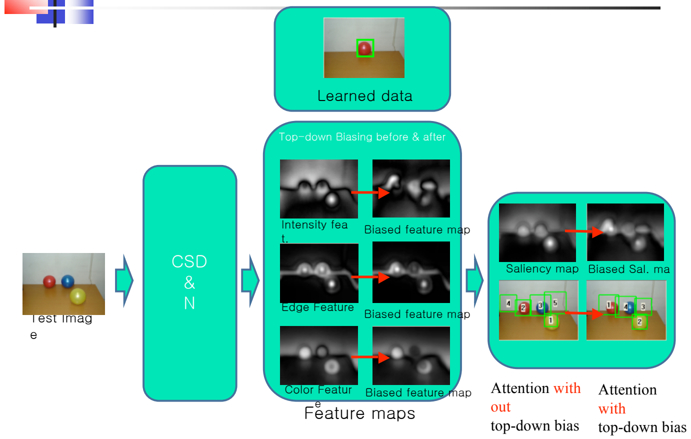
Training Growing Fuzzy Topology ART



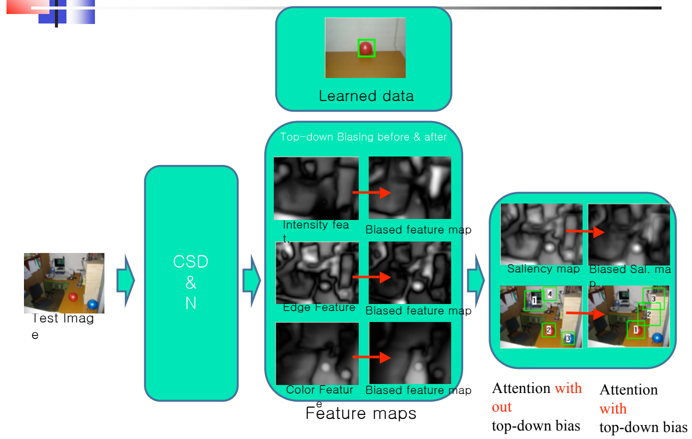
Experimental results of feature-level top-down biased attention model



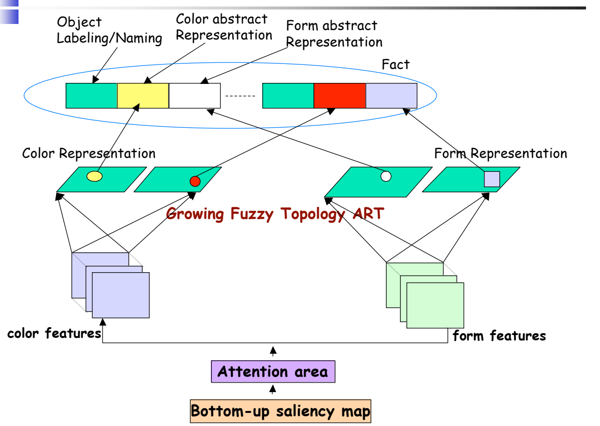
Top-down bias



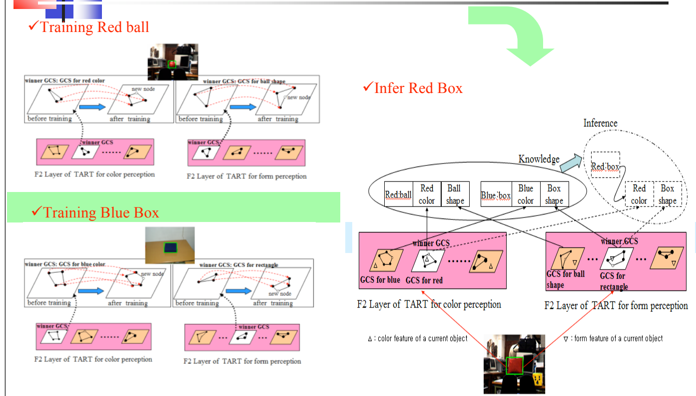
Top-down bias



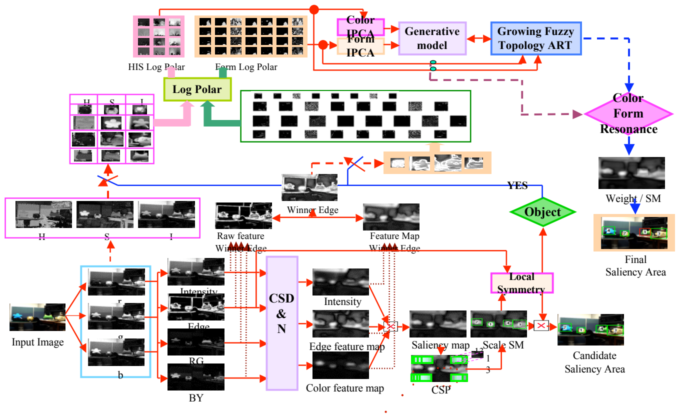
Incremental learning for visual top-down attention



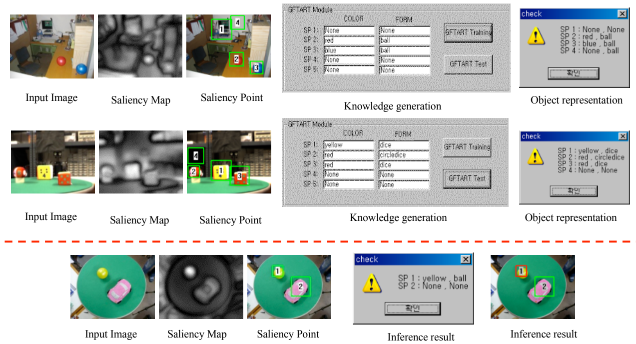
Experimental result for knowledge representation and inference



Implementation of knowledge representation

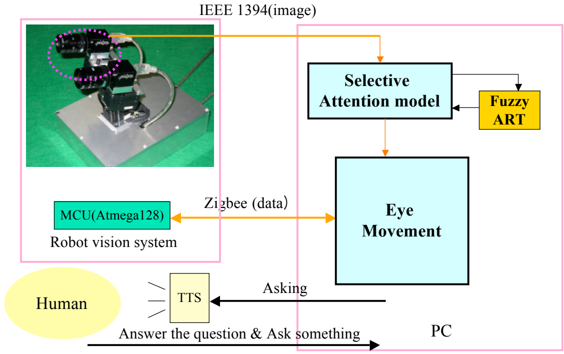


Experimental results (Inference)



System implementation(SMART-v1.0)

Self-Motivated Autonomous Robot with Trainable selective attention



Conclusion & Further works

- Development of a human-like selective attention model with th e dynamic bottom-up SM model and the top-down preference and refusal attention model
- Development of an object non-specific perception model with in cremental mechanism
- Development of a knowledge representation model by integrating Fuzzy ART and GCS
- Need fully integration of all the sub-models and more experim ents for performance verification of the proposed model
- Application: Vision system of Artificial secretary

Toward Biologically Inspired In/Out Vehicle Monitoring Model for Interactive Safety Driving Agent System

Woong-Jae Won

***Abstract*— As an increment of order-drivers and in-vehicle-infortainment system, the adaptive human machine technology have been received more attention from many intelligent vehicle research community for safety driving system. In order to implement this technology, the driver's and out-vehicle-environment monitoring technologies are really important issue. However, the previous models have troubled with influence of illumination change and image distortions in real situation. Therefore, in this paper, we introduce not only road traffic sign detection method for perception of out-vehicle environment, but also driver's head and hand detection methods for monitoring driver's inattentive behavior such as driver's visual behavior and activity based on modified selective attention model as a firs step for overcoming these problem.**

***Index Terms*— the adaptive human machine technology, driver and out-vehicle-environment monitoring system, human-like selective attention model.**

I. Introduction

Rthe National Highway Traffic Safety Administration (NHTSA) reported that the 25-35% of the traffic accidents or 1.2 million vehicle crashes per year in the United State are resulted from distraction and inattention [1, 2]. And, the Treat et. Al reported that the human error mainly cause the vehicle crash [3]. Furthermore, drivers are faced to an increasing number of on-board functions and in a near future of the massive introduction of in-vehicle system such vehicle information, entertainment, and assistance systems (audio, navigation, and mobile phone, lane departure warning system, etc). On the other hand the driver is not always in good situation to receive and understand the messages that are given to him by the vehicle/system. Major are his current status like tired, distraction, absent minded, and etc. So, it's known that more increasing number of these in-vehicle systems dramatically increase car accidents as resulting by decreasing a driver's attention. In addition, Individuals aged 65 years and over represents the most rapidly growing segment of the driving population, and are keeping their licenses longer[4, 5]. However, the order drivers not only have been trouble in detecting unpredicted visual event because of there are faced impaired visual functioning catact and related visual impairment is highly prevalent, but also are slower reacting against unpredicted situation than normal drivers[5].

The adaptive human machine interface(AHMI) technologies such as interactive workload manager and assistance system have been received more attention from many intelligent vehicle research community to support and disburden driver to significantly increase driving safety and comfort[6, 7, 8]. In order to implement these systems, the technologies of recognizing driver's inattentive behavior and out-vehicle environment have been important issues for research in computer vision research community [7, 8].

However, the previous models have troubled with influence of illumination change and image distortions in real situation.

Therefore, in this paper, we introduce the new approach for detection of road traffic signs and driver's head/hands based on modified human-like selective attention model in order to solve those problems. Since the color information of a road traffic sign board mainly good contrast level against the visual environment, we consider red-green and blue color opponent feature information[9]. And, in order to detect driver's face and hand regions for estimating driver's activity, we consider skin color filtered normalized color features to separate between hands candidate regions and background. Then, we adopt for selective attention model for making the saliency map to reduce influence of noise and reinforce the interesting areas which are road traffic sign region, and driver's head/hands[10, 11, 12]. After constructing the saliency map, the proper scale of road traffic sign and driver's hand regions are obtained through the local energy maximum searching and region growing based entropy maximization algorithms[10, 12]. And, the driver's head regions are obtained from histogram projection and ellipse fitting method. Moreover, In order to track and get driver's head pos, we adopt Active Appearance Model. And, in order to measure driver's visual behavior and head movement, the driver's gaze and eyelid/mouth movement are obtained from the face feature point which are getting from the eye feature analysis and Active appearance based feature points[13, 14].

Manuscript received April 21, 2008. (This work was supported by the Daegu Gyeongbuk Institute of science and technology(DGIST) Basic Research Program of the Most..

Woong Jae Won is with Division of Advanced Industrial Science and Technology, Daegu Gyeongbuk Institute of Science and Technology, 75 Gondanbu2gil, Dalseo-Gu, Taegu 704-230, Korea. (e-mail: wwj@dgist.ac.kr).

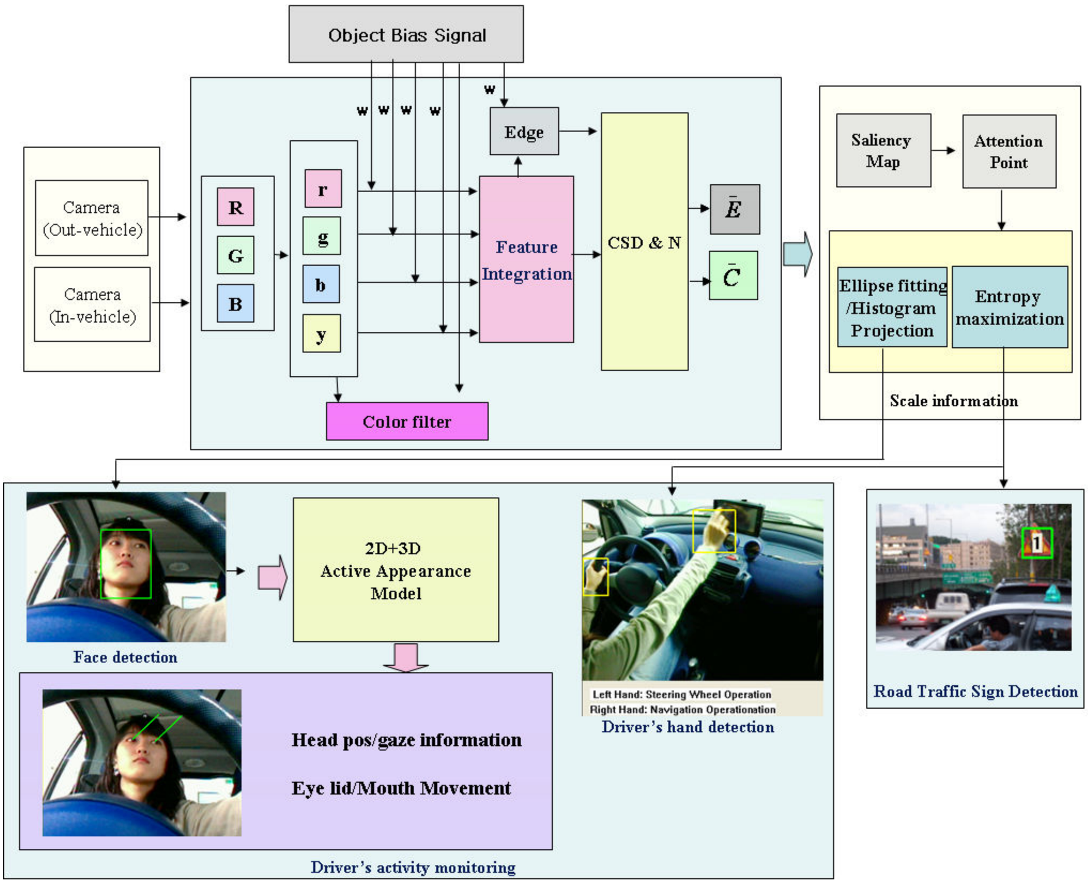


Fig. 1. Proposed Selective Attention based in/out-vehicle monitoring model

II. Selective Attention Based In/Out-Vehicle Monitoring Model

As a previous work, we have been introduced biological motivated selective attention based incremental object perception model which was implemented by integrating specific object preferable attention mode as a giving top-down bias signal to low feature information[10]. Therefore, we consider this approach for in/out vehicle monitoring model as an application.

Fig.1 shows the proposed modified selective attention based in/out-vehicle monitoring model for interactive safety driving agent system. After extracting the red(R), green(G), blue(B) color feature from input color image, The normalized red(r), green(g), blue(b) and yellow(y) color features are extracted from the R, G, B for each And, the bias signals are given to low-level color feature for detecting target object such as road traffic sign, driver's head/hand position.

Then, the center-surround difference and normalization (CSD &N) is considered to making edge and color feature map following the given task. After making color feature map, the object-oriented saliency map constructing through the weight sum of each feature map. And, the object area and adaptive scale information are extracting through simple searching local maximum energy with entropy maximization algorithms or threshold and histogram based segmentation method[10, 11].

Moreover, we consider 2D+3D active appearance Model (AAM) to track the driver's gaze direction, and estimate driver's visual and facial behavior such as eyelid and mouth movement.

A. Road Traffic Sign Detection

Since, the color information of a road traffic sign board mainly good contrast level against the visual environment, the red-green and blue-yellow (rg_by) color opponent features biased for detect the road traffic sign detection.

