EUROGRAPHICS 2002

Tutorial T5: Tutorial on Inhabited Virtual Heritage

Nadia Magnenat-Thalmann, University of Geneva
Alan Chalmers, University of Bristol
Pascal Fua, EPFL, Lausanne
Daniel Thalmann, EPFL, Lausanne

Published by
The Eurographics Association
ISSN 1017-4565

The European Association for Computer Graphics
23rd Annual Conference

EUROGRAPHICS 2002

Saarbrücken, Germany
September 2–6, 2002

Organized by

EUROGRAPHICS
THE EUROPEAN ASSOCIATION
FOR COMPUTER GRAPHICS

INFORMATIK
Max-Planck-Institut
für Informatik
Saarbrücken, Germany

Universität des Saarlandes
Germany

International Programme Committee Chairs
George Drettakis (France)
Hans-Peter Seidel (Germany)

Conference Co-Chairs
Frits Post (The Netherlands)
Dietmar Saupe (Germany)

Honorary Conference Co-Chairs
Jose Encarnação (Germany)
Wolfgang Straßer (Germany)

Tutorial Chairs
Sabine Coquillart (France)
Heinrich Müller (Germany)

STAR Report Chairs
Dieter Fellner (Germany)
Roberto Scopignio (Italy)

Lab Presentation Chairs
Günther Greiner (Germany)
Werner Purgathofer (Austria)

Industrial Seminar Chairs
Thomas Ertl (Germany)
Bernd Kehrer (Germany)

Günter Enderle Award Committee Chair
François Sillion (France)

Conference Game Chair
Nigel W. John (UK)

John Lansdown Award Chair
Huw Jones (UK)

Conference Director
Christoph Storb (Germany)

Short/Poster Presentation Chairs
Isabel Navazo (Spain)
Philipp Slusallek (Germany)

Local Organization
Annette Scheel (Germany)
Hartmut Schirmacher (Germany)

Virtual Heritage

by Nadia Magnenat-Thalmann

Outline

- The Terra Cotta Soldiers
- Generating Animatable 3D Virtual Humans from Photographs
- Flashback to the Future
- Virtual fashion of the past and the future
- The making of the SS. Sergius and Bacchus edificeFacial Animation
- From Facial Mesh to Expressive Talking Faces

The Terra-Cotta Soldiers (1995)

Nadia Magnenat-Thalmann
Marlène Arévalo
Gaël Sannier

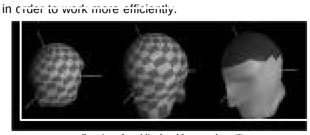
The Xian Project

- Excavation of the grave complex of the Ch'in emperor Shi Huang T i in Xian in the 1970s has revealed a field of statues depicting soldiers, servants, and horses, estimated to total 6'000 pieces. The figures were modeled after the emperor's real army, and each face is different.
- The Xian project in 1997 is intended to recreate and give again life to this army using computer-generated techniques.

Discovery of the statues

Sculpting the Soldiers' Faces (I)

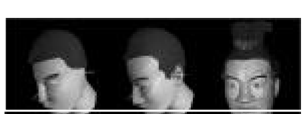
- The real soldier faces are all different and have details.
- We use a method similar to the modeling of clay; It consists of adding or eliminating parts of the material, and turning around the object.
- The steps of the first head modeling (I):
  - We apply scaling deformations on a sphere to obtain an egg shape aspect.
  - We move regions selected with triangles & also lift or move vertices.
  - We split in half in order to work more efficiently.



Creation of a soldier head from a sphere (I)

Sculpting the Soldiers' Faces (II)

- The steps of the modeling (II):
  - We model specific regions (nose, jaws, eyes, etc) by sculpting and pushing back and forth vertices and regions.
  - We obtain an half face of the soldier to which we apply a reversed scaling on X axis to produce the other half.
  - The two sides are merged together which finally give us our first soldier's face.



Creation of a soldier head from a sphere (II)

Texture-fitting (I)

- To increase realism, we apply texture fitting to objects. We map a picture onto the object, in a way that allows the user to specify some matching points between the texture and the object:
  - We can see the texture while fitting it to the object.
  - Some interesting vertices are selected, suitable for circumscribe the area and fitting the texture to some specific features of the model. All these marked vertices are projected to the texture image.
  - We move each projected vertex to its right position on the 2D texture. The 3D object is mapped in real-time in the 3D window using the information given by the position of these marked vertices on the texture image.

Adjusting features upon the texture image

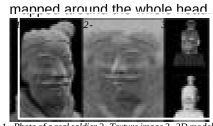
Result of the fitting in real-time in 3D

The Film (I)

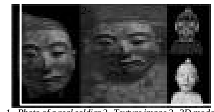
- Scenario:
  – We see first a scene with the 3D terra-cotta soldiers inside the earth.
  – It is dark with a starry sky.
  – The day is coming so more and more light is appearing. This suddenly awakes one terra cotta soldier. He is extremely astonished to see the scene around himself…

Texture-fitting (II)

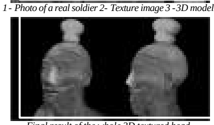
- As we only have a single photo of each soldier face to model from, we create a global texture using this photo, so that this texture can be mapped around the whole head.



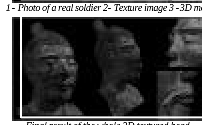
1 - Photo of a real soldier 2- Texture image 3 -3D model



1 - Photo of a real soldier 2- Texture image 3 -3D model



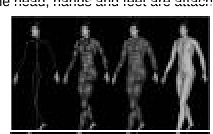
Final result of the whole 3D textured head



Final result of the whole 3D textured head

Creating the Soldier Bodies

- Our goal is to make realistic and efficient human modeling and deformation capabilities for many different bodies. So we use the metaball technique as it is inherent to interactive design.
- The metaballs hierarchy is taken from a standard model we have, we then modify the metaballs positions and shapes to fit soldiers anatomy.
- The head, hands and feet are attached to our body envelope.

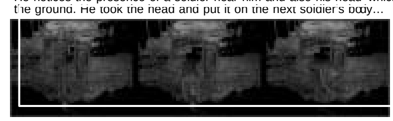


Metaball-based body

Head and Hand attached to the body

The Film (movie)

– He notices the presence of a soldier near him and also his head which is on the ground. He took the head and put it on the next soldier's body…



– This latter start to live again. They look at each other, and all the army is slowly coming to life. They start to walk again, but the first soldiers decide to let them go...

Generating Animatable 3D Virtual Humans from Photographs

Nadia Magnenat-Thalmann
Won-Sook Lee

Contents

Motivation

- Importance of realistic virtual human is getting growing
- In the future, virtual twins of us will populate the virtual worlds
  - not a simple cube, an animal, an alien
  - not only Marilyn Monroe
  - but **YOURSELF**!

Introduction

- Two techniques depending on the interest
  - *accuracy and precision of the obtained object model shapes*,
    - CAD systems, medical application.
  - *visual realism and speed for animation of the reconstructed models*,
    - internet applications
    - Virtual Reality applications.

Virtual humans for real-time applications

- What's the components to consider?
  - acquisition of human shape data
  - realistic high-resolution texture data
  - functional information for animation of the human (both face and body)

Virtual humans for real-time applications

- What to produce?
- What is the input data?
- What is the environment to get the input data?
- How much automatic is the process for users?
- How much can we animate the virtual human?

State of the Art - Face

- Plaster model
  - marks on a real model and photographs
    - [Magnenat-Thalmann 87] [DeRose 98]
- Photographs (unorganized)
  - Interactive deformation, texture mapping [LeBlanc 91][Sannier 97]
  - Generic database (unorganized photographs) [Blanz 99]

State of the Art - Face

- Features on photographs (organized) and a generic model
  - Modeling used for getting the individualized face using a few points
    - [Kurihara 91] [Akimoto 93] [Ip 96]
  - Modeling used for expression database
    - [Pighin 98]

State of the Art - Face

- Range data
  - Laser scanner
    - *Cyberware Color Digitizer™* [http:cyberscan]
    - [LeeY 95] [Guenter 98] [Blanz 99]
  - Stripe generator
    - [Proesmans 97]
  - Several photographs with sequences of contours
    - [Nagel 98] [Zheng 94]

State of the Art - Face

- Range data
  - Stereoscopy
    - [http: turing] [Fua 96]
  - Video camera
    - with markers [Guenter 98]
    - Uncalibrated video, using a generic face model [Fua 00] [Cohen 00]
    - Optical flow without markers [DeCarlo 96]

State of the Art - Body

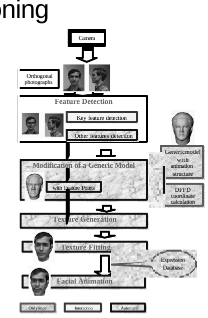
- Laser scanner
  - *Cyberware™* Whole Body scanner
- Silhouette in multiple views
  - video [Kakadiaris 95] [Kakadiaris 96]
  - photographs [Gu 98] [Weik 98] [Hilton 99]
- Stereo-video [Plänkers 99]

State of the Art - comparison

Photography	Laser Scanner
Cheaper	Expensive
Very general equipment	Special equipment
	Output: Numerous points
Usually high resolution of texture mapping	Usually low resolution of texture mapping
Easy to catch characteristic points	Often noisy to catch characteristic points
Difficult to catch non-characteristic points	Better to catch non-characteristic points
	Problems for hairy parts

Face Cloning

- Input
  - photograph
  - generic head & animation
- Method
  - Feature based
- Output
  - Animatable virtual human



Head shapes from photos

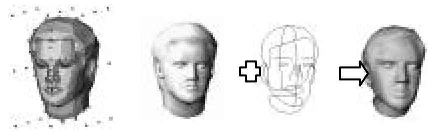
- Features on front (and side) view
  - eyes, nose, lips, hair and face outlines, etc.
- Semiautomatic structured feature detection
  - piecewise affine mapping
  - structured snake to keep structure of points

Head shapes from photos in 3D rather than in 2D

- Generation of (x, y, z) from (x, y_f) and (y_s, z)
  - criteria for giving more importance on the front view
  - robust even though the input photographs are not perfectly orthogonal
- Dirichlet FFD (DFFD)
  - the convex hull of a set of control points in general position

Head shapes from photos

- Feature points < control points



Texture mapping

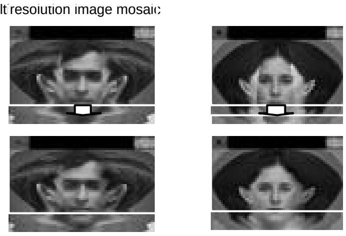
- Texture Generation
  - One texture image from two images
    - Geometrical deformation
    - Multi-Resolution techniques
- Texture Mapping
  - Projection to three planes
  - Transformation to several spaces