After extracting R, G, B color feature from the input color image, the r, g, b, and y color features are extracted form the R, G, B. The rg_by color opponent features are extracted from in order to reflect road traffic color contrast characteristics. Also, the edge of red-green and blue-yellow (rg_by) color opponent feature is considered as road shape information in order to reflect load sign shape characteristics

Therefore, the rg_by color opponent feature and edge information(e) are obtained by Eqs. (1) and (2)

$$rg\_by = \left\| r - g \right| - \left| b - y \right\| \quad (1)$$

$$e = \sqrt{(rg\_by \cdot S_x)^2 + (rg\_by \cdot S_y)^2} \quad (2)$$

Fig. 2 shows the procedure of extracting the color and edge features. As shown Fig. 2, the edge of rg_by color opponent feature is dominant in the road traffic regions

Fig. 2. The procedure of extracting edge and color features

Then, in order to reduce noise influence in various scenes and intensify the road traffic sign regions, we implement the on-center and off-surround operation by the Gaussian pyramid images with different scales from 0 to nth level whereby each level is made by the sub-sampling of 2^n, thus it is able to construct 3 feature bases such as $E(\bullet)$, $RG(\bullet)$, and $BY(\bullet)$ [9, 10]. Then, the center-surround features are constructed by the difference operation between the fine and coarse scales in the Gaussian pyramid images using Eqs (3) and (4)

$$\overline{E} = \bigoplus_{c=2}^{3} \bigoplus_{s=c+2}^{3} N(e(c,s)) \tag{3}$$

$$\overline{C} = \bigoplus_{c=2}^{3} \bigoplus_{s=c+2}^{3} N(RG(c,s) + BY(c,s)) \tag{4}$$

Then, the road traffic saliency Map(RTS_SM) is generated by Eq. (5).

$$TS(x,y) = (W_c \cdot \overline{C}(x,y) + W_E \cdot \overline{E}(x,y)) \tag{5}$$

where W_c and W_E represent the weight factor for color feature map and edge feature map, respectively.

As the shown in Fig. 3, the 5 layers of RG, BY color opponent and edge Gaussian pyramid features are selected from the 7 layer Gaussian pyramid features which are extracted from the r, g, b, y, and rg_by color opponent. Then, these pyramid features are constructing 4 layers of CSD images for color and edge

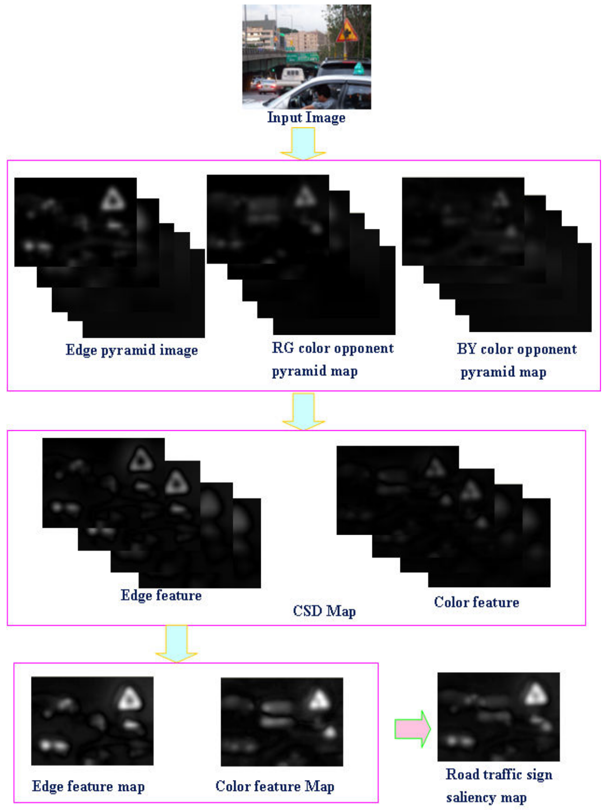


Fig. 3. The procedure of extracting the road traffic sign saliency map

feature maps through the CSD processing. The color and edge feature maps are obtained by normalizing each feature after summing the 4 layers CSD features. After we get the color and edge feature maps, the road traffic sign map is extracted through the weighted sum of edge and color feature map. The road traffic sign map effectively indicates more salient road traffic areas than another area by considering the characteristics of color contrast information in road traffic sign boards

Since the RTS_SM intensifies the road traffic sing regions and diminishes complex backgrounds, we can simply select a candidate region for a road traffic sign through searching the

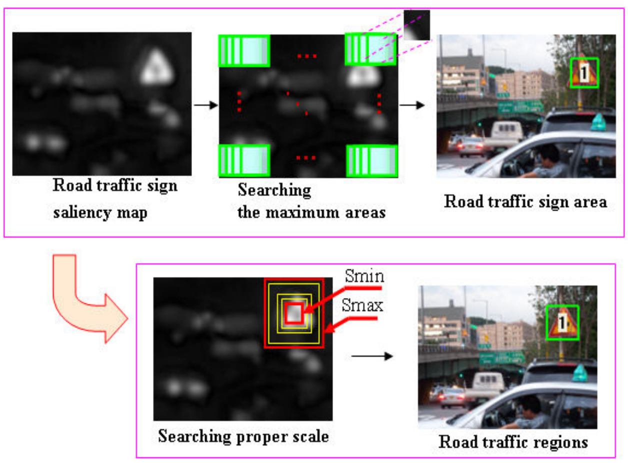


Fig. 4. Selection of candidate region for road traffic sign boards

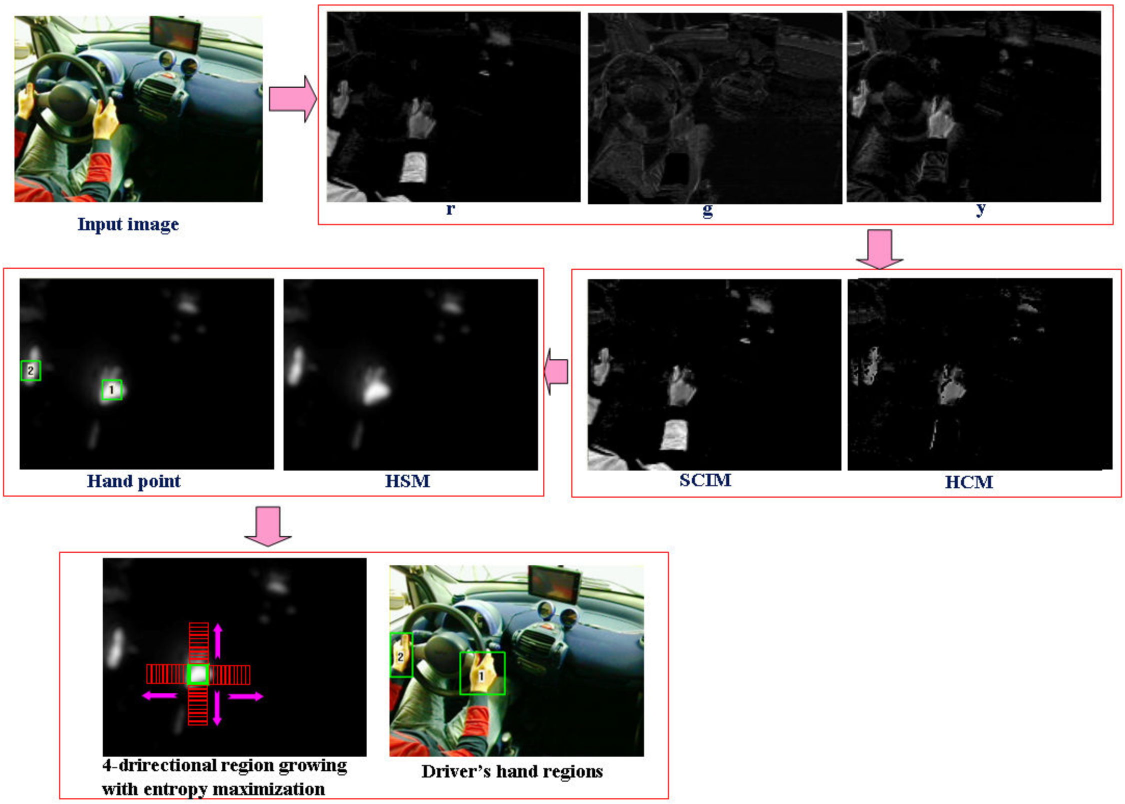


Fig. 5. The procedure of extracting hand color map

local maximum energy with fixed window size by shifting the pixels the RTS_SM as shown Fig. 4.

Each candidate road traffic sign region, which is selected in the RTS_SM, may have different size. In the proposed model, an entropy maximization approach based on Kadir's approach was adapted to select a proper scale of road traffic sign region in the RTS_SM [10, 11, 12].

As shown the Fig.4, the proposed model can simply select the road traffic regions and get a proper scale of a salient area

B. *Driver's Hands Detection*

In order to detect the driver's hands, the r, g, y color feature are biased. The normalized red(r), green(g), and yellow(y) color features are extracted from the R, G, B. The skin color intensified map(SCIM) is obtained through Eq. (6).

$$SCIM = \begin{cases} (r+g)-2\times g, & 0 < SCIM < 255 \\ 0 & ,SCIM < 0 \\ 255 & ,SCIM > 255 \end{cases} \tag{6}$$

And, the hand color map(HCM) is extracted form the skin color intensified map with skin color range filter. Then, the hand saliency map(HSM) is constructed by the Gaussian pyramid processing and CSD&N algorithm like as Eq.(7).

$$\overline{HSM} = \bigoplus_{c=2}^{3} \bigoplus_{s=c+2}^{3} N(SCIM(c,s)) \tag{7}$$

After constructing the HSM, the candidate driver's hand regions are simply selected by local search for maximum energy with constrain of energy threshold. Then, the proper scale of driver's hand regions are getting from four-directional region growing with entropy maximization algorithm and driver's hand constrains which are size and magnitude of candidate driver's hand regions .

Fig. 5 shows the experimental results of the proposed driver's hands detection model. In here, normalized r, g, and y are considered as input features for making HCIM. Then, the HCM is extracted by skin color filtering process for eliminating background. And, the HSM is consisted through CSD&N. And, the candidate region of road traffic sign boards is selected through searching local maximum point with four-directional based entropy maximization algorithm. The proposed model presents lager salient value in driver's hand area than another area. Moreover, we can get a proper scale of driver's hand regions in the HCM.

C. *Driver's Head Detection and Measuring of Visual Behavior*

As a previous work, we have proposed face selective attention and got successfully result in the complex scene[11]. And, because In-vehicle environments are relatively simple and face regions are more dominant then other regions in the vehicle, we can more simplify this model to apply for real time system.

In order to detect driver's head, the r, g, y color feature are

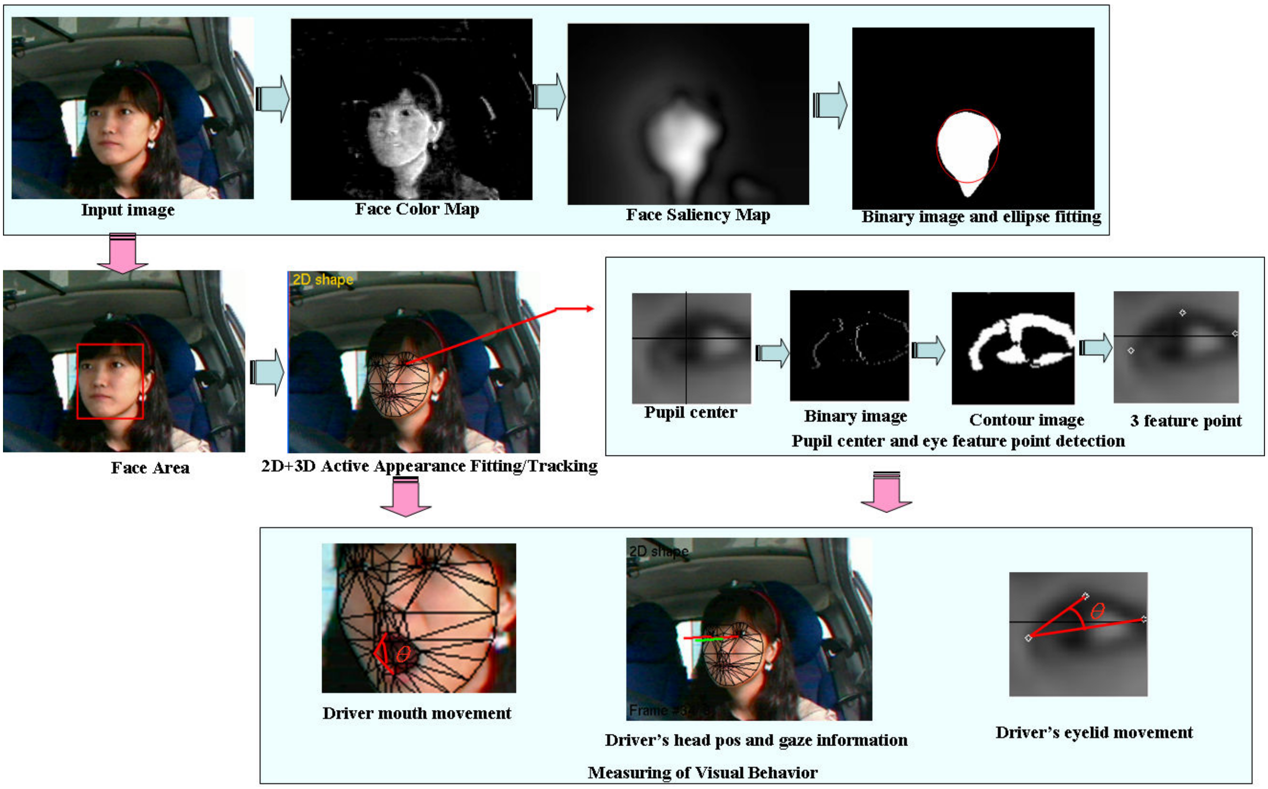


Fig. 6. Overall procedure in experimental results for the proposed driver's face detection and visual behavior monitoring model

biased. Then, The normalized red(r), green(g), and yellow(y) color features are extracted from the R, G, B. The skin color intensified map(SCIM) is extracted from r, g, y which is obtained through Eq. (6). And, the Face color map(FCM) is extracted from the skin color intensified map with skin color filter. Then, the face saliency map(FSM) is constructed by the Gaussian pyramid processing and CSD&N algorithm. After constructing the FSM, the candidate driver's face regions are segmented by Otsu's threshold method and labeling with constraint of size and ellipse shape. And the adaptive size of driver's head is getting from histogram projection method. Then, the 2D+3D AAM fitting occur in selected face area to get driver's 3D head pos and gaze information[13, 14].

Fig. 6 shows the overall procedure in experimental results for the proposed driver's head detection and visual behavior monitoring method.

III. Conclusion

We proposed a new in/out-vehicle monitoring model for interactive safety driving agent system based on modified selective attention model. In order to detect the road traffic sign regions for out-vehicle environment, we consider traffic sign color contrast characteristic against environment with CSD&N algorithms, And, for the detecting driver's head and hand, we also more intensify driver's hand regions then background and reduce background noise as considering saliency map. Moreover, we can get simple proper scale information of detected regions as intensifying target objects from the input image

As further works, we are considering road traffic recognition model for selected regions, and detection model of obstacle and lane for monitoring out-vehicle environment. And, we will develop estimating model for the driver's inattentive behavior and intent using measured driver's hand and visual movement.