Seamless texture mapping

- Texture generation
  - Image deformation

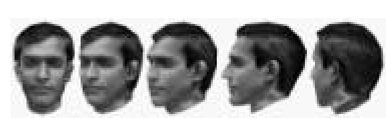
Seamless texture mapping

- Texture generation
  - Multiresolution image mosaic



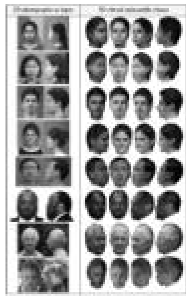
Results

- Rotation in 360 degree



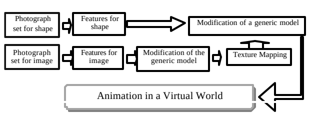
Results

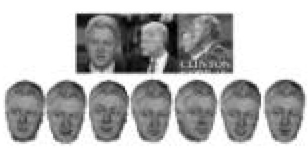
- Several ethnic group from one generic model



Results

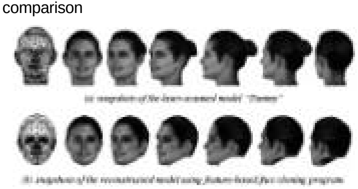
Results - shape texture separation





Results - Validation

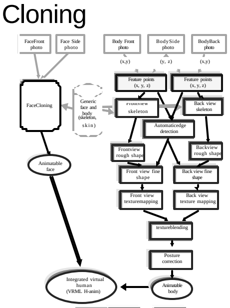
- Visual comparison



- 3D- distance measurement : 2.84306 %

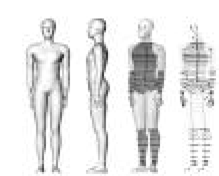
Body Cloning

- Input
  - three photographs
  - H-Anim 1.1 generic body
- Feature - edge based
- Output
  - animatable vitual human



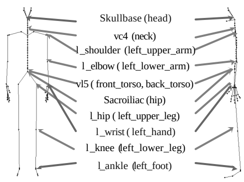
Body Cloning - Generic body

- Continuous mesh humanoids
  - MPEG-4 compatible H-Anim 1.1 formats [http:H-Anim]
  - 94 skeleton joints & 12 skin parts (different from the face with only skin)



Body Cloning - Generic body

- H-Anim joints related to skin parts
  - the local coordinates of the skin part *i* to global coordinate by 4x4 matrix M_i.

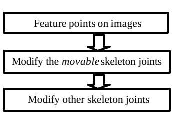


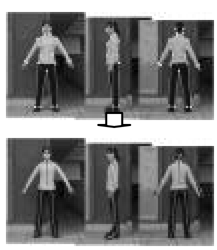
Body Cloning - Generic body

- Skin has grid structure
  - each skin part has several slices
  - each slice on the skin part has the same number of points
  - Share the same 3D coordinates between different skin part
- Resulting seamlessly continuous skin envolope

Body Features and skeleton

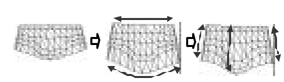
- Features and skeleton adjust





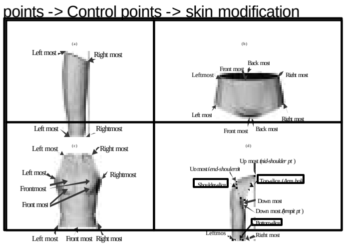
Body rough skin adjustment

- Rough skin modification
  - Problem
    - simple affine mapping
  - Solution
    - freeform deformation
    - grid structure
    - piecewise affine mapping



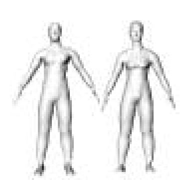
Body rough skin adjustment

- Feature points -> Control points -> skin modification



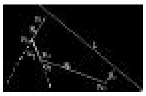
Body rough skin adjustment

- Two bodies proper orientation and rough size
  - Front+Side / Back+Side

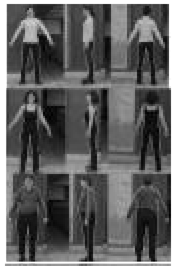


Body fine skin adjustment

- Feature driven edge extraction



  - Canny edge detector
  - Each *feature segment* indicates the vicinity and approximate direction of the boundary to be found
  - evaluate the "goodness" of the potential connection



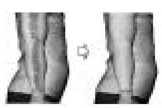
Body fine skin adjustment

- Edge-based modification
  - Fine Skin modification
    - silhouette modifies the contours of the skin surface
    - Correspond edge pixels between front(back) and side view
    - Modify a skin slice using two or four points

Body Cloning - Texture mapping

- Front and side views are used
  - Deform body and texture for each side separately
- Texture blending
  - Problem caused by

  digitization and illumination
  - Linear blending following corresponding edges on the front and back views



Body and Face together

- Automatic connection with own face from face cloning system
  - use features on face and body
- Neck adjustment
  - bridge to connect the face and body smoothly and seamlessly

Body Results

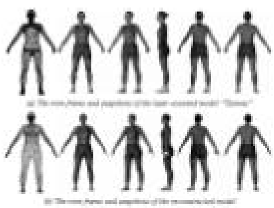
- H-Anim 1.1 format
  - visualized by web browsers
  - Animatable

Body Cloning - Results

- Sometimes postcorrection needed
  - Skeleton correction from skin envolope
    - Elbow skeleton correction
  - H-Anim & Vicon (optical motion capture system) posture
    - length and angle coordinate
    - adjust angles for arms and legs

Results - Validation with laser scanned data

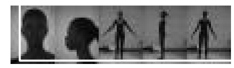
- Visual comparison & distance measurement - height 1.76m

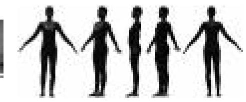


Body part	*Avg. Error (m)*
hip	0.0118313
left_upper_arm	0.0103196
front_torso	0.0103044
right_upper_leg	0.00927971
left_lower_leg	0.00835839
left_upper_leg	0.00795227
right_lower_leg	0.00763058
right_upper_arm	0.00748069
neck	0.00702401
back_torso	0.00701802
left_lower_arm	0.0066277
right_lower_arm	0.00445429
total	**0.00844698**

Results - Validation with a living person

- Tailor's measurement





	Tailor's (m)	*Model's (m)*	*Diff. (m)*
Height	1.74	1.75029	0.01029
Waist	0.68	0.727171	0.047171
Hip	0.894	0.88432	-0.00968
Chest	0.835	0.819402	-0.015598

Animation result with motion capture

- Animation with cloned body
  - Comparison with real motion

Flashback to the Future

Nadia Magnenat-Thalmann
Marlène Arévalo

The Project

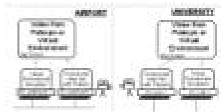
- A virtual reality experience developed in the MIRALab research laboratories of the University of Geneva. This real-time adventure, with 3D glasses, has been experienced at Palexpo in October l999, during Telecom'99.

The Project

- To illustrate telecommunications, the show communicates in real time with three distant booths, one located in Palexpo, the second one in the Uni Dufour Hall and the third one at the Geneva Airport.

Booth at Palexpo

The Project



Booths at the University and at the Airport

The Project

- Real people are being cloned, and their virtual counterparts take part in 3D scenes from the past and the future.
- To do the virtual double of each person, we use a procedure based on two photographs, that can reconstruct the faces of individuals in 3D.

Face Cloning

The Project

- This world première illustrates the face-to-face interaction within the virtual scene of individuals who in reality are situated at a distance fom each other, like you and I.
- It is also a first for the reconstruction of the Vieille Ville by computer and for the appearance of a virtual Mère Royaume.

The Vieille Ville of Geneva in real

The Vieille Ville of Geneva in virtual

The scenario

- Those who are cloned become actors in 3D scenes. First, we see a virtual reconstitution of the Bourg de Four place in Geneva as it looks today.

Present (1999)

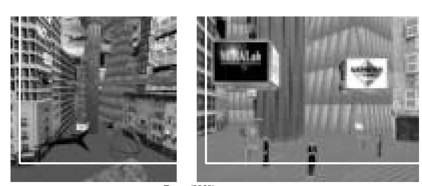
The scenario

- After, we make a ride back to the year 1602 such as the "Escalade" of 1602, which is an important date in the Geneva History.

Past (1602)

The scenario

- Then we enter into the future world of 2202, where the spectator can realize the importance of telecommunications and information in Geneva's Vieille Ville of the future.



Future(2202)

1602

- Escalade: soldiers from Haute Savoie tried to invade Geneva and were stopped by the Geneva inhabitants and more particularly the "Mere Royaume", who spilled the content of her cauldron over the invaders.

The Mère Royaume and 2 soldiers

1602: The Mère Royaume

Virtual fashion of the past and the future
by

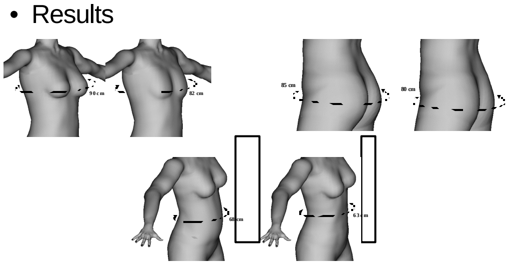
Professor Nadia Magnenat-Thalmann

MIRALab

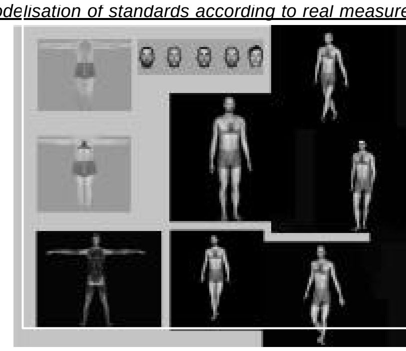
Measurement-based Body Creation

Volumetric deformation

- Results

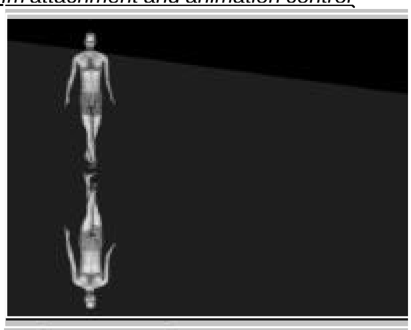


Modelisation of standards according to real measurements

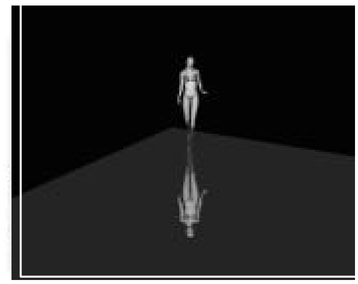


H-Anim attachment and animation control

movie



Modelisation of standards according to real measurements



The State of Art
Garment Simulation at MIRALab

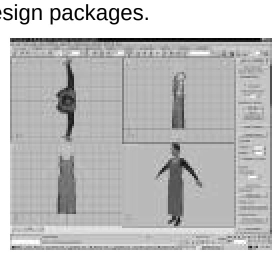
- History
  - Lafleur, Magnenat-Thalmann, 1991:
    Simple viscoelastic surfaces using Lagrange equations.
  - Carignan, Magnenat-Thalmann, Yang, Werner
  - 1991-92-93:
    Modified Terzopoulos model with octree collision detection and avanced pattern-seaming garment design.

Mechanical Simulation Systems
General Mechanical Parameters

- Internal Forces (From surface deformations)
  - Elasticity (metric, curvature)
  - Viscosity
  - Plasticity
- External Forces (From environment interactions)
  - Gravity, Air Viscosity
  - Contact reaction, Friction
  - Misc. Interactions

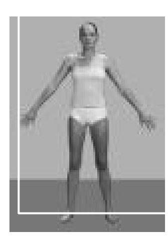
Garment Design Software

– Integration as plug-ins in common 3D design packages.

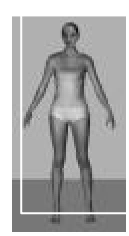


MOVIE

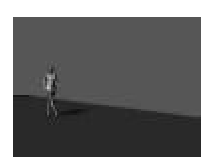
Improvement on the body: the «animated scanned Claire»



Real Claire



Virtual Claire



Virtual walk

Relevance of mechanical parameters

- Identify the relevance of mechanical parameters in the motion of fabric
  - Garments in motion: Dissipative parameters have importance (viscosity, plasticity, aerodynamic interactions).
  - Dissipative parameters are not measured by standard experiments (FAST, KES,…).
- Use mechanical simulation to reproduce the effect of dissipative parameters
  - Implementation of an efficient model adapted for simulation of cloth with usual elastic properties plus additional dissipative properties.
  - Evaluation of the relevance of properties through experiments.

Numerical Parameters

- Testing different time steps and different mesh discretizations.
  - Too few polygons: Inaccurate deformations.
  - Too large time steps: Incorrect motion.
  - Adequate dynamic motion accuracy:
    - Resolution with implicit Euler.
    - 0.2 milliseconds time steps.
    - 800 polygons for a 40x40 cm square.

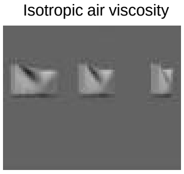
Viscosity Parameters

- Even high, viscosity does not suppress quickly the residual oscillations.
- Plasticity effects may have to be considered for efficient energy dissipation.
- It is difficult to distribute realist energy dissipation between aerodynamic effects and internal viscosity.

Aerodynamic Parameters

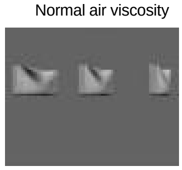
- Simple Aerodynamic Model
  - Isotropic and Normal air viscosity coefficients which respectively represent how the fabric pushes and slides onto the surrounding air masses.

Isotropic air viscosity



2.10^{-8} N.m.s^{-3} 2.10^{-9} N.m.s^{-3} 2.10^{-10} N.m.s^{-3}

Normal air viscosity



2.10^{-7} N.m.s^{-3} 2.10^{-8} N.m.s^{-3} 2.10^{-9} N.m.s^{-3}

Material (internal) parameters

- Metric elasticity: measurement of the fabric elongation elasticity (N.m^{-1})
  - Weft and Warp elasticity: elasticity along the Weft and Warp directions
  - Shear elasticity: elasticity for a shearing deformation between weft and warp directions
- Bending elasticity: measurement of the fabric bending elasticity (N.m)
  - Weft and Warp bending: bending along the Weft and Warp directions
- Viscosity parameters: defined for each elastic parameter
- Density: mass per surface unit of the fabric (Kg.m^{-2})

Contact parameters

- Thickness of the fabric (m)
- Coulombian friction: ratio between the maximum tangential contact force and the normal pressure force between two surfaces in contact

Environment (external) parameters

- Gravity: nominal acceleration of objects left at rest (9.81 m.s^{-2})
- Aerodynamic viscosity: aerodynamic force exerted on a fabric per surface unit and per velocity unit between the fabric speed and the air speed:
  - wind
  - Normal (Flowing: N.m^{-3}.s) and tangential (Damping: N.m^{-2}.(m.s^{-1})$^{-1}$) components relative to the orientation of the fabric surface

Examples of material (internal) parameters

Lycra

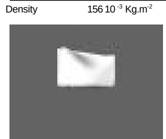
Weft Elasticity	16.67N.m^{-1}
Warp Elasticity	10 N.m^{-1}
Shear G	217N.m^{-1}
Weft Bending	17 10^{6} N.m
Warp Bending	6.5 10^{-6} N.m
Density	156 10^{-3} Kg.m^{-2}

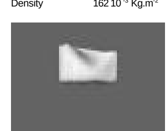
Cotton

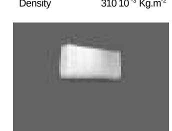
Weft Elasticity	16.67N.m^{-1}
Warp Elasticity	16.67N.m^{-1}
Shear G	60 N.m^{-1}
Weft Bending	10.5 10^{6} N.m
Warp Bending	6.7 10^{-6} N.m
Density	162 10^{-3} Kg.m^{-2}

Linen

Weft Elasticity	50 N.m^{-1}
Warp Elasticity	50 N.m^{-1}
Shear G	55 N.m^{-1}
Weft Bending	208.1 10^{-6} N.m
Warp Bending	153.9 10^{-6} N.m
Density	310 10^{-3} Kg.m^{-2}



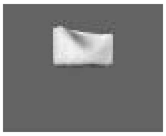
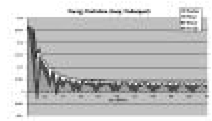




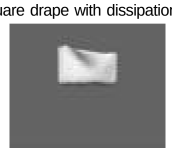
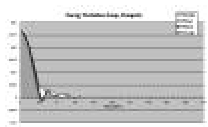
MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

Comparison & Drape Energy

- Cotton Square drape without dissipation

- Cotton Square drape with dissipation

www.miralab.unige.ch
University of Geneva

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

Material for Dressed Bodies: Lycra & Silk

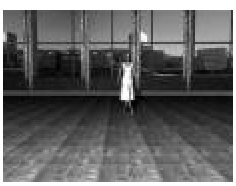

Lycra


Silk

www.miralab.unige.ch
University of Geneva

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

Material for Dressed Bodies: Cotton & Cupro

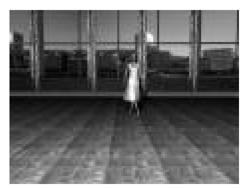

Cotton

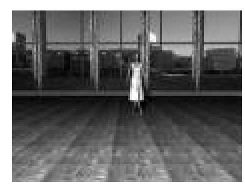

Cupro

www.miralab.unige.ch
University of Geneva

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

Material for Dressed Bodies: Linen & Tencel


Linen


Tencel

www.miralab.unige.ch
University of Geneva

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

Comparison: Cotton

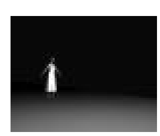
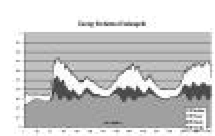

Cotton undamped Cotton damped

www.miralab.unige.ch
University of Geneva

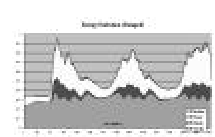
MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

Garment Energy

- Animated dress without dissipation

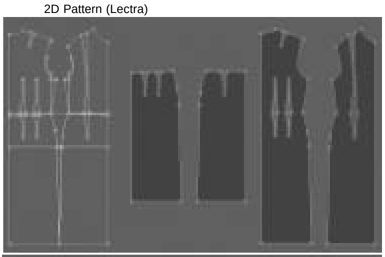



- Animated dress with dissipation

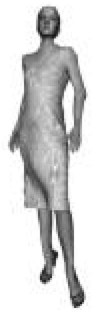



www.miralab.unige.ch
University of Geneva

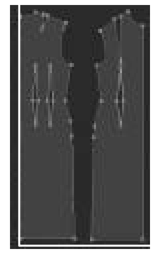
2D Pattern (Lectra)

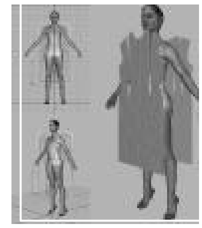


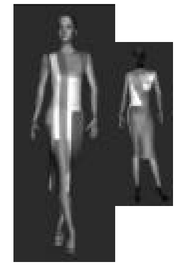
3D Basic Dress



MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

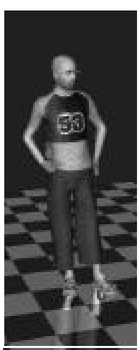


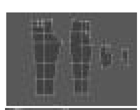




www.miralab.unige.ch
University of Geneva

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

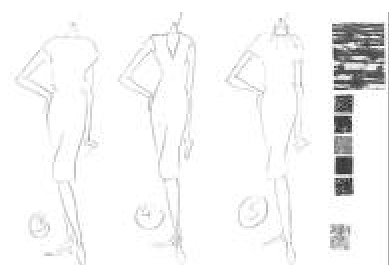




www.miralab.unige.ch
University of Geneva

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

E-Tailor

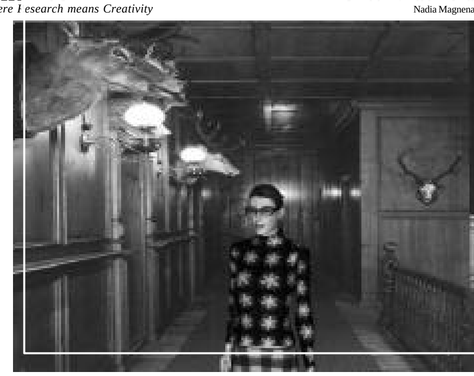


www.miralab.unige.ch
University of Geneva

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

www.miralab.unige.ch
University of Geneva

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann



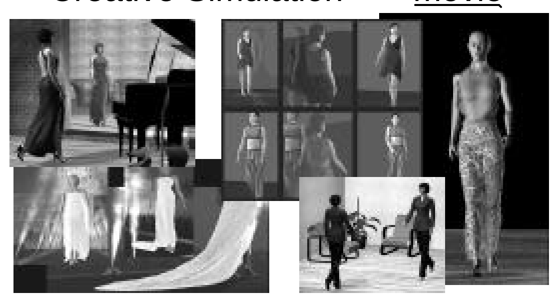
www.miralab.unige.ch
University of Geneva

first simulated outfit

texture tests

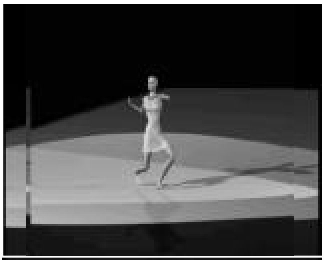
Virtual Fashion Design

Creative Simulation movie

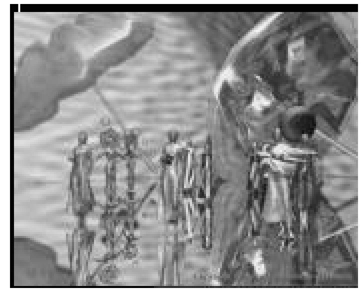


Fashion Show

movie



Utopians



The making of the SS. Sergius and Bacchus edifice

Nadia Magnenat-Thalmann
Allessandro Foni
Grégoire L'Hoste
Georgios Papagiannakis

The CAHRISMA project (I)

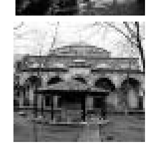
- Main objective of the CAHRISMA project (Conservation of the Acoustical Heritage by the Revival and Identification of the Sinans Mosques) is to innovate the concept of hybrid architectural heritage.
- Hybrid architectural heritage is a new way of identification that covers acoustical characteristics besides visual peculiarities.
- It states that, for the spaces, having acoustical importance, architectural heritage concept should be upgraded covering acoustical and visual properties. The effects of this improvement will reflect to actual implementation of conservation and restoration.