Acknowledgment

This work was supported by the Deagu Gyeongbuk institute of science and technology (DGIST) Basic Research Program of the MOST.

References

[1] L. R. Shelton, "Statement before the subcommittee on highways and transit, committee on transportation and infrastructure", U. S hous of representatives, May 9, 2001.http://www.nhtsa.dot.gov/nhtsa/announce/.
[2] J . C. Stutts, D. W. Reinfurt, L. S. Staplin,E. and A. Rodgman, "The role driver distraction in traffic crashes," ,Report Prepared for AAA Foundation for Traffic Safety, June 10, 2003.
[3] J. R. Treat, N.S. Tumbas, S. T. McDonald, D. Shinar, R.D. Hume, R.E. Mayer, R. L Stansifer, and N. J. Castellan, "Tri-Level study of the Causes of Traffic Accidents: Final Report-Executive Summary," Department of Transportation, National Highway Traffic Safety Administration, Washington D.C , Tech. Rep. DOT HS805 099, 1979.
[4] S. Lyman, s. A. Ferguson. E.R. Braver, and A. F. Williams, "Older driver involvements in police reported crashes and fatal crashes: trends and projections," *Injury Prevention*, vol 8, no. 2, pp.116-120, 2002.
[5] E. Rochtchina. B. N. Mukesh, J. J. Wang, C. A. McCarty, H. R. Taylor, and P. Mitchell, "Projected prevalence of age-related cataract and cataract surgery in Australia for the years 2001 and 2021: pooled data from two population based surveys," *Clinical and Experimental Ophthalmology*, vol. 31, no. 3, pp. 233-236, 2003.

[6] P. Green, "Driver Distraction, Telematics Design, and Workload Managers Safety Issues and Solution," *in Proc. The 2004 International Congress on Transportation Electronics*, Warrendale, 2004, pp.165-180.

[7] SAVE_IT, Safety Vehicle(s) using adaptive Interface Technology (SAVE_IT) Program, Volpe National Trasnportation Systems Center. Available: http://www.volpe.dot.gov/hf/roadway/saveit/.

[8] T. V. Helene, B. Thirry, B. Serge, and etc, "Development of a Driver Situation Assessment Module in the AIDE Project," *in Proc. IFAC 16th World Congress*, Czech Republic, 2005.

[9] W. J. Won, S. M. Jeong, and M. Lee, "Road Traffic Sign Saliency Map Model," *in Proc IVCNZ2007,* 2007, pp. 91-96.

[10] W. J. Won and J. Y. Yeo, S. W. Ban, M. Lee, "Biologically Motivated Incremental Object Perception Based on Selective Attention", *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 21, no. 8, pp. 1293 -1305, 2007.

[11] W. J. Won, Y. M. Jang, S. W. Ban, and M. Lee, "Biologically Motivated Face Selective Attention Model," *in Proc ICONIP2007,* 2007.

[12] T. Kadir and M. Brady, "Scale, saliency and image description", *International Journal of Computer Vision*, vol. 45, pp. 83 -105, 2001.

[13] I. Matthes, J. Xiao, and S. Baker, "On the Dimensionality of Deformable Face Models," tech. report CMU-RI-TR-06-12, Robotics Institute, Carnegie Mellon University, 2006.

[14] T. Ishikawa, S. Barker, I. Matthews, and T. Kanade, "Passive Driver Gaze Tracking with Active Appearance Models," *in Proc. 11th World Congress on Intelligent Vehicle System*, 2004.

[15] Z. H. Zhou, X. Geng, "Projection Functions for Eye detection," *Pattern Recognition*, vol. 37, 2004, pp. 1049-1056.

[16] V. Vezhnevets and A. Degtiareva, "Robust and Accurate eye Contour Extraction," *in Proc. GRAPHICON*, 2003, pp.81-84.

A Subspace Method Based on Data Generation Model with Class Information

School of Electrical Engineering and Computer Science,
Kyungpook National University, Korea
Minkook Cho,
Hyeyoung Park

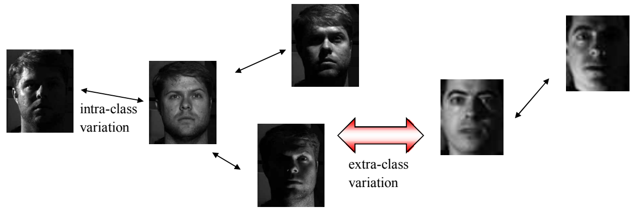
Contents

Introduction

- Subspace method : low dimensional features of data
- For what?
  - Reduce capacity of memory or complexity of system
  - Expect to increase classification performance
- Proposed method
  - Based on data generation model
  - Undercut the effect of intra-class information
  - Enlarge the effect of extra-class information

Data Generation Model

- Consider that we obtained several images from different persons
  - Pictures of different persons are obviously different
  - Pictures of a person are not exactly same
    - Some environmental condition such as illumination

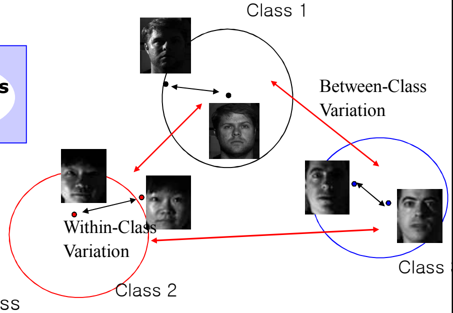


Intra-Class Factor and Extra-Class Factor

- Assumption on data

  DATA
  intra-class factor + **extra-class factor**

  - The intra-class factor
    - Represent within-class variations
  - The extra-class factor
    - Represent between-class variations



Function Form of Data Generation Model

- Function form

$$\mathbf{x} = f(\xi, \eta)$$

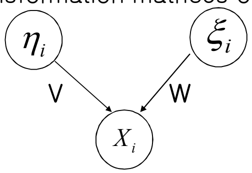
  - ξ : extra-class factor
    - Keeps some unique information in each class
  - η : intra-class factor
    - Represents environmental variation in one class

Linear Data Generation Model

- Apply a linear additive factor model

$$\mathbf{x}_i = f(\xi_i, \eta_i) = \mathbf{W}\xi_i + \mathbf{V}\eta_i$$

  - $\mathbf{x}_i$: a data in class C_i
  - ξ_i : discriminative information among class
  - η_i : some variation in class C_i
  - $\mathbf{W}$, $\mathbf{V}$: transformation matrices of corresponding factors



How to obtain the subspace?

- Goal :
  - Small intra-class variance, large extra-class variance
- Estimation of intra-class information
  - Define new random variable : difference vector δ between two data from same class

$$\delta_{kij} = \mathbf{x}_{ki} - \mathbf{x}_{kj}$$
$$= \left(\mathbf{W}\xi_{ki} - \mathbf{W}\xi_{kj}\right) + \left(\mathbf{V}\eta_{ki} - \mathbf{V}\eta_{kj}\right)$$

    - Two data are came from the same class -> the extra-class factor does not make much difference
    - We may ignore the first term

Maximize Intra-Class Information

- Approximate relationship

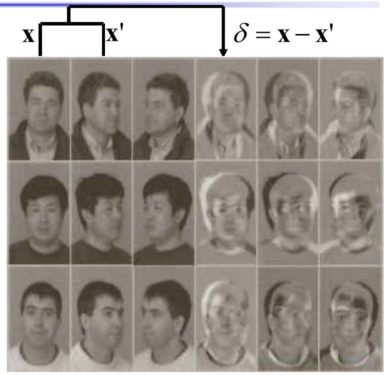
$$\delta_{kij} \approx \mathbf{V}(\eta_{ki} - \eta_{kj})$$

$$\Delta = \{\delta_{kij}\}_{k=1,\ldots,K,\, i=1,\ldots,N,\, j=1,\ldots,N}$$

- Apply PCA to Δ

$$(\Delta^T \Delta)\Lambda = \lambda\Lambda$$

  - Λ gives the basis of the subspace which maximizes the intra-class variation.



Intra-Class Information

- Project the original data $\mathbf{X}$ to the subspace of Λ

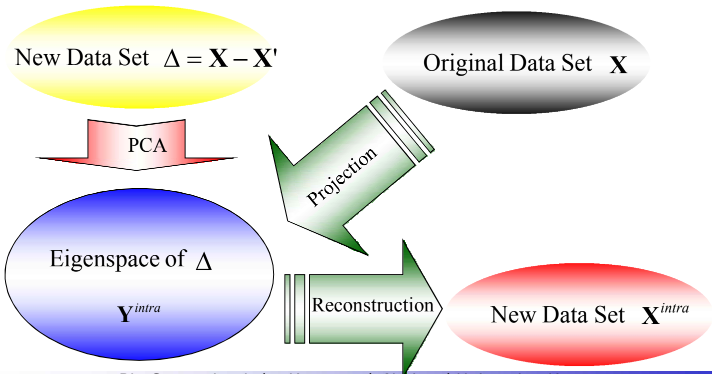
$$\mathbf{Y}^{intra} = \mathbf{X}\Lambda$$

  - Note : $\mathbf{Y}^{intra}$
    - Low dimensional data set
    - Include the intra-class information of $\mathbf{X}$
- Reconstruction to $\mathbf{X}^{intra}$ in original dimension

$$\mathbf{X}^{intra} = \mathbf{Y}^{intra}\Lambda^T(\Lambda\Lambda^T)^{-1}$$

  - Note : $\mathbf{X}^{intra}$
    - The intra-class information in original dimension
    - The undesirable information for classification

How to get Intra-Class Information



Extra-Class Information

- Remove the undesirable information from $\mathbf{X}$

$$\tilde{\mathbf{X}} = \mathbf{X} - \mathbf{X}^{intra}$$

- $\mathbf{x}^{intra}$ in data set $\mathbf{X}^{intra}$
  - Reconstructed from the intra-class factor
- New approximate relationship

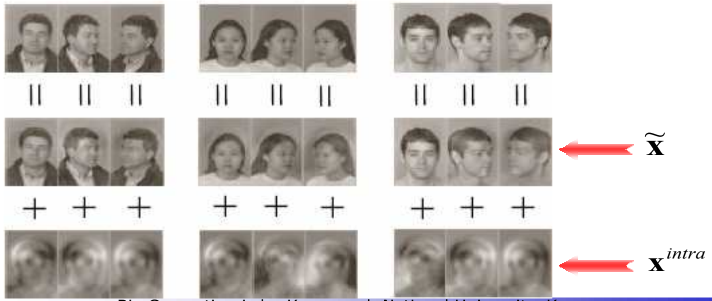
$$\mathbf{x}^{intra} \approx \mathbf{V}\eta$$

  - Combine it with data generation model

$$\tilde{\mathbf{x}} \approx \mathbf{x} - \mathbf{V}\eta = \mathbf{W}\xi$$

Sample Images of Proposed Data Generation Model

- $\tilde{\mathbf{X}}$ mainly has the extra-class information
  - Preserve the extra-class information as much as possible



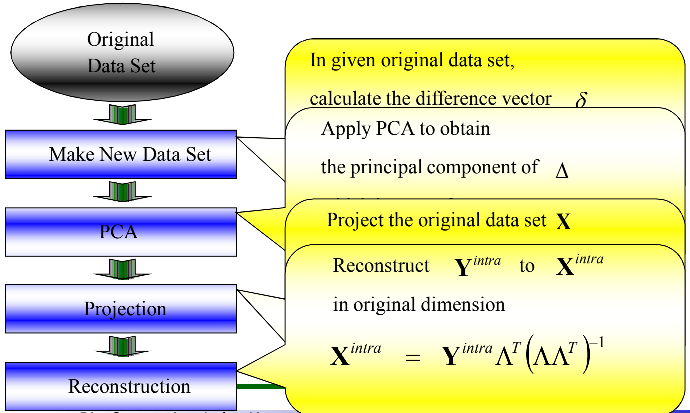
Maximize Extra-Class Information

- Apply PCA to the new data set $\tilde{\mathbf{X}}$

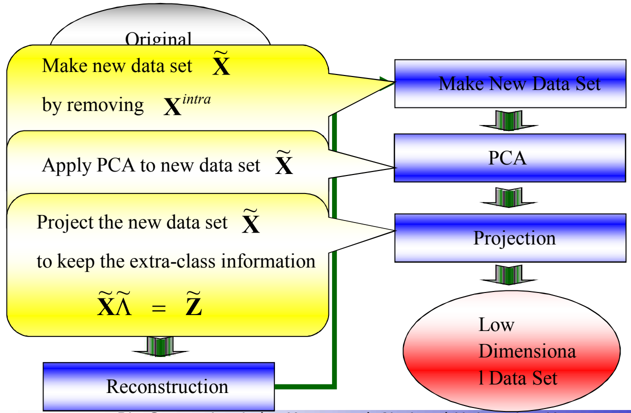
$$\tilde{\mathbf{Z}} = \tilde{\mathbf{X}}\tilde{\Lambda}$$

- Note : $\tilde{\mathbf{Z}}$
  - Intra-class variance is small
  - Extra-class variance is large

Flow Chart of the Proposed Method



Flow Chart of the Proposed Method



Experimental Results

- Experiment on facial database : FERET/PICS
  - Comparison with PCA, LDA and Intra_person
- Classification task
  - Face Classification on FERET
  - Pose Classification on FERET
  - Facial Expression Classification on PICS
  - Face Classification on PICS
- Classification method
  - Minimum Euclidean distance method

Database

- FERET(http://www.itl.nist.gov/iad/humanid/feret/)
  - 450 images from 50 subjects
  - Each subject has 9 images of different poses
  - 70 x 50 size

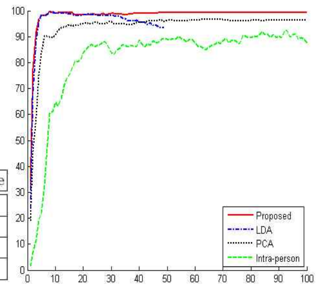
Database

- Psychological Image Collection at Stirling (http://pics.psych.stir.ac.uk/)
  - 276 images from 69 subjects
  - Each subject has 4 images of different expressions
  - 90 x 80 size

Face Classification on FERET

- 50-class classification
  - Training : 3 data from each class
  - Testing : 6 data from each class

Method	Dimension	Classification Rate
PCA	117	97
LDA	9	99.66
Intra-Person	92	92.33
Proposed	8	$\boxed{100}$

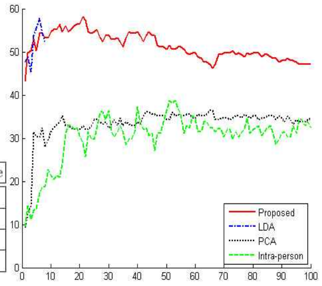


Basis : not depend on a number of class
Robust in various subspace

Pose Classification on FERET

- 9-class classification
  - Training : 25 data from each class
  - Testing : 25 data from each class