The CAHRISMA project (II)

- MIRALab's involvement:
  - Real-time visualisation of selected spaces.
  - Creation of people (virtual bodies, faces and cloth textures).
  - Animation of virtual humans.
  - Integration of visual and acoustical models into a virtual 3D interactive system.
- One of the monuments selected for this project is SS. Sergius and Bacchus edifice in Istanbul.

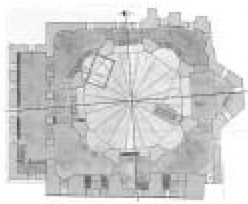
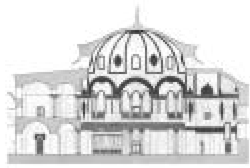
SS. Sergius and Bacchus church

- The church of the SS. Sergius and Bacchus, a landmark in Byzantine ecclesiastical architecture, was founded by Justinian probably in 527, the first year of his reign.
- The church of the SS. Sergius and Bacchus known to this day as "the Little Hagia Sophia", because the general principles of its architecture are comparable with those of the Great Church.
- Sometime between 1506 and 1512, the church of the SS. Sergius and Bacchus was converted into a mosque. The atrium was replaced by a peristyle, surviving to this day, and a courtyard where the medrese (religious school) stands today.



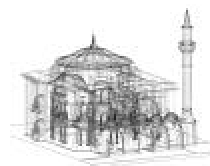
Reconstruction of the edifice 3D model (I)

- The 3D model of the SS. Sergius and Bacchus edifice is reconstructed from the available architectural plans and the visual data resulted from the data collection process performed by UNIGE.

Reconstruction of the edifice 3D model (II)

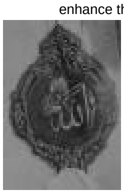
- The whole edifice is reconstructed in three dimensions using polygonal method of 3D Studio Max software.



View of the mesh model from 3D Studio Max

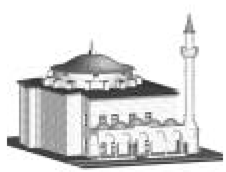
Texturing the 3D model

- The texture are created from 2D photographs, they are used as texture image maps to improve the visual details of the 3D model. A spec ial care is taken to correct for the perspective of the picture and to enhance the aspect of the texture.



Actual picture

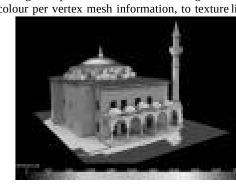
Texture extracted from the picture



Textured 3D model

Lighting the 3D model

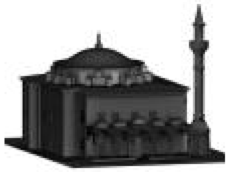
- 1) Importing the CAD (3DSMax) model together with its already specified materials and lights. Re -specifying more precisely material values.
- 2) Processing the Radiosity solution (using Lightscape) depending on the required level of detail (high)
- 3) Analysing photo metrics and adding daylight support
- 4) Extracting the 2D light maps from the 3D model using "Mesh to Texture" methods, which convert the colour per vertex mesh information, to texture light maps.



Distribution of light on the surfaces of the 3D model

Use of light maps for realistic visualisation

- 1) Utilise a global illumination simulator to create, extract and store the 2D light maps
- 2) Texture map the original unmapped 3D model, with the light maps generated in the previous stage



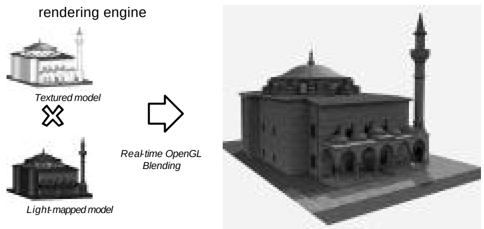
Light-maps applied on the 3D model

Real-time visualisation using open standards (no 3D game engines)

- These include from ISO C++ and OpenGL, to OpenGL Optimizer and VRML97 in any operating system that supports the above (currently under Windows PC and IRIX SGI systems).
- For the particle systems simulation we used the freely available Particle Systems API by integrating it into our OpenGL Optimizer framework whilst creating the VRML97 extension counterpart so that the various ef fects can be scripted easily and efficiently.
- The 3D virtual environments are still modelled using professional commercial 3D modelling packages for maximum precision and accuracy.
- In our methodology we describe how we pre-calculate static lighting information and store it in 2D textures called light maps. These are then used to modulate the pixel information of the ordinary 2-D texture maps, as both are blended in real-time using OpenGL's alpha unit and a standard dual-pass multitexturing technique.

Multi-pass Rendering of the 3D model

- Both textured and light-mapped models are exported in VRML and blended in real-time for interactive visualisation on MIRALab's real-time rendering engine



Final model of the edifice

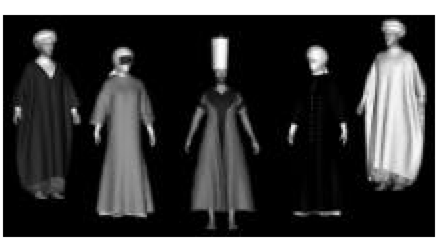
Use of Multi-pass multitexture in the 3D reconstruction

- The **Multi-pass multitexture rendering** is a technique that builds up a final rendered image from a 'combination' of the results of a set of separate rendering passes.
- These rendering passes involve pixel-by-pixel operations and in general they can be exemplified by masking an image with a binary mask-image or, as in our case, blending two images together.
- To blend the two maps together in real-time, the OpenGL alpha blending unit is utilised. During blending, the colour values of the RGBA image already stored in the frame buffer (called destination) are combined with the incoming RGBA image (called source) in a two-stage process.
  - » screen = polygon * src + screen * dst

The results (I)

The results (II)

"Generation of 3D Single Clothed-Mesh, Animatable and Deformable Virtual Humans "



Istanbul, Ottoman Empire, 15th 16th Century

By Marit LIPKA

The Production Methodology

1. Creation of Virtual Human Heads and Merging them with already Created Virtual Human Bodies and Clothes in order to Obtain one Unified and Optimized Model

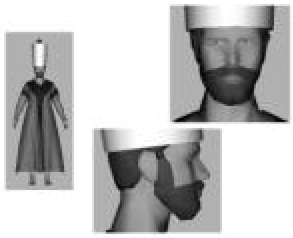
   Enhancing the Visual Impact of the Selected Virtual Humans with Texture Mapping, Material Editing, etc...
2. Attaching the Unified Model (Cloth + Mesh) to the Underlying Skeleton
3. Exportation of the Virtual Human in VRML97 H-Anim Format
4. Perform Tests with Specific Animation Sequences on the Efficiency of the Attachment Process .

   (E.g. for some animations the model exhibits cracks on the mesh or self collisions, that should be minimized through careful design modification on the results of stages 1,2)

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

1.Modeling

Vizir model



www.miralab.unige.ch
University of Geneva

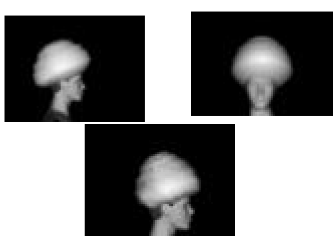
MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

Modeling the Head

www.miralab.unige.ch
University of Geneva

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

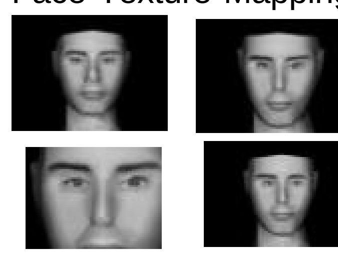
Modeling the Turban



www.miralab.unige.ch
University of Geneva

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

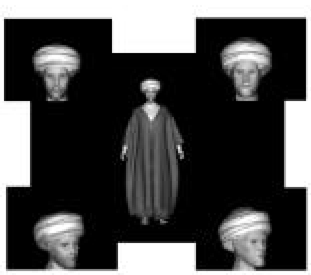
Face Texture Mapping



www.miralab.unige.ch
University of Geneva

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

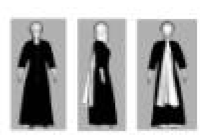
Adding Character to the Male Models

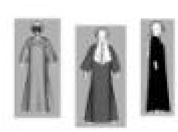


www.miralab.unige.ch
University of Geneva

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

Adding Character to the Female Models

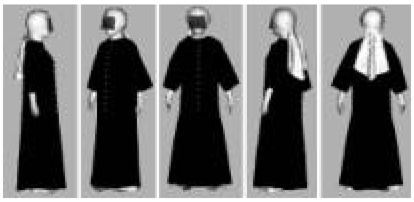




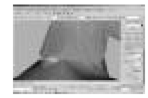
Enlarging the clothes,
Shorten veil, etc…

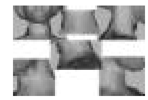
www.miralab.unige.ch
University of Geneva

Adding Character to the Feminine Models

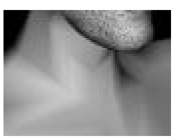
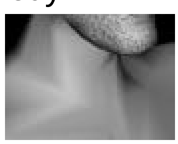


Attaching the Head to the Body

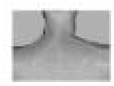


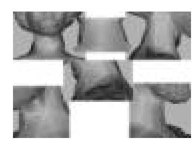
Attaching the Head to the Body

Neck: Correction of the Shading Difference

Polygon Adjustments



Polygon Adjustments

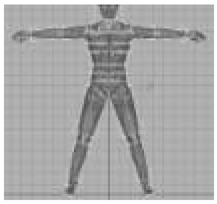



2. Attaching the Unified Model (Cloth + Mesh) to the Underlying Skeleton

Skeleton (Importation and Placement)
Comparing the Parameters to Other Skeletons and Correction
Adjustment of Rotation Points and Bone Shapes
Positioning the Body with the Skeleton:
Entering the Body into the Clothes' Shape
Deleting Polygons (Body and Cloth)
Polygon Corrections
Unifying Body and Cloth to one Single Mesh and Attachment to the Skeleton
Adjustment of Parameters (*Falloff* and Vertex Inclusion/ Exclusion)

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

The Skeleton

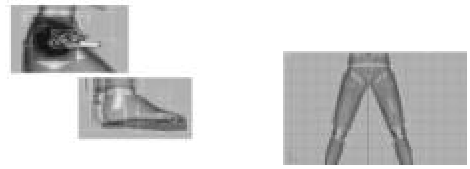


Importation

www.miralab.unige.ch
University of Geneva

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

The Skeleton



Positionning: Mesh into Skeleton

www.miralab.unige.ch
University of Geneva

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

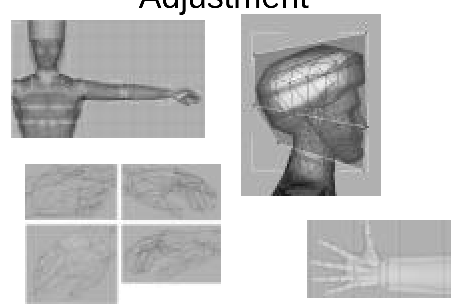
Adjustment

The Adjusted Skeleton

www.miralab.unige.ch
University of Geneva

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

Adjustment

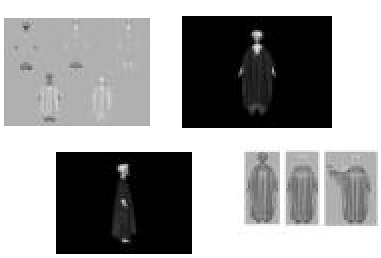


Adapting the Shape of Each Bone to the Body

www.miralab.unige.ch
University of Geneva

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

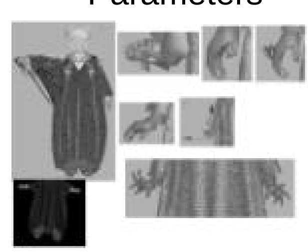
Unifying Body and Cloths to one Single Mesh and Attachment to the Skeleton



www.miralab.unige.ch
University of Geneva

MIRALab
Where Research means Creativity
EG'2002 Virtual heritage
Nadia Magnenat-Thalmann

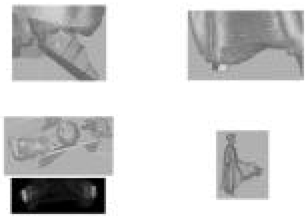
Parameters



Detachment of the Hands

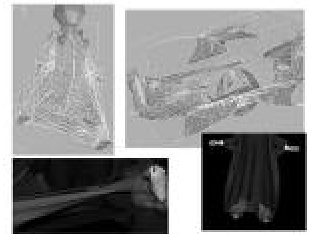
www.miralab.unige.ch
University of Geneva

Parameters



Detachment of the Trousers

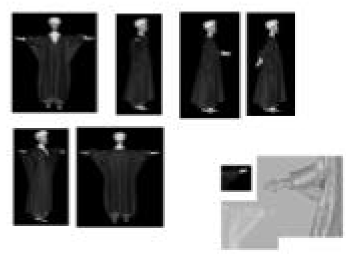
Parameters



Parameters

Detachment of the Coat

Parameters



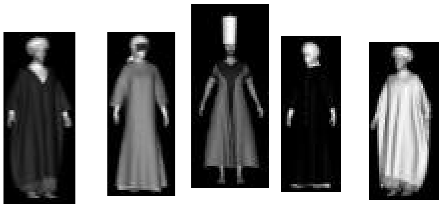
Problems with large Cloths

Final Steps on the production pipeline

3. Exportation of the Virtual Humans in VRML97 H-Anim Format

4. Tests with Specific Animation Sequences on the Efficiency of the Attachment Process

5. Process Iteration to achieve the desirable result



Facial Animation

From Facial Mesh to Expressive

Talking Faces

By

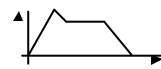
Nadia Magnenat-Thalmann

Overview

- Hierarchy in Facial Animation
- Definition of Static Expressions
- From Expressions to Animation
- Speech Animation Overview

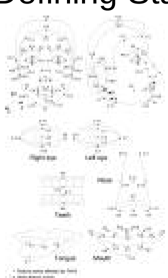
Hierarchy in Facial Animation

- Face Object : Collection of mesh vertices and topology
- Static Expressions : Deformation of this mesh controlled by parameters



- Animation : Varying the static expressions with time

Defining Static Expressions



Need of Parameterization to define static expressions
MPEG-4 Facial Animation Parameters

Feature Points defined on the Specific locations of the face

Animation defined by the displacements of these Feature Points

Designing Facial Expressions

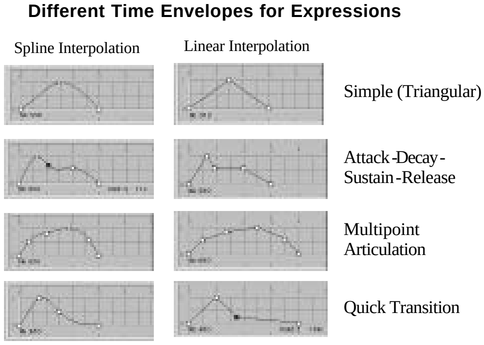
Facial Animation Parameters divided into three groups

•Lips : Lower inner midlip, Stretch corner lip etc.

•Eyes : Close right eyelid, Raise left eyebrow etc.

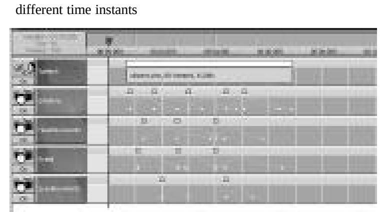
•Other : Puff right cheek, Roll head etc.

Different Time Envelopes for Expressions



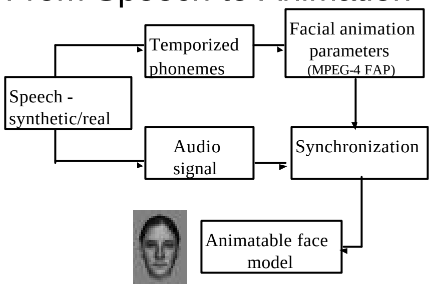
Building Animations

Possibility to add different expression envelopes at different time instants



Different animation tracks enables the designer to design head movements, facial expressions, eyebrow movements independently

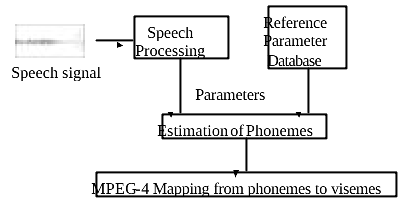
From Speech to Animation



From Natural Speech to Visemes

Extracting parameters from speech that are related to mouth shapes

Parameters : LPC, pitch, zero crossing



Towards Interactive Real-Time Crowd Behavior Simulation

Branislav Ulicny and Daniel Thalmann
Virtual Reality Lab, EPFL, CH-1015 Lausanne, Switzerland

Abstract. Virtual crowds are becoming common in non-real-time applications, however the real-time domain is still unexplored. In this paper we discuss challenges of such simulations, especially the need to efficiently manage variety. We introduce the concept of levels of variety. Then we present our work on crowd behavior simulation aimed at interactive real-time applications such as computer games or virtual environments. We define a modular behavioral architecture of a multi-agent system allowing autonomous and scripted behavior of agents supporting variety. Finally we show applications of our system in a virtual reality training and a virtual heritage reconstruction.

Keywords: autonomous agents, crowd simulations, levels of variety, multi-agent systems, virtual environments, virtual heritage, virtual reality training systems

1. Introduction

In the recent years virtual crowds became a more and more common element of our cinematic experience. Whether it was a photo-realistic crowd of digital passengers in *Titanic*, a legion of animated ants in *AntZ*, or a rendered army of droids in *Star Wars*, computer graphics (CG) generated crowds of characters added significantly to the impact of the movies employing them, allowing to visualize scenes not possible only some years ago. Crowds of CG human and non-human characters help to overcome prohibiting costs and complexities of working with large number of extras, stand as virtual stuntmen in dangerous shots and integrate well with the virtual scenes.

The situation is different, however, in the realm of real-time applications: CG crowds are still rare in computer games, virtual reality educational or training systems. Out of several thousands of titles produced every year only few employ larger number of characters. Several sport games include spectator crowds with very simple behaviors, in Rockstar Games's *State of Emergency* a virtual mob provides background to this action game and Elixir Studios's *Republic The Revolution* features crowds of virtual citizens.

Because of different requirements and constraints in almost every aspect (such as character generation and rendering, or motion and behavior control), different approaches are needed compared to the relatively simpler domain of non-real-time crowds used in motion pictures. In addition to the common tasks of managing the variety and

Figure 1. Crowd in the virtual world

easing control of multiple characters, real-time applications need to handle interactivity and have to deal with the limited computational resources available.

Several works in different fields have been exploring issues connected to the domain of crowd simulations. In his pioneer work (Reynolds, 1987) described distributed behavioral model for simulating aggregate motion of a flock of birds. Bouvier and Guilloteau (1996) used combination of particle systems and transition networks to model human crowds in visualization of urban spaces. Brogan and Hodgins (1997) simulated group behaviors for systems with significant dynamics. Aubel and Thalmann (2000) introduced dynamically generated impostors to render virtual humans. Tecchia et al. (2002) proposed image-based method for real-time rendering of animated crowds in virtual cities. O'Sullivan et al. (2002) described crowd and group simulation with level of details for geometry, motion and behavior. McPhail et al. (1992) studied individual and collective actions in temporary gatherings. Still (2000) used mobile cellular automata for simulation and analysis of crowd evacuations. However, only few works such as (Musse and Thalmann, 2001; Ulicny and Thalmann, 2001) tried to explore more general crowd models integrating several sub-components such as collision avoidance, path-planning, higher-level behaviors, interaction or rendering.

In this paper we present our current work on the real-time crowd simulation system built upon (Ulicny and Thalmann, 2001). We model a crowd as a multi-agent system with emphasis on individuals (in contrast to groups in (Musse and Thalmann, 2001)). We define a layered modular behavioral architecture allowing autonomous and scripted behavior of the agents supporting a variety, based on the combination of rules and finite state machines for higher level behavior computation, and path-finder and collision-avoidance for lower-level motion control.

The structure of the paper is as follows. First we explore assumptions and challenges for a real-time interactive crowd simulations, especially compared to non-real-time crowd and real-time single agent systems, followed by the analysis of the levels of variety. Then we present the overview of the model of the world, with a more detailed description of the behavior model. We briefly discuss the implementation of the system, and finally before concluding, we present two case studies, a virtual reality training system and a virtual heritage reconstruction, employing our crowd simulation system.

2. Assumptions and Challenges

With increasing speed of computers more complex simulations are becoming possible: nowadays it is feasible to display scenes containing hundreds of thousands polygons at interactive rate. Main challenge is becoming how to bring life to these scenes, how to animate virtual objects in a persuasive way.

Simulating human beings is complex task at every level. Ideally we would like to have complete model of the world indistinguishable from the reality running in the computer in the real-time. With the current state of the art it is obviously not possible (and it is questionable if it ever will be (Lloyd, 2000)), therefore we need to focus on such aspects of the reality that allow to sufficiently model subset of the world for the application to be useful. We need to select target space and time scale and resolution of our simulations: for example if modeled scenario is taking place during five minutes it will use probably much more detailed model than scenario lasting five hours, however than it cannot be expected to give reasonable results when forced to be run for longer time.

In our work we focus on multi-agent virtual human simulations able to run in real-time with 3D visualization allowing user interaction such as computer games, shared virtual worlds, or VR training systems. This narrows area of our interest to the situations observable in real-time as they happen, excluding for example studies of crowd behavior happening over larger time periods not localized in the exact space common in sociology (McClelland, 1989).

Our target simulations bring different challenges compared to the systems either involving small number of interacting characters (e.g. majority of contemporary computer games), or non-real-time applications (e.g. crowds in movies, visualization of crowd evacuation after off-line model computation).