Method	Dimension	Classification Rate
PCA	65	36.44
LDA	6	57.78
Intra-Person	51	38.67
Proposed	21	$\boxed{58.22}$

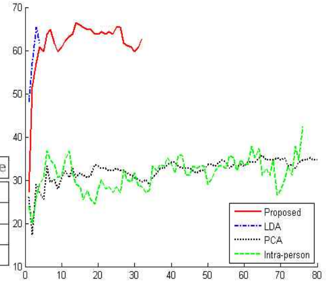


Variance of between-class is very small

Facial Expression Classification on PICS

- 4-class classification
  - Training : 20 data from each class
  - Testing : rest 196 data

Method	Dimension	Classification Rate
PCA	65	35.71
LDA	3	65.31
Intra-Person	76	42.35
Proposed	14	$\boxed{66.33}$

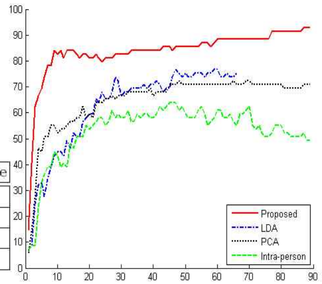


Variance of between-class is very small

Face Classification on PICS

- 69-class classification
  - Training : 3 data from each class
  - Testing : rest 69 data

Method	Dimension	Classification Rate
PCA	48	72.46
LDA	47	76.81
Intra-Person	45	63.77
Proposed	87	$\boxed{92.75}$



Number of training data is very limited

Conclusions

- New subspace method based on data generation model with class information
- Obtained subspace : small intra-class variance, large extra-class variance
- Advantages of proposed method
  - No need to compute a inverse matrix
  - Simple to get subspaces
  - The number of basis of subspace does not depend on a number of class
  - Extensions of the proposed method are possible, through more general data generation model

Further Study

- Further study
  - Generalization of data generation model
    - Ex) class-dependant transformation matrix W
  - Extension of dimension reduction method
    - Ex) Kernel method for the non-linearity projection

Thank you for your attention Q&A

Management Method for Concurrent Digital Photos using EXIF Metadata

GRAVIKON 2008
Chuljin Jang
Graphics Application Lab.
Pusan National University

Motivation

- Popularization of digital camera
  - A number of photos generated
    - Very low cost to take picture compared to film camera
    - Convenience, cheap memory, camera equipped phone, etc.
    - Need to classified according to users' preference
  - Several cameras can be used in an event
    - Each camera can have photos of same object
    - Sometimes people take picture each other with their own camera

-> We need managing method/system for digital photos.

Our goal

- clustering concurrent photos
  - concurrent photos are all collected photos taken by few tourist on a well-known sightseeing places (e.g. Eiffel Tower, Pyramid)
  - Concurrent photos share their contents
  - Everyone has their own preference to managing photos
  - Our goal is to cluster all photos according to a specific photographer's preference

Previous work (1/2)

- Digital photo clustering
  - clustering private photos mainly from one digital camera
  - clustering by using content and temporal data
  - using metadata(EXIF)
    - EXIF includes various information about picturing environment

Previous work (2/2)

- clustering using temporal and spatial features

a sequence of photos given → after arrangement

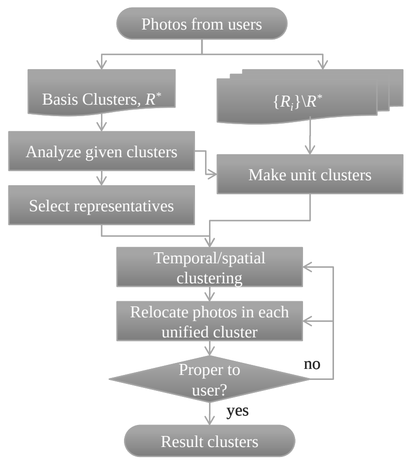
Definitions

- Private photo album
  - a set of photos obtained from one digital camera photographed by an ordinary user
- Photo roll
  - a set of photos taken specialized period which is subset of private photo album
- Concurrent photos
  - a set of photo rolls taken by a group of digital cameras I a specifi ed region on a specified time period

Interesting problems on concurrent photos

- Issues on concurrent photos
  - How to compensate differences caused by camera itself
    - Each camera has different resolution, tone of color, zoom ratio
  - How to adjust timestamp differences between cameras
    - Each camera of group members' can have time differences compare d to the others
    - Some member can go ahead, and this cause time gap between them
  - How to find similar photos
    - Photos showing different content can be taken concurrently
    - Similar photos can be scattered around time space
    - -> difficult to cluster using only temporal analysis with time gap

Overview of clustering method



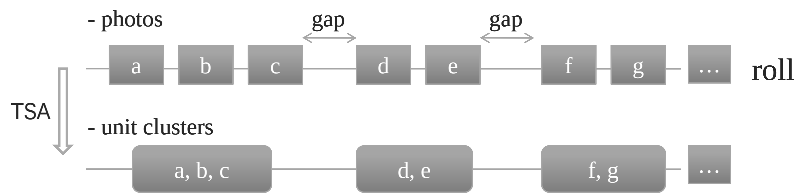
- R is photo roll
- We are given clustered R* as a preferred photo roll which is specified by a user
- We extract features from basis Cluster R*, then clust er other rolls according to them

Problem of Concurrent Photo Clustering (PCPC)

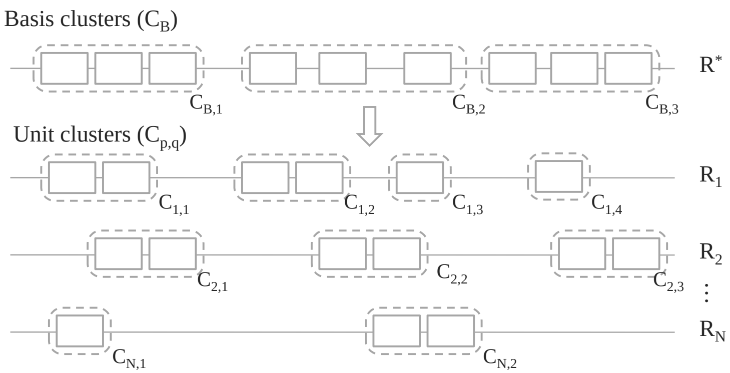
- PCPC
  - o Concurrent photos, CP = $\{R_i\}$
  - o R* is basis roll indicated by a user
  - o Extracting the preference feature set $\{F_i\}$ from R*
    - F_i <= Character(R*)
  - o $\{F_i\}$ includes
    - time interval of one photo cluster specified
    - mean, median, differences and standard deviation of time-sta mps among all photos
    - longest time interval between neighbor photos
    - spatial similarities between photos in a cluster

Generating unit clusters by temporal analysis

- To cluster photo set U
  - o U = $\{R_i\} \setminus R^*$
  - o Cluster photo set U by using $\{F_i\}$
  - o We apply Temporal Similarity Analysis (TSA) to preprocess input photo rolls



Unit clusters obtained from TSA



Refining basic clusters

- Basis clusters from R*
  - o Need to compare basis clusters with the unit clusters of R_i
  - o If one cluster has many similar photos
    - the overall similarity of two clusters is getting lower
  - o Refining basis clusters by discarding photos which can be an obst acle in comparison
- Refining basis clusters
  - o Selecting representative photos by applying mutually N-nearest sub-graph (discarding duplicated photos)
  - o Reserve unique photos in a basis cluster which are important to distinguish cluster

Refining algorithm

Algorithm 1 Refining Basis Clusters

procedure REFINE_BASISCLUSTER($C_{B,x}$)
  R_h = GetIntraHighInterLow($C_{B,x}$)
  R_l = GetIntraLowInterLow($C_{B,x}$)
  $R = R_h \bigcup R_l$
  return R
end procedure

procedure GETINTRAHIGHINTERLOW($C_{B,x}$)
  for $k = 1$ to N **do**
    Build adjacency graph G_k^* from $C_{B,x}$ by linking edges between mutually k-nearest photos
    Compute intra/inter cluster similarity S_i, S_o
    $E_k = S_i \ / \ S_o$
  end for
  return vertices having maximum weight W at each subgraph of G_k^* having maximum E_k
end procedure

procedure GETINTRALOWINTERLOW($C_{B,x}$)
  for $k = 1$ to N **do**
    Build inadjacency graph G_k^+ from $C_{B,x}$ by linking edges between mutually k-farthest photos
    Compute intra/inter cluster similarity S_i, S_o
    $E_k = S_i \ + \ S_o$
  end for
  return $V(G_k^+)$ of G_k^+ having minimum E_k
end procedure

- similarity

$$S_i(G) = \frac{1}{n(G)(n(G)-1)} \sum_{p_i \in G} \sum_{p_j \in G, p_i \neq p_j} S_s(p_i, p_j)$$

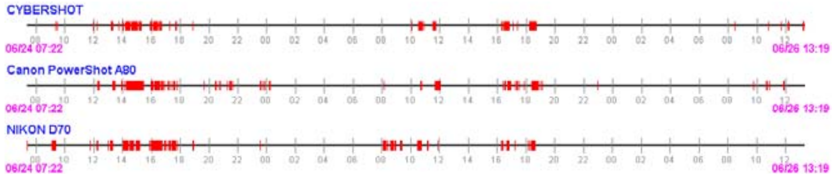
$$S_o(G) = \frac{1}{n(G)(n(P)-n(C_B))} \sum_{p_i \in G} \sum_{p_j \notin C_B} S_s(p_i, p_j)$$

- weight sum W

$$W(p_x) = \sum_{p_i \in N(p_x)} w(p_x, p_i)$$

Temporal / Spatial combined clustering

- Temporal similarity
  - Input data are photos sharing activities



  - Even though we use multiple cameras, the overall temporal coherence is maintained
  - We can apply the both of spatial and temporal analysis to compar e clusters

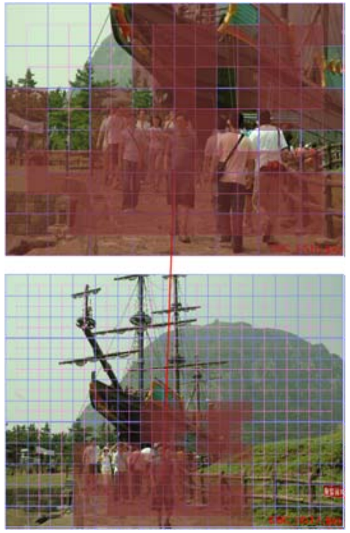
$$K(C_{B,i}) = \alpha(\ interval(C_{B,i}) + dist_L(C_{B,i}))$$

$$S_t(C_{B,i}, C_{p,q}) = \exp\left(-\frac{|ts(C_{B,i}) - ts(C_{p,q})|}{K(C_{B,i})}\right)$$

Temporal / Spatial combined clustering

- Spatial similarity
  - Use block expansion based on tessellated unit blocks.
  - Find N-best match seeds, then expands similar block starting from seed block
  - temporal and spatial cluster ing S_C
    - ratio can be adjusted by a user

$$S_c = t \cdot S_t + (1-t)S_s, \quad (0 \leq t \leq 1)$$



Integrating unit clusters

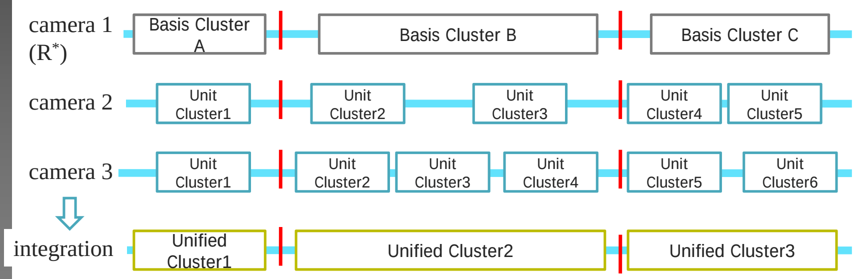
- Through Algorithm 2, we can get unified clusters

Algorithm 2 : Integrating Clusters

procedure INTEGRATECLUSTERS
  Initialize a set of integrated clusters C_R^* using basis clusters C_B
  for each unit cluster $C_{i,j}$ in U **do**
    Find maximum S_c by comparing unit cluster with each basis cluster $C_{B,k}$
    if $S_c > threshold$ **then**
      Integrate the unit cluster into $C_{R,k}^*$
    else
      if maximum S_c over *threshold* is found by comparing with each $C_{R,l}^+$ then merge it.
      Otherwise, create a new cluster in C_R^+
    end if
  end for
  $C_R = C_R^* \bigcup C_R^+$
  Sort clusters of C_R by temporal order.
end procedure

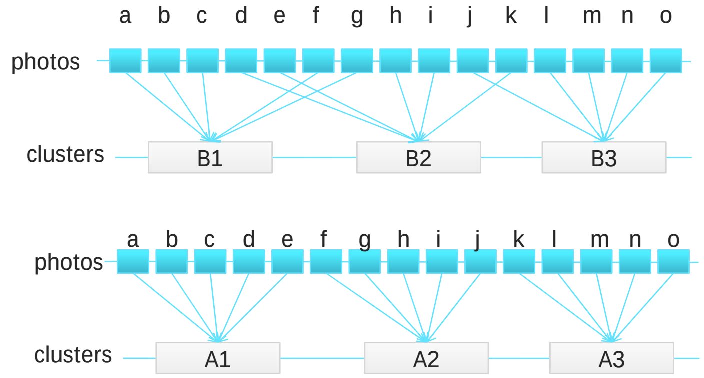
Unified clusters

- Integrating the basis and unit clusters
  - into unified clusters by using combined temporal/spatial clustering

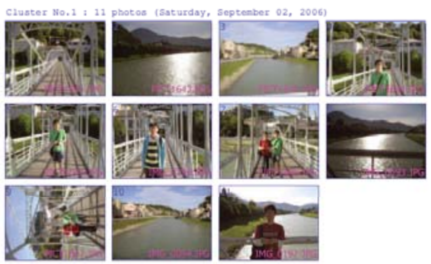


Contextual flow in unified clusters

- Previous clustering method keeps the temporal order
- We rearrange photos across the unit clusters according to its spatial similarity if necessary



Rearranged photos



- Photos of a unified cluster ordered by timestamp

- The rearranged photos of a unified cluster

Clustering result

- Result of 4 concurrent photo set

Tester/Method		CP1	CP2	CP3	CP4
TSA (fixed time gap)	K=100	0.727	0.703	0.848	0.685
	K=500	0.638	0.653	0.848	0.648
	K=1000	0.450	0.656	0.848	0.699
Person A	HSV	0.773	0.794	0.916	0.974
	CIELuv	0.865	0.812	0.937	0.979
Person B	HSV	0.876	0.834	0.896	0.977
	CIELuv	0.811	0.767	0.862	0.962
Person C	HSV	0.869	0.882	0.928	0.931
	CIELuv	0.876	0.817	0.865	0.924
Person D	HSV	0.792	0.848	0.904	0.890
	CIELuv	0.839	0.871	0.920	0.879
Person E	HSV	0.749	0.846	0.910	0.918
	CIELuv	0.761	0.857	0.892	0.893