In comparison with single-agent simulations the main conceptual difference is the need for efficient variety management at every level, whether it is visualization, motion control, animation or sound rendering. As everyday experiences hint, virtual humans composing crowd should look different, move different, react different, sound different and so forth. Even if assuming perfect simulation of single virtual human would be possible, still creating simulation involving multiple such humans would be difficult and tedious task. Methods easing control of many characters are needed, however such methods should still preserve ability to control individual agent.

In comparison with non-real-time simulations, the main technical challenge is increased demand on computational resources whether it is CPU time or memory space. Fast and scalable methods to compute behavior, able to take into account inputs not known in advance, are needed.

3. Levels of Variety

One of the issues arising in the importance with increasing number of simulated entities is the question of their variety (Merriam-Webster's Dictionary defines variety as "the quality or state of having different forms or types"). Even subtle variations on the motions or look of the individual virtual humans can greatly enhance realism of the virtual crowd as a whole. For analysis of the crowd simulation systems and their components it is useful to be able to define degree of variety more precisely.

We introduce the notion of levels of variety: we say that system has level of variety zero (LV0) if for a given task it is using only single solution, level of variety one (LV1) if it is able to make choice from finite number of solutions (here it can be useful to distinguish another sublevel LV1+ if solution is composed of combinations of sub-solutions), and level of variety two (LV2) if it is able to use solutions out of infinite number of possible solutions.

LV0 systems can be relatively easy upgraded to LV1 with adding meta-layer able to select one solution out of defined set; LV2 systems need generative models. In practice LV1 and LV2 systems can be perceptually indistinguishable as it is always possible to inject more pre-defined solutions into LV1 system, however this is feasible only for a small number of required solutions.

Let us consider example of crowd visualization: system where the virtual crowd is composed of only one type of the human would be LV0, system where the crowd is composed of multiple humans selected

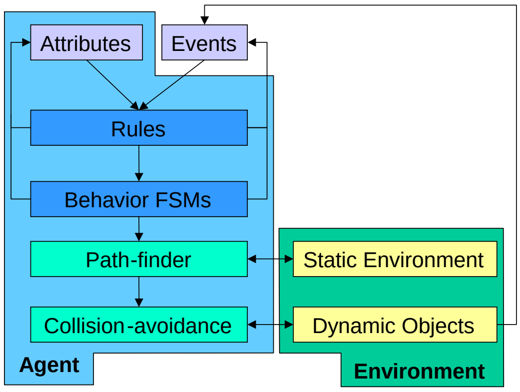


Figure 2. The model of the world

from pre-defined set would be LV1 (or alternatively LV1+ if these humans would be composed of set of exchangeable parts such as heads, bodies, textures) and finally system would be LV2 if it would be able to display potentially infinite number of unique humans generated for example by parametrizable anthropometric model generating humans with different morphologies (Seo et al., 2002).

The challenge with applying most of the classical artificial intelligence or computer graphics approaches to the domain of crowd simulations is that they were not designed with the aim of achieving variety. For example usual goal of path-finding algorithms is to find best solution (that is LV0) how to get from one place in some environment to another one. Using such ideal path without further modification for multiple entities, however, would result in unwanted artefacts of the entities moving in the queues.

We would like to note, however, that even while the real world exhibits large variety, systems with lower level of variety are justifiable, sometimes even preferable over the systems with higher variety. For example when simulating emergency egress of a crowd from a burning building, perfect visualization could be distracting; simpler uniform visualization of the individuals could help to emphasize problems with the flow of fleeing people.

4. The Model of the World

Our simulations consist of autonomous virtual human agents existing in dynamic virtual 3D environment. Model (see Figure 2) is composed of the agents, dynamic objects and static environment. We distinguish between static part of the environment like layout of the streets and

Figure 3. Crowd using simple humanoid visualization

buildings and dynamic part consisting of objects that can change their position or state during the scenario like fire or gas cloud.

Parts of the model can, but don't necessarily have to, have visualization counterparts. For example invisible objects such as exits can convey semantics of the environment. We discern between behavior displaying part of the virtual human agent (further refered to as a virtual human) and behavior controling part (refered to as an agent). The same simulation can be run using different visualizations: Figures 3 and 4 show the same scenario (crowd in the park) using LV0 simple humanoids and LV1 complex humanoids with deformable bodies as visualizations.

With the most complex visualization agents have 3D graphics body representations, which are able to perform certain low-level actions, such as playing of pre-recorded body animation sequences (such as gestures or changes of postures), walking to specified location with different gaits (Boulic et al., 1997), displaying facial animation sequences (Goto et al., 2001), looking at specified places or playing 3D localized sounds. Higher-level behaviors are then composed of particular combinations of these low-level actions.

Agents contain set of internal attributes that can correspond to various psychological or physiological states needed to model particular scenario (such as memory, fear, mobility, or level of injuries), set of higher-level complex behaviors (such as wander, flee or follow path)

Figure 4. Crowd using complex humanoid visualization

and set of rules determining selection of these behaviors. Agents can interact with both static and dynamic parts of the environment.

Interaction between the agents and the static environment is done by shared path-finder module which allows agents to move around the scene in the correct way. Waypoints on the path are not defined as exact locations, but as random places from some epsilon surrounding of a path node, therefore ensuring LV2 variety of the individual trajectories of more agents following the same path.

Interaction between the agents and the dynamic objects is done via exchanges of events, simulating either physical interactions (e.g. effects of the fire on the agent, or the agent on the fire) or perceptual interactions (e.g. agent perceiving danger in specified distance to the threat). Further events serve also as a means of inter-agent communication (e.g. agent which is injured sends message requesting for help) and user-agent interaction where user can send events (such as order to stop) to virtual humans by user interface.

5. Behavior Model

In order to behave in believable way agents must act in accordance with their surrounding environment, be able to react to its changes, to the other agents and also to the actions of the real humans interacting with the virtual world. We need a model connecting perception of the agents with their actions.

Our aim is to have behavior model that is simple enough to allow for real-time execution of many agents, yet still sufficiently complex to provide interesting behaviors. Considering requirements mentioned in Section 2 we proposed following model (see Figure 2) based on the combination of rules and finite state machines (FSM) for determining agents' behaviors using layered approach.

At the highest level, rules select high-level behaviors according to the state of the agent constituted by attributes and the state of the virtual environment conveyed by events. The rules consist of three parts:

1. **selection part** - for *who* (e.g. particular agent, or agents in particular group),
2. **condition part** - *when* the rule is applicable (e.g. at defined time, after receiving event, when some attribute reaches specified value or any boolean combination of such conditions),
3. **consequent part** - *what* is the consequence of rule firing (e.g. change of agent's behavior or attribute, or sending the event).

The reason for splitting usually single antecedent part into two sections is the optimization of the rule-base use, where the condition part is evaluated only for relevant agents. According to our experiences it is more practical to store all the rules for all the agents in the single rule-base (as opposed to each agent keeping own set of rules). In such way it is easier to maintain consistency of the rules.

Example of the rule from the actual simulation (see Section 7) is:

```
FOR  ALL
WHEN EVENT = in_danger_area AND ATTRIBUTE fear > 50%
THEN BEHAVIOR FLEE
```

Variety of the reactions to the same situation is achieved by different agents having different values of the attributes (at the beginning through different initializations, later because of their different histories), which consequently leads to different rules triggered.

At the middle level, high-level behaviors are implemented using hierarchical finite state machines. Each behavior is realized by one FSM which drives selection of the low-level actions for virtual human (like move to location, play short animation sequence), manages connections with the environment (like path queries, or event sending) and also can call other FSMs to delegate subtasks such as path following.

There are two types of high-level behaviors. First we can specify scripted behavior which is more precise, but less autonomous and with less environment coupling by using explicit sequences of low-level actions. Or second we can let agents perform autonomously complex behaviors with feedback from the environment. Examples of such autonomous behaviors are wandering, fleeing, neutralizing the threat, or help requesting and providing. Both types can be mixed as needed.

At the lowest level, motion control is also organized into layers (see Figure 2). As the result of higher-level behavior, agent's behavior FSM decides (or is told directly by the rule) that agent wants to move to particular location. This request is forwarded to the path-finding layer, which constructs sequence of waypoints that need to be passed to get to the location. Finally it is responsibility of the collision-avoidance layer to move agent between waypoints correcting trajectory in order to avoid collisions.

6. System Implementation

In addition to the requirements posed by the nature of targeted simulation, development of the complex system is by itself becoming complex task bringing additional requirement for highly modular and flexible architecture with ability to exchange sub-components and test them separately.

Our system is designed with the clear separation of the model part (where the behavior is computed) from the visualization part (where the behavior is displayed). Every component of the model of the world (agents, objects, environment) is independent from its visualization: the model can be run without any graphical output. Such organization allows to use different representation of the virtual humans, objects and environments according to the needs of particular application or in the different stages of the software life-cycle. For example simplified visualization (see Figure 3) proved to be very helpful in shortening the development cycles. In extreme case there doesn't have to be any visualization at all, for example crowd module can be used to control external entities in distributed simulation (see Section 7).

Another advantage of such design is that it allows to use different update rates for different components: higher-level behavior computation can be run with much lower frequency (and consequently much smaller CPU time slice) than lower-level motion control, which again needs lower frequency update then refreshing of the screen image.

Desirable side effect of this organization is that our crowd system is platform independent - it started as the monolithic application on SGI,

Figure 5. Crowd reacting to the emergency event

continued as the module of the distributed system based on standard PC, and currently it is the part of the multi-platform virtual reality development framework. In all the applications not just conceptual architecture, but most of the actual implementation code of the model is the same.

This architecture also addresses variety of the animations issue (see Section 2) by separating action selecting part (behavior FSM) from action executing part (virtual human controller). Behavior FSM is ordering controller to do type of the animation (such as waving of hand) and controller then randomly chooses particular one from the set of such animations with the random variation of the speed of the playback, so that even if more agents are executing the same behavior they don't necessarily act exactly the same.

7. Case Study: Crowds in Virtual Reality Training System

In this section we will present application of our crowd simulation in the virtual reality training system called CROSSES (CROwd Simulation System for Emergency Situations). CROSSES is the training system for emergency situations. Its aim is to train people to efficiently react to emergency situations such as occurrence of a fire or leakage of a poisonous gas in the town with proximity of a chemical factory.