- Evaluation measure

$$\frac{\sum_i^{|C_R^*|} good(P_i, C_{R,i}^*) + \sum_j^{|C_R^+|} extra(P_j, C_{R,j}^+)}{n(CP)}$$

Snapshot of result clusters

- Clustered concurrent photos

Conclusion & Future work

- We proposed clustering algorithms for a set of concurrent photos classifies according to the features of given basis cluster
- Our system clusters concurrent photo rolls to satisfy the user's pr eference hidden in the basis cluster
- We expect more smart personally customized clustering will be n eeded to confront the explosion of digital images
- Clustering by persons shown in photo will be very useful in the n ear future
- Geographical information based clustering will be possible with GPS equipped digital camera

References

1. ACDSee: http://www.acdseee.com/
2. Boutell, M., Luo, J.: Bayesian fusion of camera metadata cues in semantic scene classification. In: CVPR '04: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 623–630 (2004)
3. Boutell, M., Luo, J.: Photo classification by integrating image content and camera metadata. In: ICPR '04: Proceedings of the 17th International Conference on Pattern Recognition, vol. 4, pp. 901–904 (2004)
4. Chen, J.C., Chu, W.T., Kuo, J.H., Weng, C.Y., Wu, J.L.: Tiling slideshow. In: MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia, pp. 25–34 (2006)
5. Chu, W.T., Chen, J.C., Wu, J.L.: Tiling slideshow: An audiovisual presentation method for consumer photos. IEEE Multimedia **14**(3), 36–45 (2007)
6. Cooper, M., Foote, J., Girgensohn, A., Wilcox, L.: Temporal event clustering for digital photo collections. ACM Trans. Multimedia Comput. Commun. Appl. **1**(3), 269–288 (2005)
7. Graham, A., Garcia-Molina, H., Paepcke, A., Winograd, T.: Time as essence for photo browsing through personal digital libraries. In: JCDL '02: Proceedings of the Joint Conference on Digital Libraries, pp. 326–335 (2002)
8. Jang, C.J., Lee, J.Y., Lee, J.W., Cho, H.G.: Smart management system for digital photographs using temporal and spatial features with exif metadata. In: ICDIM '07: Proceedings of the 2007 IEEE International Conference on Digital Information Management, pp. 555–558 (2007)
9. JEIDA: Digital still camera image file format standard (1998)
10. Loui, A., Savakis, A.: Automatic event clustering and quality screening of comsumer pictures for digital albuming. Multimedia, IEEE Transactions on **5**(3), 390–402 (2003)
11. Mills, T.J., Pye, D., Sinclair, D., Wood, K.R.: Shoebox: A digital photo management system. Tech. rep., AT&T Laboratories Cambridge (2000)
12. Naaman, M., Song, Y.J., Paepcke, A., Garcia-Molina, H.: Automatic organization for digital photographs with geographic coordinates. jcdl **00**, 53–62 (2004)
13. Picasa: http://picasa.google.com/
14. Platt, J.C.: Autoalbum: Clustering digital photographs using probabilistic model merging. cbaivl **00**, 96 (2000)

Autonomous Vehicle Detector using Spatial-Temporal Brightness Changes based on a Saliency Map for a Blind Spot

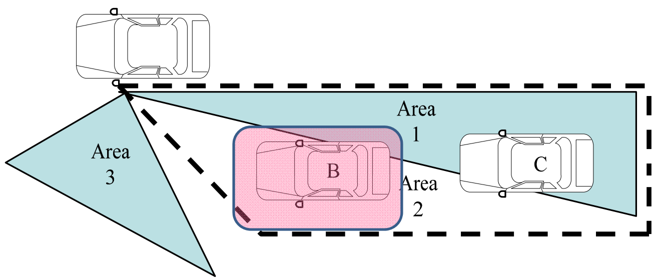
Sungmoon Jeong, Minho Lee*

Kyungpook National University, Deagu, Korea
Artificial Brain Research Lab.

Motivation

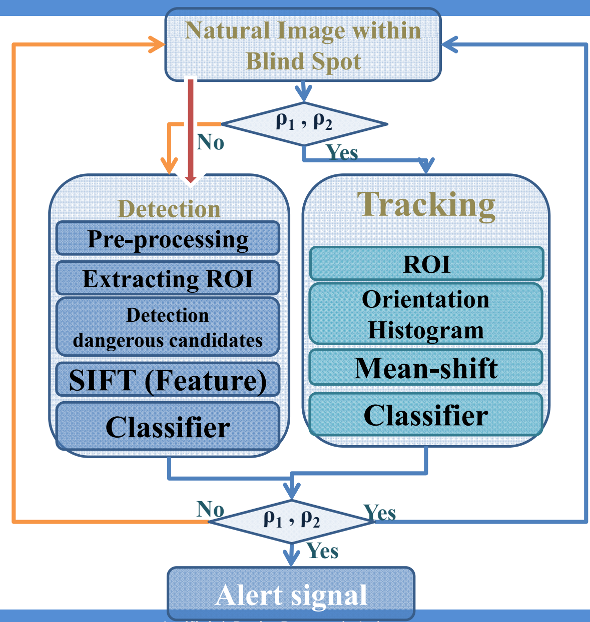
- The conventional auto safety technology was limited passive as seat belts and air bags.
- Car accidents mainly cause from the failure of drivers to stay within a lane.
  → Active safety systems will be required to alert the driver before a collision happens.
- Vision-based technology including smart sensing has been used in order to improve automotive safety.
  → Biologically motivated selective attention model and Scale Invariant Feature Extraction and mean shift tracking

Blind Spot & ROI



Blind spot area from a side-view camera (Area 2)

ROI in Blind spot area (Car 'B')



Pre-processing

Gaussian Kernel Function

$erosion: (f \ominus g)(x) = min\{f(z) - g_x(z) : z \in D[g_x]\}$

$dilation: (f \oplus g)(x) = max\{f(z) - g^*_x(z) : z \in D[g^*_x]\}$

f : gray image, g : strucuring element

$D[g]$: Domain of g

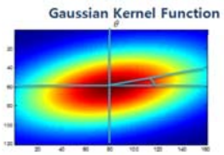
$closing: f \bullet g = -((-f) \circ (-g))$

$opening: f \circ g = (f \ominus g) \oplus g$

$top-hat \quad : f - (f \circ g)$

$bottom-hat: (f \bullet g) - f$

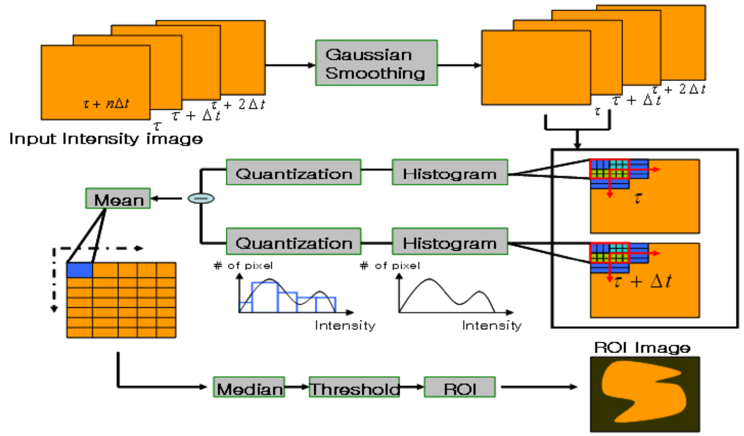
Top-hat , Bottom-hat Transform (morphology transform)



$\sigma_x > \sigma_y \quad , \quad \theta = 15^\circ$

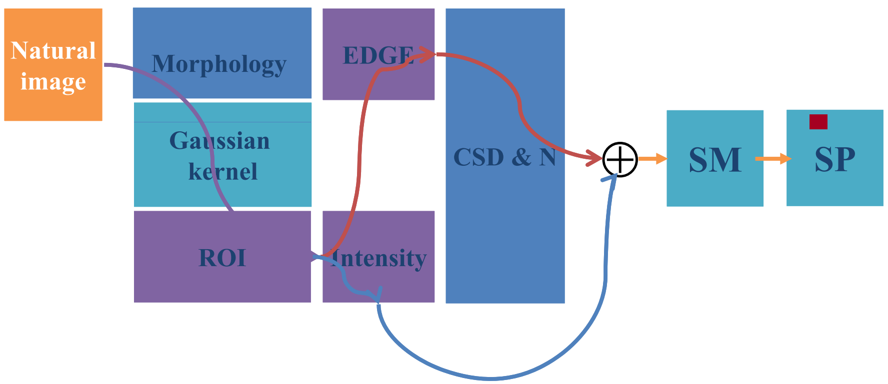
Parameter considering the way of dangerous points

Selection of ROI using differential histogram



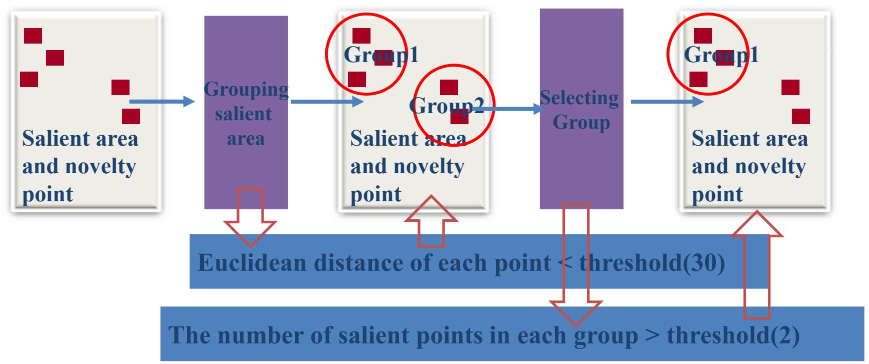
Feature Extraction in Image

The modified bottom – up saliency map model



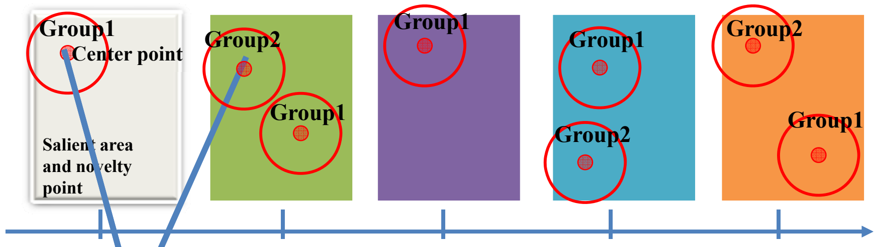
Feature Extraction in Image

Grouping salient points in a frame



Feature Extraction in Image

Detecting candidate dangerous points using topology of SP



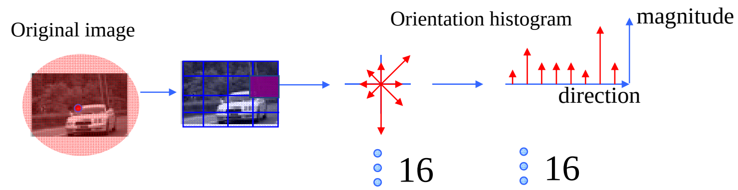
Euclidean distance of center points in successive frame < Threshold (20)

Successive satisfaction of condition == 4 → Alert!!!!!

Feature Extraction in Image

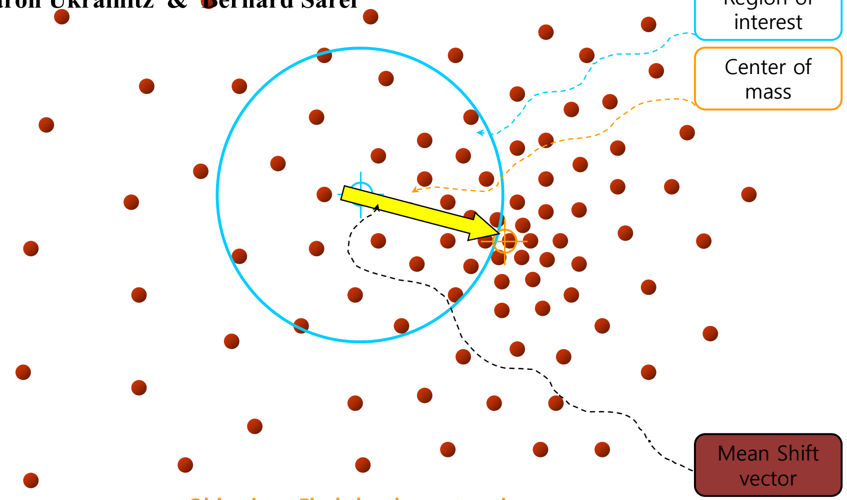
Detecting dangerous points using SIFT

Salient point descriptor



→ Number of subregion: 16 , size of region: 60 X 40
→ Direction: 8
→ Dimension of feature vector: 16 X 8 = 128

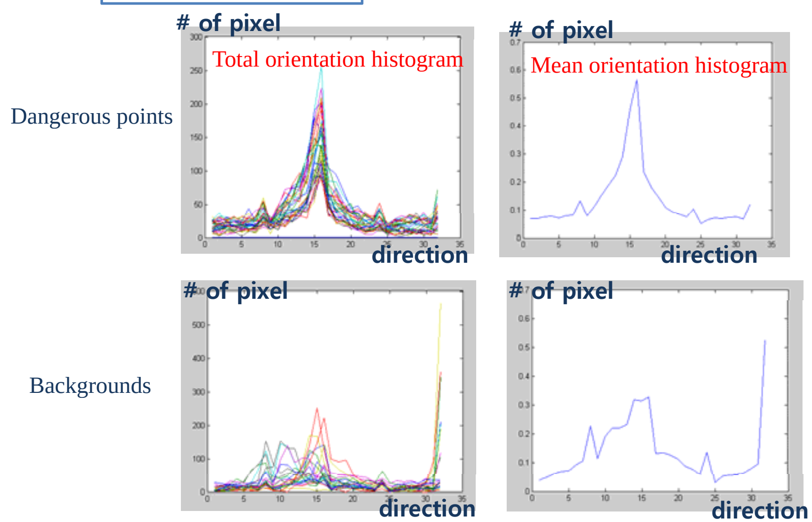
Intuitive Description of mean shift
"Yaron Ukrainitz & Bernard Sarel"



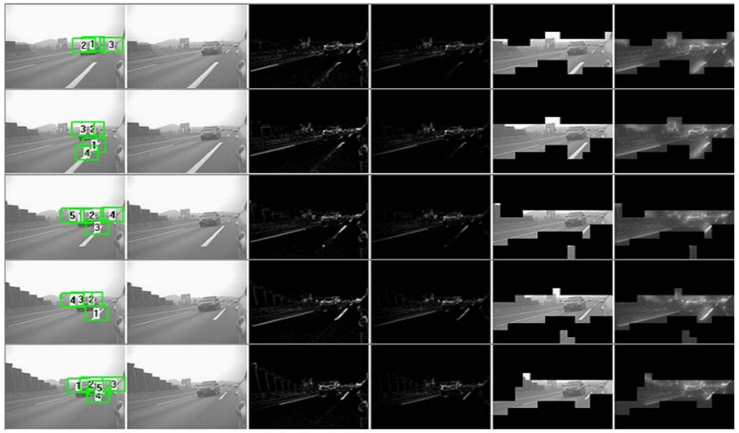
Objective : Find the densest region

Distribution of identical billiard balls

Orientation histogram



Computer simulation results



Salient points | Intensity image | Morphology transform image | Gaussian kernel function | ROI | Saliency map

Computer simulation results

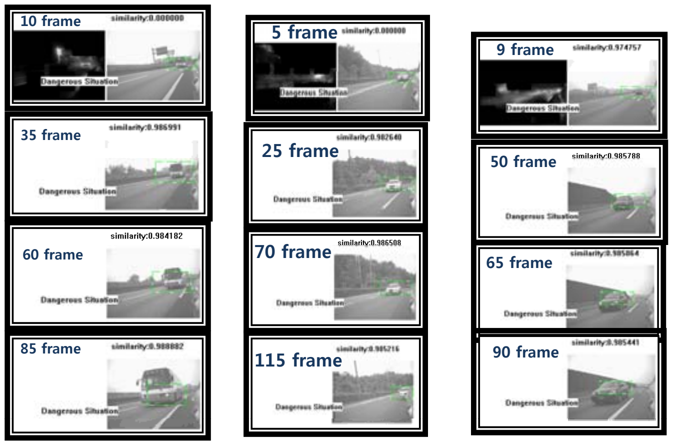
Using Saliency points topology

Using Saliency points topology + SIFT

	Not contain dangerous situation		Contain dangerous situation	
	Negative true	Negative false	Positive true	Positive false
Input Image	174(65%) 221(82%)	94(35%) 47(18%)	227(89%) 211(83%)	27(11%) 43(17%)

Computer simulation results

Tracking results



Conclusion & Further Work

- The proposed model can detect the dangerous situations in the ROI using novelty points from the biologically motivated selective attention model
- We proposed new algorithm compounding the saliency map model and SIFT model and mean shift model
- We reduce in most cases of false alarm performance but will consider the positive alarm performance
- Proposed algorithm apply to object tracking or motion detection
- Test with large data base and compare the other algorithm
- Considering more efficient selection of basis orientation histogram
- Considering computational load for real time processing

Emotional Scene Understanding Using GIST

Qing Zhang, Minho Lee

School of Electrical Engineering and Computer Science
Kyungpook National University

Abstract

- An emotion understanding system based on the gist of natural scene image is proposed.
- The machine will extract the emotional features of natural scenes and learn to discriminate positive and negative emotions.
- The experimental results show that the proposed algorithm can be used to recognize an emotional status.

Introduction

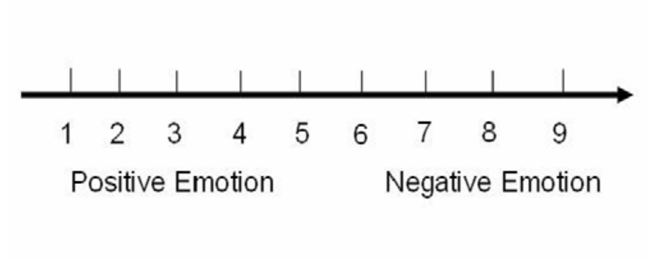
- Emotional factors help us to describe and simulate the human feedback of the natural scene.
- Most image processing methods ignore emotional factors.
- Facial expression recognition is a classic method to research emotion.
- Emotion cannot be exhaustively apprehended only using facial expression.
- Our motivation is to recognize the emotions in natural scene images.

Proposed Model

- We divide the natural scenes into positive and negative.
- According to the MOS, the “GIST” visually extracts features of natural scenes.
- The proposed system can recognize an emotional status by supervised learning.

Methods

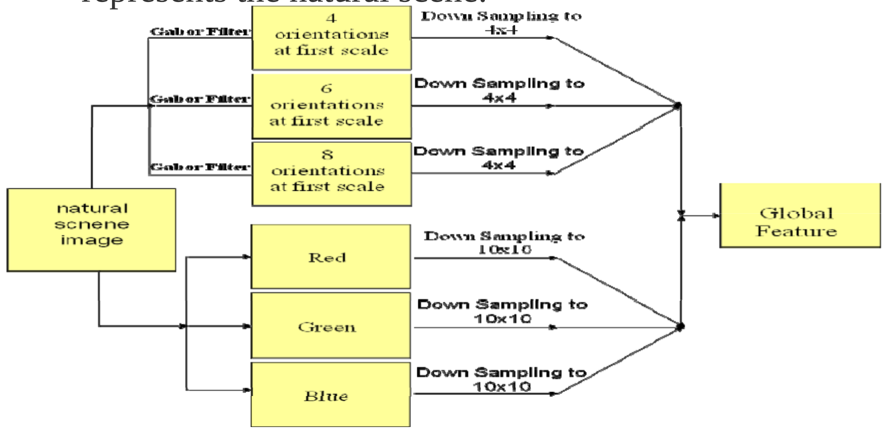
- Emotion Description by MOS



- Image Database is International Affective Picture System (IAPS)

GIST

- The "GIST" is used to form a global feature that represents the natural scene.

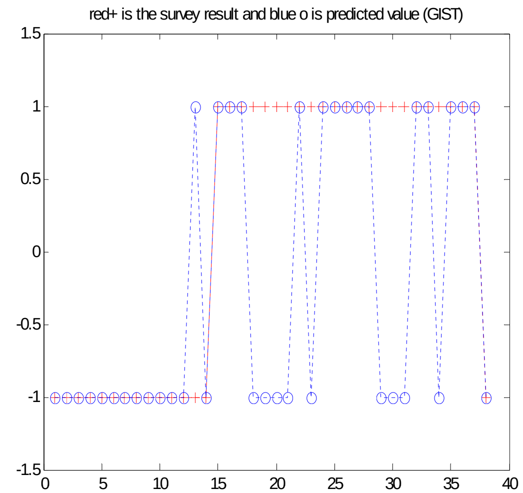


Classification

- In this work we employ linear support vector machine to perform classification. It is an supervised classifier to recognize natural scene as positive or negative category.

Experiment Results

- We exclude those most disputed natural scenes according to the MOS, and remove data with one-to-many mapping.
- Apply linear SVM classifier to classify different emotional natural scenes. The accuracy can achieve 73.68%.



1st Korean-New Zealand Workshop on Advances in Computer Graphics, Computer Vision and Virtual Reality (GRAVIKON 2008)

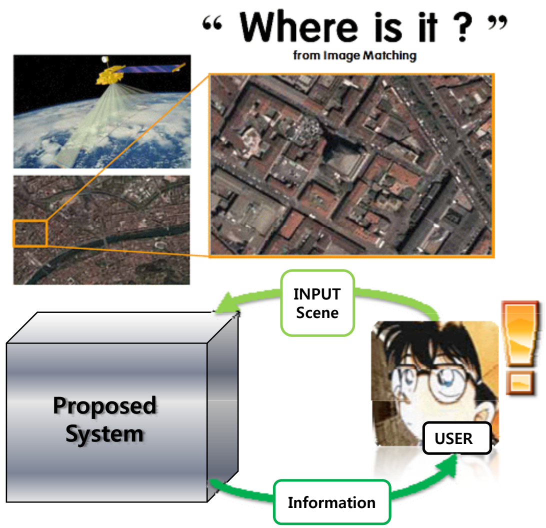
Planar-Object Position Estimation by using Scale & Affine Invariant Features

Seokjun Lee
sukjuni@vr.knu.ac.kr

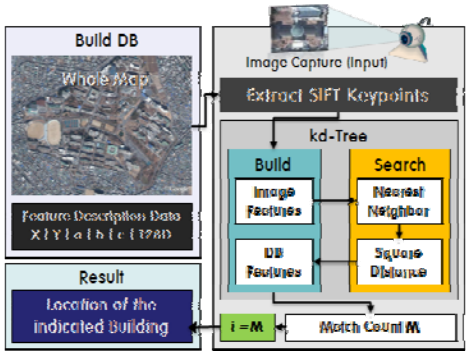
Soon Ki Jung
skjung@knu.ac.kr

Virtual Reality Laboratory
Dept. Computer Engineering in Kyungpook National University

motivation & concept



System Flow

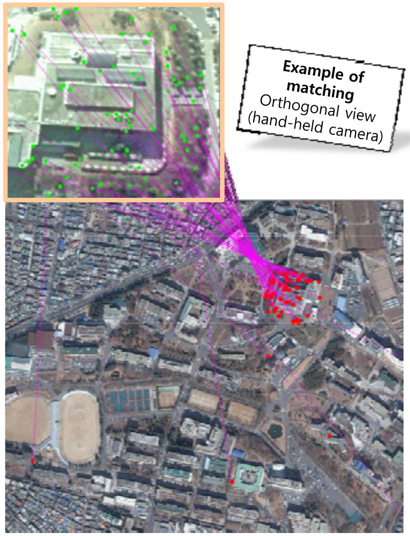


This system propose an experiments of image matching and recognition based on image feature detection and description techniques from artificial satellite photographs.

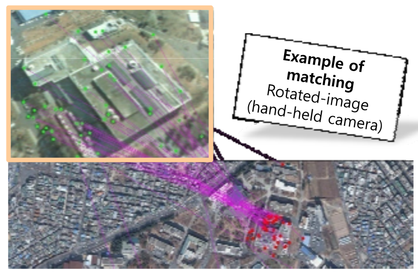
System equipment is pretty simple. We use PC-webcam for input scene of specific region or place from real whole MAP. The system extracts key point of pixel gradient from acquired image.

Key points are buildup a form of feature DB, And then, key points defined by feature as **2D coordinate** and **affinity** and **128-dimensional vector description**. This description data effects on matching process with pre-processed feature DB.

Key-point matching use by SIFT



- **SIFT**(Scale-Invariant Transform Feature) detector & descriptor
  - by David G. Lowe
- Criterion
  - Very robust : 80% Repeatability at
  - 10% image noise
  - 45° viewing angle
  - 1k-100k keypoints in database



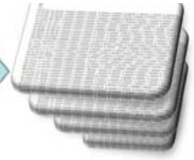
Build up a Feature DB

- **Construction process of Feature DB**
  - **TARGET Region**
    Artificial satellite imagery of ***KNU*** / Google earth
  ✓ *50* sample facilities in campus

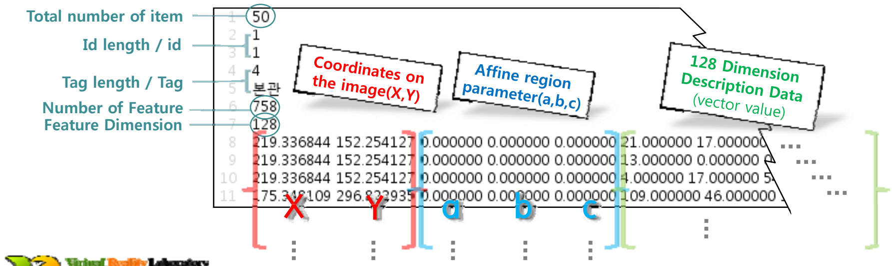
MAP of Specific region from Artificial Satellite

Identification
(50 facilities)

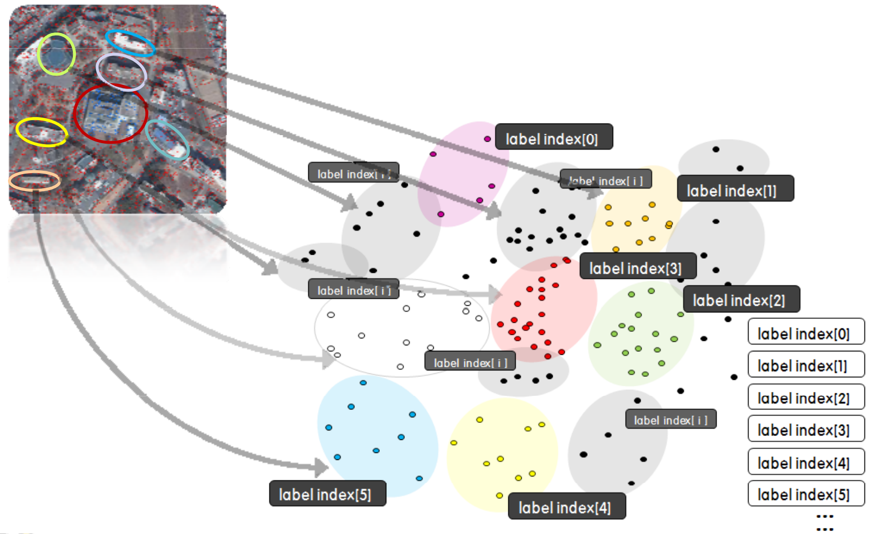
Archive a Feature Data Set
witch are represented by 128 dimensional vector description



- **Detail structure of Feature DB**



Key-point Clustering & Labeling (Identification)



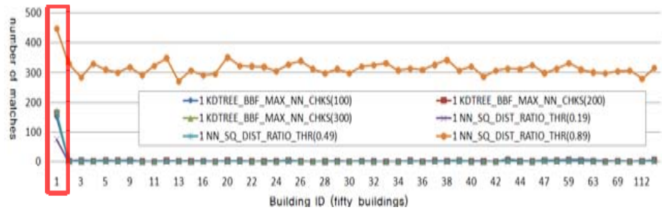
Result. Example of Building Recognition 1/2

- The result screen shot of 'central library' building in the camera scene**(left, pink points)** and matched region in the whole MAP**(right, blue points)**.
- 1280×999×24bit / 24,245 feature points
- Kyungpook National University, <Google EARTH>

Result. Example of Building Recognition 2/2

- **10 Result of building scene**

- **Response rate of search parameter changes**(query is 1st builing)



- **KDTREE_BBF_MAX_NN_CHKS**
  : KD-TREE Best Bin First MAX Nearest Neighbor Checks, The bigger parameter value is, the more need searching time.
- **NN_SQ_DIST_RATIO_THR**
  : Nearest Neighbor Square Distance Ratio Threshold, The bigger parameter value is, the less accuracy is.