Training sessions take place in a virtual world populated by virtual humans that have complex behaviors dependent on the events in the simulation (see Figure 5). The virtual 3D urban environment including buildings and trees is reconstructed from the aerial images (Straub

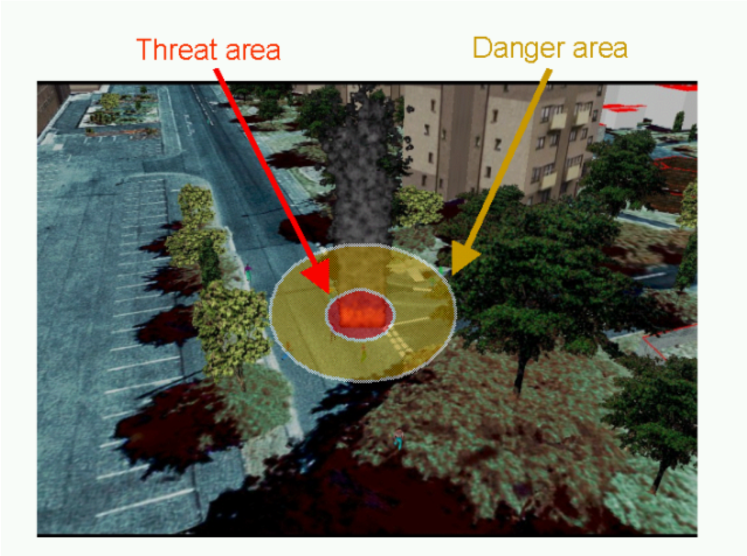


Figure 6. Threat, semantic environmental object

et al., 2001) of the actual town. Areas accessible for walking have to be specified, allowing path-finder to construct correct paths (Farenc et al., 1999). Realistic 3D virtual population of people with different professions (such as workers, firemen, policemen or paramedics) is constructed by automatic low-cost modeling (Seo et al., 2002).

The system is designed as a distributed application with components able to run on different computers and communicating by exchanging messages over the network.

CROSSES project aims to reproduce scenarios of the urban emergency situations involving crowds of virtual human agents with behaviors based on the behaviors of the real persons in such situations. The goal of the crowd module is to provide real-time approximation of the given behaviors before, during and after emergency situation happens with the possibility of real human participants of the simulation (such as trainee or trainer) affecting the outcome of the scenario by their actions. The objective is to confront human participants with the real-time 3D reconstruction of the given scenario which is plausible enough to be useful in the training process, that is providing trainee with enough information to be able to assess the situation and eventually also to make the decisions influencing scenario in desired way.

In modeling the behavior we focus on reactivity to the scenario events such as occurrence of a fire or a gas leak and interactivity towards the users, for example reacting to the "Stop" command given by the user interface.

Figure 7. Crowd entering the mosque

We can illustrate autonomous behaviors on the example of the agent fleeing from the threat. When the threat (environmental semantic object) becomes active by scenario or by user interface, it sends events to the affected agents. Perception of the danger is simulated by threat emitting events for the agents inside its danger area (see Figure 6). For particular agent the rule eventually becomes activated and “Flee” behavior is triggered. Agent selects the way out of danger by using the environmental object called “Exit” and starts running toward the exit area. After passing through the exit, agent returns to the normal behavior.

8. Case Study: Crowds in Virtual Heritage Reconstruction

Another application using our crowd system is a reconstruction of a virtual heritage site. The aim of CAHRISMA (Conservation of the Acoustical Heritage by the Revival and Identification of the Sinan’s Mosques’ Acoustics) project is to create integrated 3D audio-visual system to conserve architectural heritage including both acoustical and visual characteristic of the Sinan’s mosques and Byzantine churches. Realism of the reconstructed mosques can be increased by recreating life inside architectural models.

CAHRISMA application is built with VHD++ real-time development framework (Ponder et al., 2002).

The goal of the crowd module is to simulate a crowd of virtual humans able to move and interact within real-time photo-realistic simulation of the complex buildings (Papagiannakis et al., 2001). In this case our focus is on ability to control the scenario and quality of the animation. Crowd module allows construction of a different scenarios involving scripted behaviors of the virtual humans such as letting group of the worshippers enter the mosque, walk toward the area designated for praying and then to start praying sequence (see Figures 7, 8).

One of the challenges arising in this application was the orchestration of the convincing animation sequences. From the scenario came requirement of people performing synchronously different steps of the praying sequence. Observations of the real world praying sequence tell, however, that synchronicity is not perfect, there are small variations in the timing. Because it was not feasible to record each animation for each member of the crowd individually, single sequence of motion captured animation had to be reused. It was the task of the crowd module to create illusion of the variety avoiding mechanic look of the crowd, where each member is performing exactly the same motion at exactly the same time.

We used the events and rules of the crowd system in conjunction with the ability to change the speed of the motion sequence for synchronization and desynchronization of the animation. After arriving to the area designated for the praying, agents receive events telling them to start with the first part of the praying.

```
FOR  GROUP worshipers
WHEN EVENT = start_pray_1
THEN SCRIPT
       WAIT RANGE 0.0 2.0
       PERFORM_ACTION Pray1
       SEND_EVENT ready_to_pray_2 TO AGENT leader
```

The rule (see above) becomes triggered at the same time for every agent, however because they take different time to react, they start with the animation action asynchronously. Varying speed of the animation clip for individual members of the crowd brings additional increase in the realism. After finishing the animation, agents send event announcing end of the step to one agent having function of the synchronization. After all agents finished this step, the leader emits command to proceed to the next one. In such way the next step of the praying sequence doesn't start till everybody doesn't finish the previous one.

Figure 8. Crowd performing praying sequence inside the mosque

9. Conclusions and future work

This paper presented our work on the crowd simulation system aimed at the real-time applications. First we discussed challenges of the real-time crowd simulations, especially the need to efficiently manage variety. We introduced the notion of the levels of variety. We defined modular behavioral architecture for multi-agent simulations allowing management of the variety for crowd simulation. multi-layer behavior control model supporting crowd simulation. We demonstrated validity of our approach on two case studies, where our crowd system was used to manage crowd behavior in virtual reality training system and virtual heritage site reconstruction.

For the future work we plan to continue with enhancing the levels of variety: for the animations we want to employ motion models able to synthesize variations of the given motion (Lim and Thalmann, 2002). Another possible extension is to improve behavior model by incorporating new behaviors based on the sociological observations from the real world gatherings.

Acknowledgements

We are grateful to Mireille Clavien and Alessandro Foni for the design of the virtual humans and the mosque; Jan Ciger and Marcello Kallmann for their work on path-finding and collision-avoidance code; Michal Ponder, George Papagiannakis and Tom Molet for their support with VHD++ framework. This work has been supported by the Swiss

National Research Foundation and the Federal Office for Education and Science in the framework of the European projects CROSSES and CAHRISMA.

References

Aubel, A. and D. Thalmann: 2000, 'Real-time Display of Virtual Humans: Level of Details and Impostors'. *IEEE Transactions on Circuits and Systems for Video Technology* **10**, 207–217.

Boulic, R., P. Becheiraz, L. Emering, and D. Thalmann: 1997, 'Integration of Motion Control Techniques for Virtual Human and Avatar Real-Time Animation'. In: *Proc. VRST '97*. pp. 111–118, ACM Press.

Bouvier, E. and P. Guilloteau: 1996, 'Crowd Simulation in Immersive Space Management'. In: *Proc. Eurographics Workshop on Virtual Environments and Scientific Visualization '96*. pp. 104–110, Springer-Verlag.

Brogan, D. and J. Hodgins: 1997, 'Group Behaviors for Systems with Significant Dynamics'. *Autonomous Robots* **4**, 137–153.

Farenc, N., R. Boulic, and D. Thalmann: 1999, 'An Informed Environment Dedicated to the Simulation of Virtual Humans in Urban Context'. In: *Proc. Eurographics'99*. pp. 309–318, Blackwell.

Goto, T., S. Kshirsagar, and N. Magnenat-Thalmann: 2001, 'Automatic Face Cloning and Animation'. *IEEE Signal Processing Magazine* **18**(3).

Lim, I. and D. Thalmann: 2002, 'Construction of animation models out of captured data'. In: *Proc. IEEE International Conference on Multimedia and Expo '02*.

Lloyd, S.: 2000, 'Ultimate Physical Limits to Computation'. *Nature* **406**, 1047–1054.

McClelland, J.: 1989, *The crowd and the mob: from Plato to Canetti*. Unwin Hyman.

McPhail, C., W. Powers, and C. Tucker: 1992, 'Simulating individual and collective actions in temporary gatherings'. *Social Science Computer Review* **10**(1), 1–28.

Musse, S. and D. Thalmann: 2001, 'A Hierarchical Model for Real Time Simulation of Virtual Human Crowds'. *IEEE Transactions on Visualization and Computer Graphics* **7**(2), 152–164.

O'Sullivan, C., J. Cassell, H. Vilhjlmsson, S. Dobbyn, C. Peters, W. Leeson, T. Giang, and J. Dingliana: 2002, 'Crowd and Group Simulation with Levels of Detail for Geometry, Motion and Conversational Behaviour'. In: *Proc. Third Irish Workshop on Computer Graphics, Eurographics Ireland*. Dublin, Ireland, pp. 15–20.

Papagiannakis, G., G. L'Hoste, A. Foni, and N. Magnenat-Thalmann: 2001, 'Real-Time Photo Realistic Simulation of Complex Heritage Edifices'. In: *Proc. Virtual Systems and Multimedia 2001*. pp. 218–227.

Ponder, M., G. Papagiannakis, T. Molet, N. Magnenat-Thalmann, and D. Thalmann: 2002, 'VHD++ Real-Time Development Framework Architecture: Building Flexible and Extendible VR/AR Systems with Reusable Components'. In: *Proc. ACM Symposium on Virtual Reality Software and Technology 2002*. Hong Kong, China. (submitted).

Reynolds, C.: 1987, 'Flocks, Herds, and Schools: A Distributed Behavioral Model'. In: *Proc. SIGGRAPH '87*. pp. 25–34.

Seo, H., L. Yahia-Cherif, T. Goto, and N. Magnenat-Thalmann: 2002, 'GENESIS : Generation of E-Population Based on Statistical Information'. In: *Proc. Computer Animation 2002*. IEEE Press.

Still, G.: 2000, 'Crowd Dynamics'. Ph.D. thesis, Warwick University.

Straub, B.-M., M. Gerke, and A. Koch: 2001, 'Automatic Extraction of Trees and Buildings from Image and Height Data in an Urban Environment'. In: *Proc. International Workshop on Geo-Spatial Knowledge Processing for Natural Resource Management*. pp. 59–64.

Tecchia, F., C. Loscos, and Y. Chrysanthou: 2002, 'Image-Based Crowd Rendering'. *IEEE Computer Graphics and Applications* **22**(2), 36–43.

Ulicny, B. and D. Thalmann: 2001, 'Crowd simulation for interactive virtual environments and VR training systems'. In: *Proc. Eurographics Workshop on Animation and Simulation*. pp. 163–170, Springer-Verlag.