Building Modeling System on Satellite Image using Footprint and Shadow

Seonho Oh, Jaeseok Jang, Kyungho Jang, Soonki Jung
Kyungpook Natl. Univ. Virtual Reality Lab

Introduction

- Building Modeling on Satellite
  - Multiple image
    - DEM by stereo algorithm, extracting 3D model
  - Image with sensor information
    - More accurate DEM
    - Difficulty of source data acquisition, cost problem, synchronize
  - Single Image
    - Using meta information and shadow, depends on manual operation
- We propose semi-automatic building modeling system on single satellite image using footprint and shadow

System overview

- System overview
  - Rooftop extraction
    - Closed polygon
    - Euler operator & Half-edge data structure
  - Shadow region extraction
    - HSI color transform & SRI
  - Height estimation
  - Texture mapping

Satellite Image & Meta Information → Rooftop Extraction → Shadow Extraction → Height Manipulation → Texture Mapping

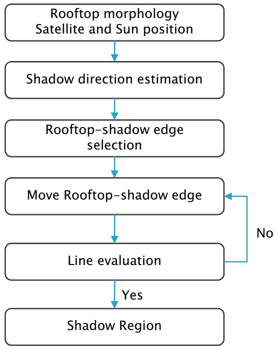
Building modeling

- Rooftop Extraction
  - User's manual operation
  - Using simple Primitives(Rect, Line, Circle, Arc)
  - Most buildings consists of composition of rectangular shapes
    - Add/remove rectangular shape(Lee[2000])
- Shadow region extraction
  - Properties of shadow
    - Lower intensity
    - Strong Blue-violet component, Hue value
  - SRI(Spectral Ratio Image) as shadowness
  - Getting SRI with HSI color transform

$$SRI(x,y) = \frac{Hue(x,y)+1}{Intensity(x,y)+1}$$

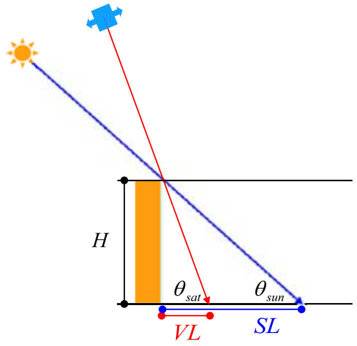
Building modeling

- Shadow region extraction
  - Rooftop-shadow edge selection
  - Extracting candidate shadow region from SRI image of rooftop and adjacent area
  - Applying threshold by histogram (Otsu[1979])
  - Edge-based region growing about shadow-rooftop edge



Building modeling

- Height estimation
  - Relationship between height of building and shadow

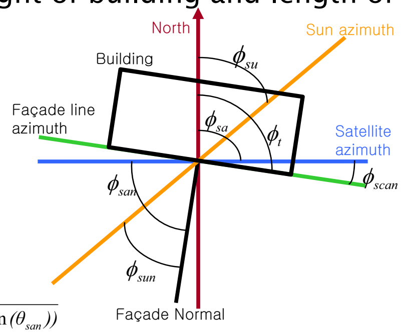


$$SL = H / \tan(\theta_{sun})$$
$$VL = H / \tan(\theta_{sat})$$
$$SL - VL = H\left(\frac{\tan(\theta_{sat}) - \tan(\theta_{sun})}{\tan(\theta_{sun})\tan(\theta_{sat})}\right)$$
$$H = (SL - VL)\left(\frac{\tan(\theta_{sun})\tan(\theta_{sat})}{\tan(\theta_{sat}) - \tan(\theta_{sun})}\right)$$

Building modeling

- Height estimation
  - Relationship between height of building and length of shadow

$$SL = H/\tan(\theta_{sun})$$
$$VL = H/\tan(\theta_{sat})$$
$$S_{sun} = SL \times \cos(\varphi_{sun})$$
$$S_{san} = VL \times \cos(\varphi_{san})$$
$$\varphi_{sun} = \varphi_{su} + 90 - \varphi_t$$
$$\varphi_{san} = \varphi_{sa} + 90 - \varphi_t$$
$$S = S_{sun} - S_{san}$$
$$K_{\varphi t} = \frac{\sec(\varphi_{scan})}{(\cos(\varphi_{sun})/\tan(\theta_{sun})) - (\cos(\varphi_{san})/\tan(\theta_{san}))}$$
$$H = S \times K_{\varphi t}$$



Shettigara et al, 1998

Building modeling

- Height estimation
  - Building modeling with shadow
    - Rooftop-footprint offset 적용

$$R_1 = \begin{pmatrix} (H - S.z)D_1.x / D_1.z + S.x \\ (H - S.z)D_1.y / D_1.z + S.y \\ H \end{pmatrix}$$

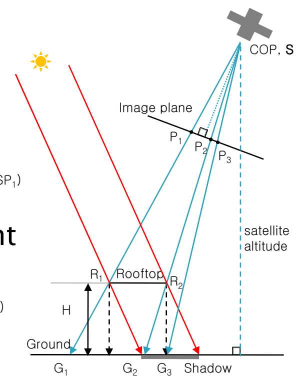
D_1 : direction vector (SP_1)

  - Building modeling with footprint

$$H = D_2.z(G_3.x - S.x) / D_2.x + S.z$$

$$R_2 = \begin{pmatrix} G_3.x \\ G_3.y \\ H \end{pmatrix}$$

D_2 : direction vector (SP_2)



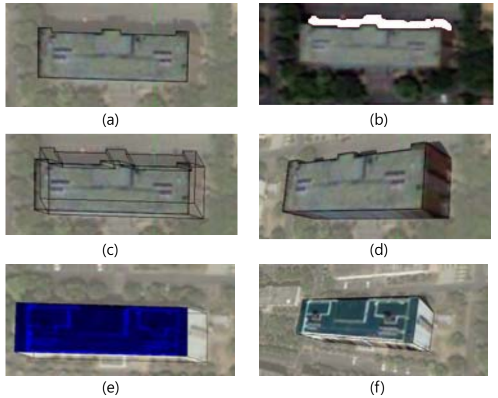
- Texture mapping

Experimental result

- Source image
  - Kyungpook Natl. Univ. Campus – Quickbird

Experimental result

- Building modeling with shadow



(a) Rooftop outline
(b) Extracted shadow region
(c) Estimated footprint and height of building
(d) Building model from shadow
(e) Height estimation using footprint and rooftop
(f) Building model from footprint

Experimental result

- Some result using shadow

Name	Estimation(m)	Real height(m)	Error(m)
E-7	16.153206	16	0.153206
E-9	22.985665	20.6	2.385668
D4	20.737482	17	3.737482

- Error caused from
  - Error from rooftop region extraction
  - Error from shadow region extraction

Conclusion

- Semi-automatic system for building modeling on single satellite image
  - Shadow region extraction from rooftop outline
  - Height estimation using shadow region
  - Acquisition of 3D model
- Future works
  - Minimizing shadow region error from Greenish object
  - Minimizing height error
  - Primitive based automatic rooftop extraction

Image Registration of Side and Rear Views for Panoramic Vision System

2008. 4. 24
Virtual Reality Lab.
Kyungpook National University
Min Woo Park, Kyung Ho Jang, Soon Ki Jung
mwpark@vr.knu.ac.kr, khjang@knu.ac.kr, skjung@knu.ac.kr

Introduction

- **What is blind spot ?**
  - The areas of the road that cannot be seen while looking forward or through either the rear-view or side mirrors
- **Assistant Vision System**
  - Reduce blind spot → Reduce accidents

Blind Spot

Blind Zone Mirror
(Magna Donnelly)

Video Vision
(Magna Donnelly)

Related work

- **System for Eliminating Blind Spot**
  - BMW Concept Car Z22
  - Ford CamCar
  - Magna Donnelly Panoramic Vision
    → General Motors (Motorshow 2000)
  - KIA Ophirus
  - NISSAN Infiniti Ex-35 Around View

NISSAN Around View

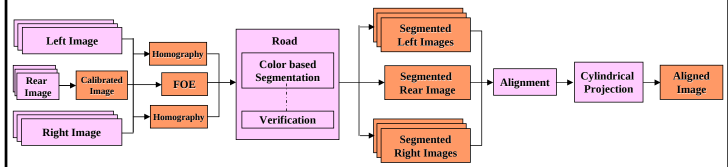
BMW Z22

KIA Ophirus

PanoramicVision

Overview

- **Assumption**
  - The vehicle is **running forward** in 70~80km/h
  - Position of the cameras is **fixed**
- **System Process**

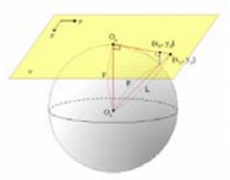


Calibration Step | Segmentation Step | Alignment Step

on the Fly

Calibrate Distorted Image in the Rear

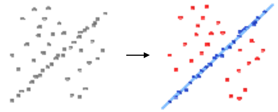
- **Wide angle camera (FOV 120˚)**
  - **Radial Distortion**
  - **Calibration (Non-metric method)**
    - Assumption
      - Lens is Sphere
    - Simple Method



5

Calibration

- **Features matching step**
  - Between rear and side images
  - Road and Background



RANdom SAmple Consensus

- **Homography**
  - Compute homography between rear and side images
  - Using RANSAC

Image a

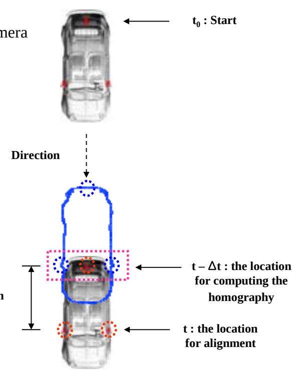
Image b

$p_b = \mathbf{H}_{ab} p_a \qquad \mathbf{H}_{ab} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$

6

Calibration

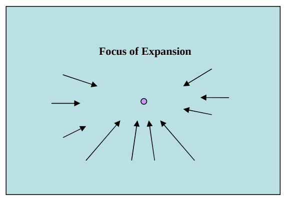
- **Images in the similar location**
  - Most image in the **similar location**
  - **Minimize translation** between rear and side camera
- **Problem**
  - Lost scene for the time Δt
- **Get time Δt**
  - Known car's speed
  - Known distance between side and rear camera
    - s = 2.8m
    - v = 80km/h
  - Δt = Δs/v = 2.8(m)/22.22(m/sec) ≈ 0.126 sec
    - 1 sec = 15 frame
    - 1 frame ≈ 0.067 sec
    - About 2 frame

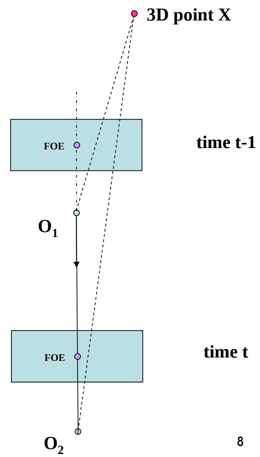


7

Calibration

- **Focus of expansion**
  - Compute focus of expansion
    - Find corresponding points
    - Compute feature flow
    - using RANSAC

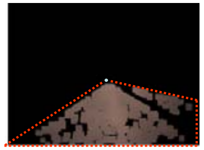


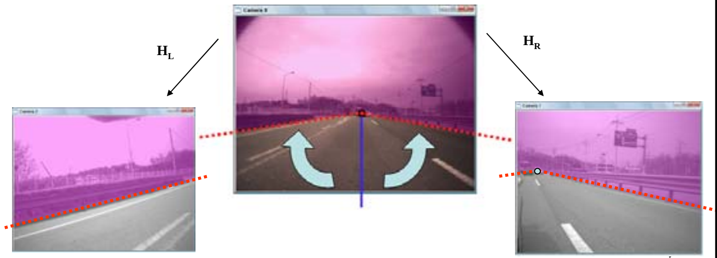


8

Segmentation

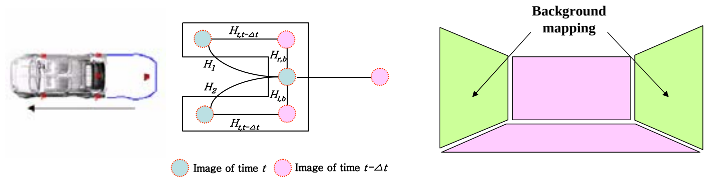
- **Color Extract in Sample road**
  - Extract and Use HSV color model
- **Road Detection**
  - Close operation (Erode → Dilate)





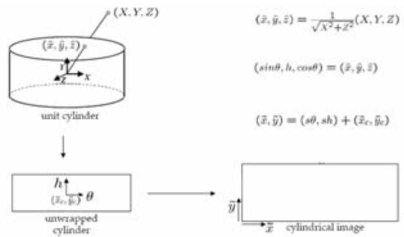
Alignment

- **Road alignment**
  - Compute homography of time t using Homography time of t-⊿t
- **Background alignment**
  - Set the plane in the side
  - Get background images
    - Background is the inverse of the road image
  - Transform background images into the side plane



Cylindrical Projection

- **Cylindrical projection**



$$(\hat{x}, \hat{y}, \hat{z}) = \frac{1}{\sqrt{X^2+Z^2}}(X, Y, Z)$$

$$(\sin\theta, h, \cos\theta) = (\hat{x}, \hat{y}, \hat{z})$$

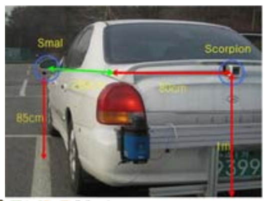
$$(\tilde{x}, \tilde{y}) = (s\theta, sh) + (\tilde{x}_c, \tilde{y}_c)$$

$$u = f^{*}\tan\left(\frac{x-0.5\alpha}{f}\right) + x_g$$

$$v = \frac{(y-y_g)\sqrt{(x-x_g)^2+f^2}}{f} + y_g$$

Experiment

- **Image Acquisition**
  - The vehicle is running forward in 70~80km/h
  - Rear camera
    - FOV 120˚
    - Point Gray scorpion
  - Side camera
    - FOV 30˚
    - 15~30˚ out panning



Experiment

13

Conclusion and Future work

- **Constrained Environment**
  - Motion parallax problem
  - Scale problem
- **Alignment**
  - Segmented plane
  - Align the several plane
- **Future work**
  - **<u>Moving object detection</u>**
  - Robust road detection

14

Design and Implementation of Presentation Support System based on Mobile Networks

Seung Dae Jeong, Yoon Suk Kwak, Soon Ki Jung

Virtual Reality Lab., Dept. of Computer Engineering, KyungPook National University

sdjeong@vr.knu.ac.kr, myshout@vr.knu.ac.kr, skjung@knu.ac.kr

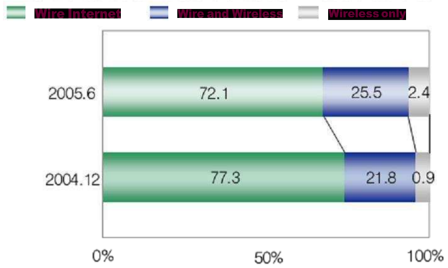
Motivation

❖ The world-wide number of mobile device(PDA, Smart-phone, MP3P…) shows a tendency to increase.

❖ Increased performance of mobile devices.

❖ Increased efficiency of wireless network.

❖ Enlarged utilization of wireless Internet.

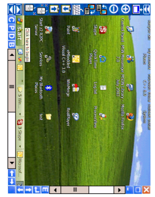


Related Works

❖ z2remote2PC
- ✓ Sharing of the PC screen to mobile devices
- ✓ P2P connection, like a VNC

❖ sidewindow
- ✓ Making PDA into extra monitor
- ✓ Expanding of PC desktop screen
- ✓ More PDAs → Available more various display

❖ Pebbles
- ✓ PDA → New extra input device of PC
- ✓ Research project of Carnegie Mellon University



z2remote2PC

sidewindow

Introduction

❖ **Traditional presentation**
- ✓ All audiences watch the screen at front.
- ✓ In order to control a reference data. The presenter leaving the platform is difficult.
- ✓ The audience interaction with the presenter is difficult.
- ✓ A case of remote presentation needs additional equipment of high price.

Introduction

❖ **Presentation with proposed system**
- ✓ Sharing reference data with audiences by their mobile devices.
- ✓ The presenter using his mobile device, control freely a reference data.
- ✓ The presenter is able to permit controlling a reference data by audience.
- ✓ Additional equipment there is not a necessity.

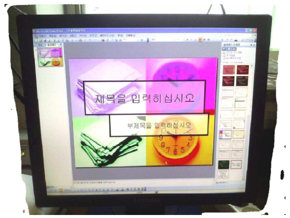
Camelot System

Camelot System

- ❖ **Presentation Client**
  - ✓ A Client of presentation software
  - ✓ Remote controlling by the presenter
- ❖ **Server**
  - ✓ Managing connection and transporting screen
  - ✓ The protocol is platform-independent
- ❖ **Presenter Client**
  - ✓ Presenter client can run on the PC or PDA
  - ✓ Having all permission. Can be able to permit audience
- ❖ **Audience Client**
  - ✓ Audience client can run on the PC or PDA
  - ✓ After getting permission by presenter, can control the presentation

Camelot System - Interface

Conclusion & Future Work



- ❖ **For efficiency presentation, Audiences can get permission of controlling the reference data.**
- ❖ **Additional equipments is not necessary.**
- ❖ **Providing platform-independent protocol and using various input style.**

- ❖ **Future Work**
  - ✓ Break from one-way transporting screen.
    - Two-way transporting possibly does, data of the audience according to necessity joint ownership.
  - ✓ Providing various interfaces according to devices
  - ✓ Providing extra functions according to presentation programs

Presentation: Jae Seok Jang @ vrlab
<jsjang@vr.knu.ac.kr>

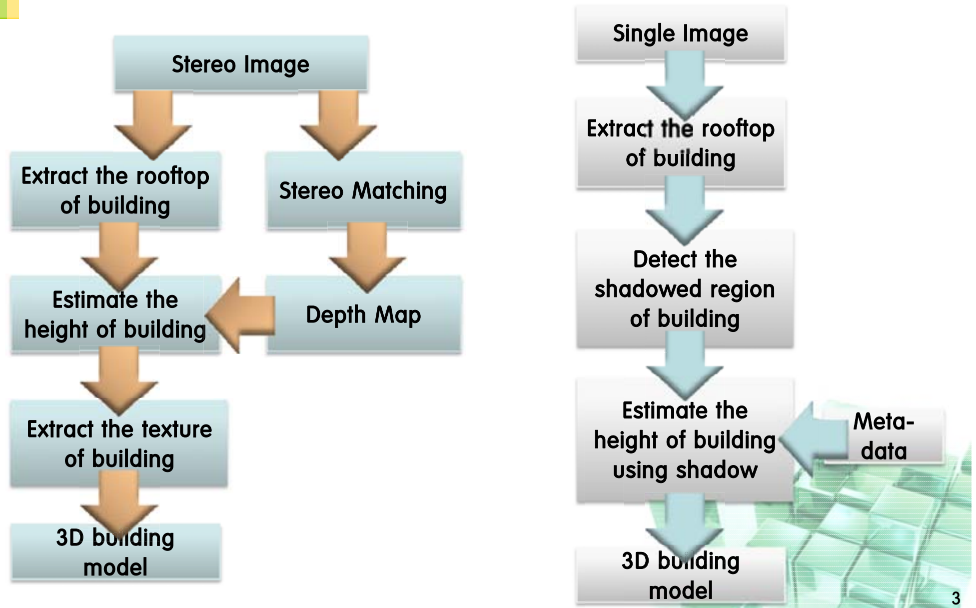
Shadow Analysis of Satellite Images for 3D Building Reconstruction

Jae Seok Jang[1] ,Kwang Hee Won [2]
Kyoung Ho Jang[3],Soon Ki Jung[4]
-Kyoungpook National University, Korea-

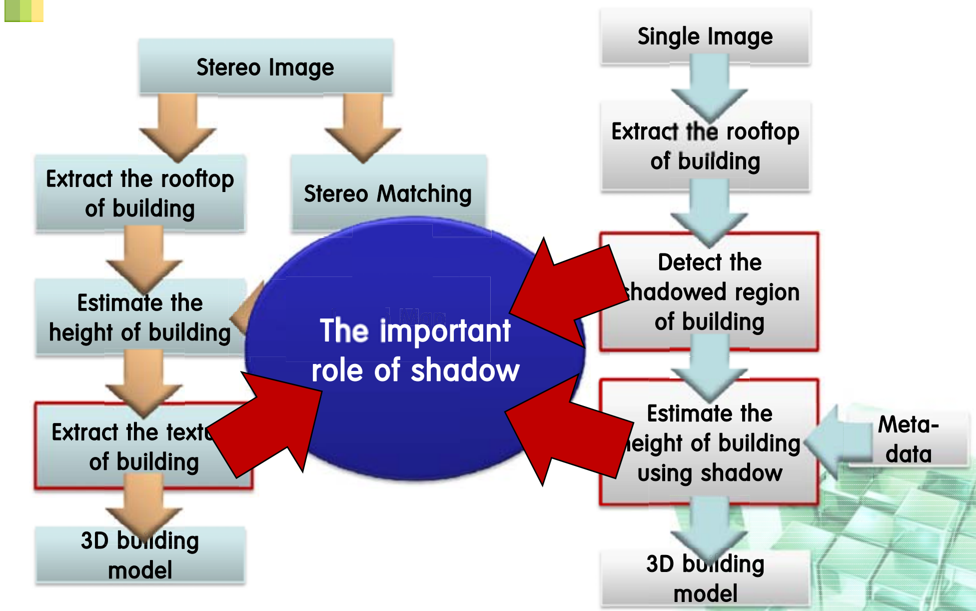
[1]jsjang@vr.knu.ac.kr,
[2]khwon@vr.knu.ac.kr
[3]khjang@knu.ac.kr
[4]skjung@knu.ac.kr

Contents

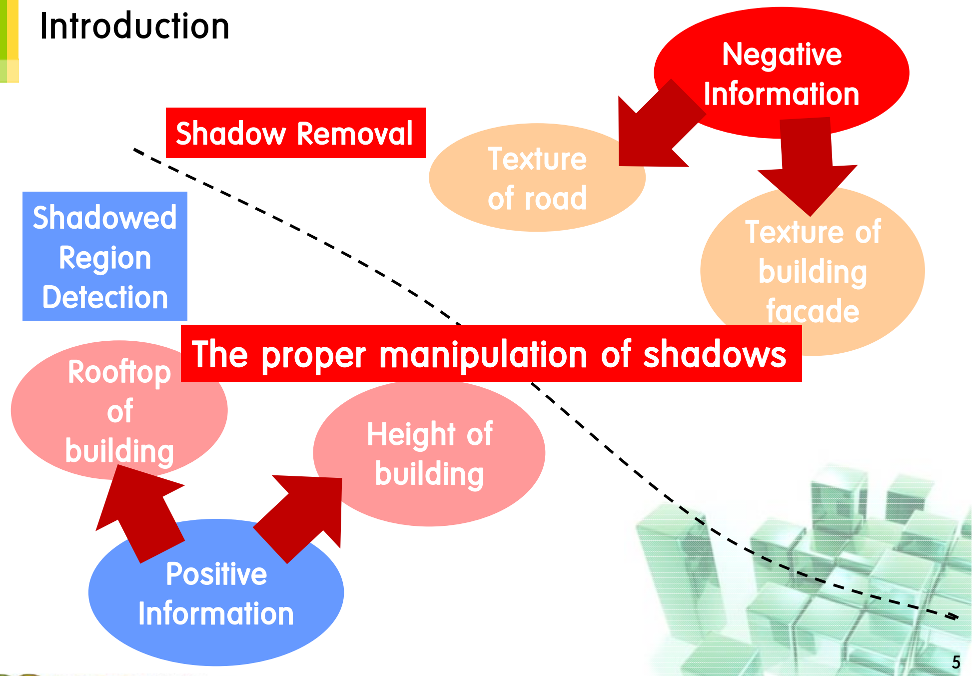
Introduction



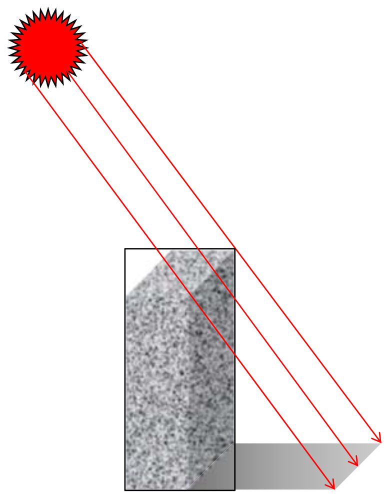
Introduction



Introduction

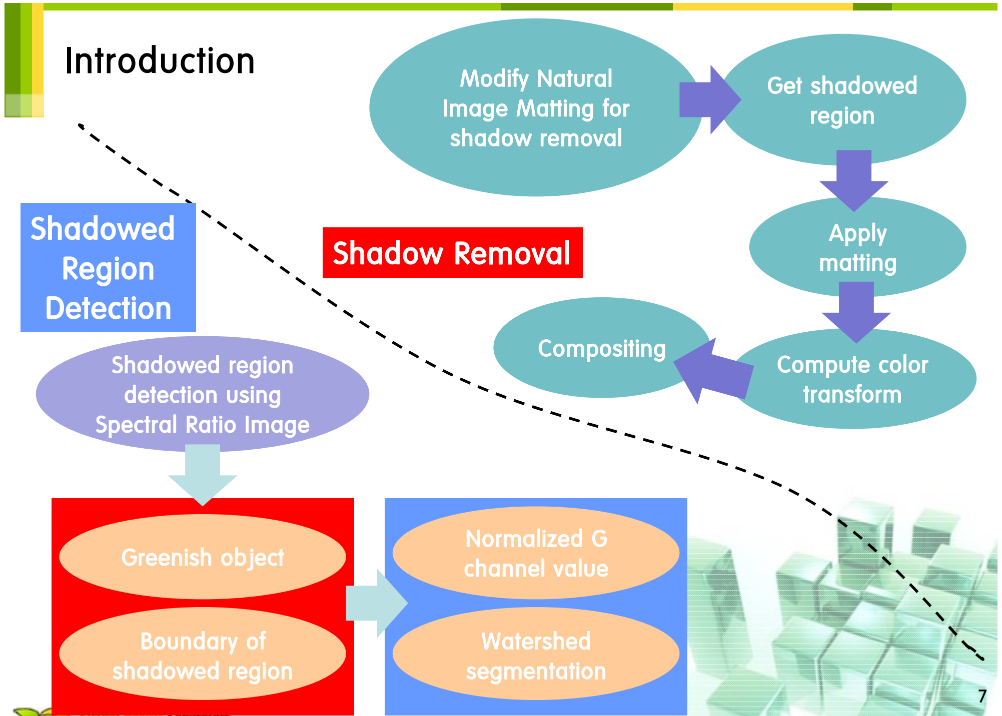


Introduction



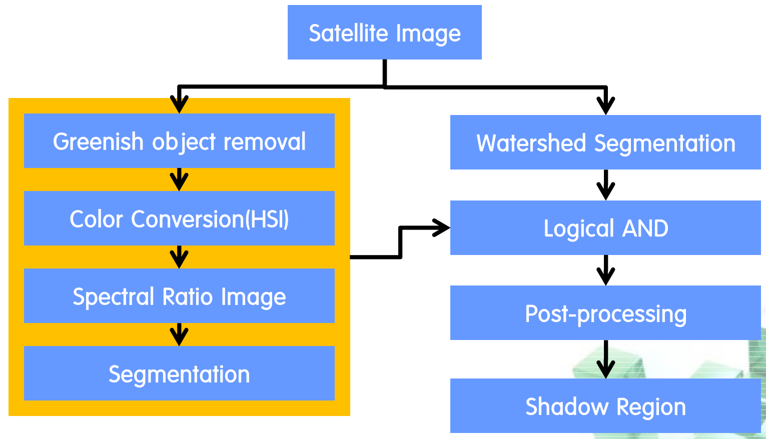
- "A Comparative Study on Shadow Compensation of Color Aerial Images in Invariant Color Models" ,Victor J. D. Tsai[2003]
  - Shadow Properties
    - Low intensity
    - Higher saturation with short blue-violet wavelength
    - Increased hue value

Introduction



Shadow Manipulation

- **Shadowed region detection**



Shadow Manipulation

- **Shadowed region detection**
  - Shadowness from V. J. Tsai research[2003]
    - Spectral Ratio Image
      - High value in shadowed region
    - Greenish object
      - Remove normalized G value higher than threshold
  - The boundary of shadowed region
    - Watershed segmentation

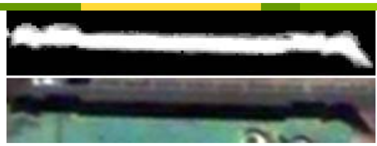
Virtual Reality Laboratory
Kyungpook National University

Shadow Manipulation

- **Shadow removal with Natural Image Matting**
  - Strategy
    - Step 1. Find β, the blending factor of shadowed and lit region, especially at the boundary of the shadow
    - Step 2. Apply color transfer function to recover lit color from shadowed color
    - Step 3. Blend transferred color(Step 2.) and lit color using blending factor β (Step 1.)

Virtual Reality Laboratory
Kyungpook National University

Shadow Manipulation



βvalue(up) and input image(down)

- **First step**

$$C_o = \beta C_s + (1-\beta) C_l$$

- Unknown region
  - Sample data
    - Using Gaussian weighted window
  - Estimate C_s and C_l to calculate β (0~1)
  - Clustering the sample colors of C_s and C_l
  - Evaluate the reconstruction error to find the best pair

Virtual Reality Laboratory
Kyungpook National University

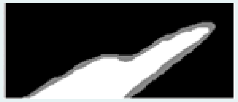
Shadow Manipulation

- **Shadow removal with Natural Image Matting**
  - Second step
    - Find the color transfer between shadowed and lit colors
    - Two method
      - The linear color transform into RGB color space
        - Only consider translation and rotation
      - The Gaussian mixture model(GMM) matching
        - Extract GMM from the histogram of shadowed and lit colors
        - Transfer the colors of each pixel in shadowed region

Virtual Reality Laboratory
Kyungpook National University

Shadow Manipulation

- **Shadow removal with Natural Image Matting**
  - Final step
    - Determine the transferred colors of the region {S ∪ U}

Source Image		
Tri-map		
Linear Color Transform		
GMM Matching		

Result

- **Experiment data**
  - Pansharped satellite image of the area, Kyoungpook National University, Korea from Quickbird
  - The GSD is about 0.6m

Result

- **Extracted Shadowed Region**

Conclusion

- **Shadowed region detection**
  - Removal of greenish object(trees, grasses, etc…)
  - The boundary of shadowed region
    - Watershed segmentation
- **Shadow removal**
  - We employed natural image matting to shadow removal
    - The discontinuity problem of the boundary pixels
      - By blending factor (beta)

Discussion

- **Greenish object**
  - Not sufficient only G channel value
- **Unnatural look of shadow-less image**
  - Shadowed region
    - Lost their information
  - The amount of sample
    - Too small to recover the original color distribution

Future Work

- **Distinguish of shadowed region and greenish object**
- **Complete boundary of shadowed region**
- **The loss of information due to the shadow phenomenon**
- **The problem of insufficient sample data**