Archaeological Visualisation
The need for realism

- Computer Graphics allow virtual environments to be "constructed" on a computer in a straightforward manner
- Computer reconstructions can be easily misleading
- Realism is *essential* if we are to provide an insight into how these sites may have appeared

Realistic flame

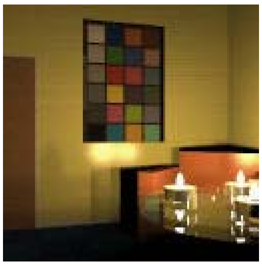
oil

oil+water

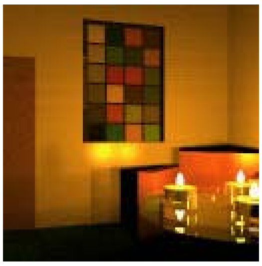
oil+salt

Spectral readings of different fuel types

Different fuel types



Modern lighting



Tallow candle

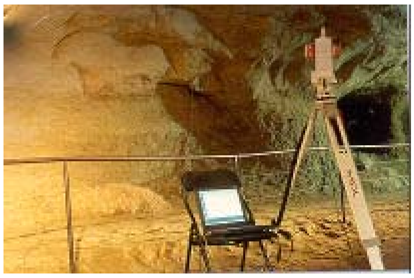
Olive oil

With salt

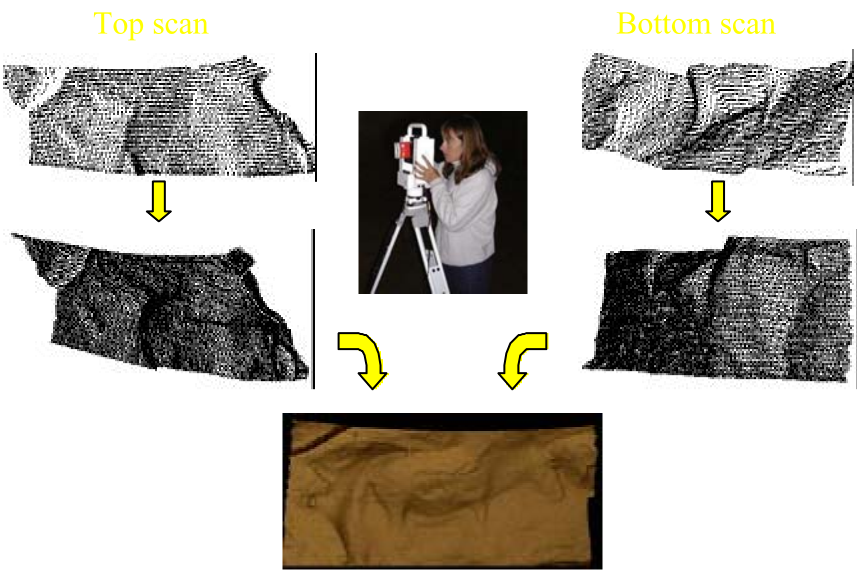
With water

Cap Blanc

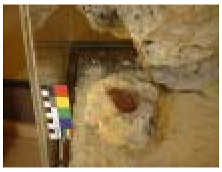
Capturing the data at Cap Blanc



From depth map to 3D model



Reconstructing Cap Blanc



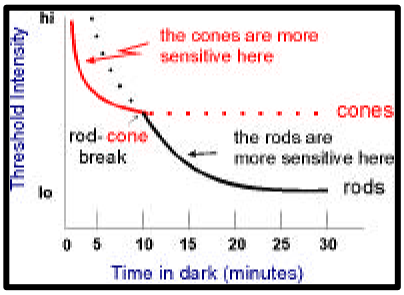
Realistic Lighting

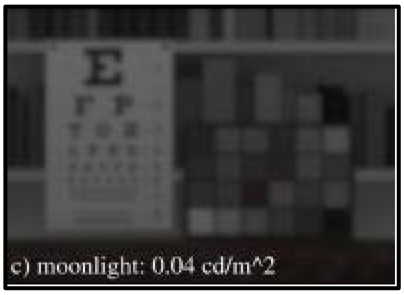
Modern lighting

Animal tallow candle

Visual Adaptation

Ferwerda et al, 1996





Heritage sites offer significant challenges to computer graphics

- Media participation
  - accurate flame and smoke
- The need for realism
  - laser scanning, psychophysics
- Complexity of environments
  - parallel processing, visual perception
- Multi-disciplinary nature
  - archaeology, psychology, engineering, art history
- Multi-sensory
  - acoustic rendering

AUGMENTED REALITY TUTORIAL

P. Fua
VRLab
EPFL, Lausanne
http://vrlab.epfl.ch/

OUTLINE

1. Introduction to AR
2. AR devices
3. Registration
4. Occlusions
5. Illumination
6. AR for real and virtual humans

VIRTUAL REALITY

GOAL:
- Create realistic synthetic environment

APPLICATIONS:
- Animation
- Flight simulators
- Medical simulation

APPROACH:
- Build geometric model
- Postulate physical models
- Compute synthetic images

→ **Images must be believable and realistic but do not need to match reality exactly.**

AUGMENTED REALITY

GOAL:
- Annotate a real scene
- Add new information
- Help to understand

APPLICATIONS:
- Head-up display and annotation
- Special effects
- Surgery and medicine
- Training
- Architecture

APPROACH

- Build precise models
- Match against real data
- Augment them
- Merge real and synthetic images

→ **Models must match reality as closely as possible**

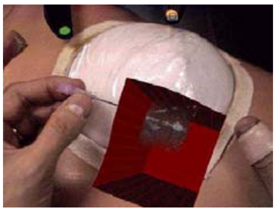
ANNOTATION

SPECIAL EFFECTS

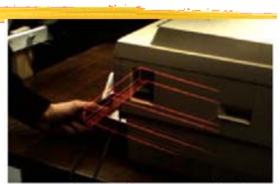
Integrate virtual elements into the real scene

GAMES

MEDICAL APPLICATIONS



TRAINING



ARCHITECTURE

OPTICAL DISPLAY

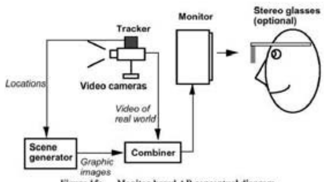


Figure 15: Monitor-based AR conceptual diagram

SEE THROUGH HEAD MOUNTED DISPLAY

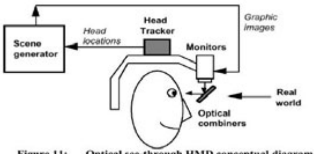
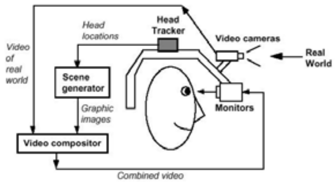


Figure 11: Optical see-through HMD conceptual diagram

Figure 12: Two optical see-through HMDs, made by Hughes Electronics

VIDEO SEE THROUGH HEAD MOUNTED DISPLAY



ADVANTAGES / DRAWBACKS

An optical approach has the following advantages over a video approach:

- Resolution
- Safety
- No Eye Offset
- Simplicity

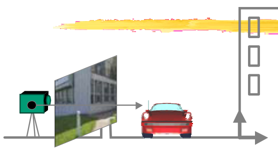
ADVANTAGES / DRAWBACKS

Video blending offers the following advantages over optical blending:

- Flexibility in composition strategies
- Wide field-of-view
- Real and virtual view delays can be matched
- Additional registration strategies
- Brightness matching of real and virtual objects

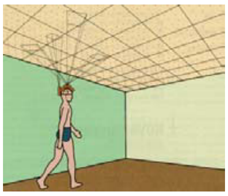
AUGMENTED SCENES

PROBLEMS TO ADDRESS



1. Register the scene, i.e find the position, orientation, and internal parameters of the real camera.
2. Find occlusions between real and virtual objects.
3. Compute lighting parameters wrt to the real scene.

REGISTRATION



THE REGISTRATION PROBLEM

The objects in the real and virtual worlds must be properly aligned with respect to each other, or the illusion that the two worlds coexist will be compromised. [**A Survey of Augmented Reality** Ronald T. Azuma]

The error between the real and virtual has to be a fraction of a degree of arc. (a full moon view is 0.5 degrees of arc).

CALIBRATION

CAMERAS:
- Internal Camera Parameters
- External Camera Parameters

→ Standard photogrammetry techniques

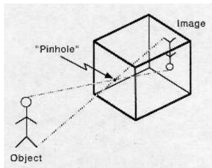
MEDICAL SENSORS:
- Matching referentials

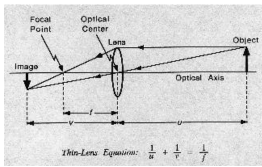
3D POINTERS:
- Calibration of tracking system

→Measuring points with a known position

HEAD MOUNTED DISPLAYS
- Head and gaze tracking

PINHOLE CAMERA MODEL





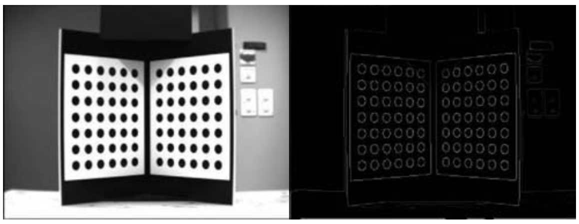
Idealized model that defines perspective projection:
- All rays go through a hole and form a star of lines
- The hole allows the formation of an inverted image
- A whole line projects to a single point

CAMERA PARAMETERS

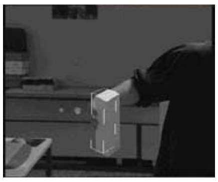
Internal parameters
- Focal length (1)
- Principal points (2)
- Stretching (1)

External parameters
- Rotations (3)
- Translations (3)

CALIBRATION OBJECT



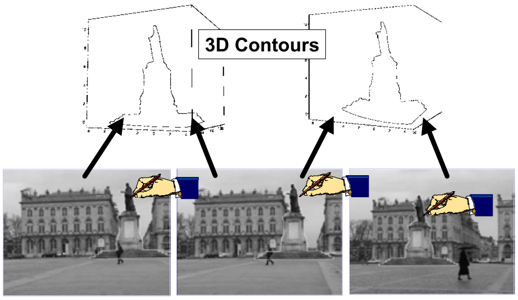
OBJECT TRACKING



At every time step, compute the motion parameters that yield the correct projection.

OCCLUSIONS

OCCLUSION MASK

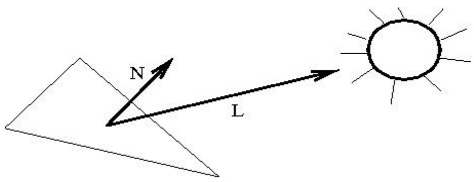


Key Frame 1 Key Frame 2 Key Frame 3

DIMINISHED REALITY

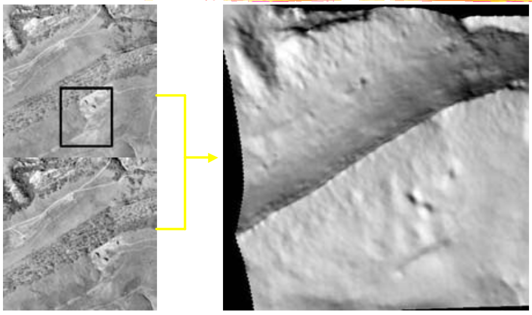
ILLUMINATION

LAMBERTIAN MODEL



$$I_{ij} = a_i L_{ij} + b_i$$
$$L_{ij} = \alpha_j (l_i^t \cdot n_j + \mu_i)$$

ILLUMINANT RECOVERY



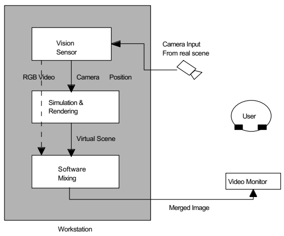
DE-LIGHTING AND RE-LIGHTING

Must explicitly model specularities, shadows, self-reflections, etc …

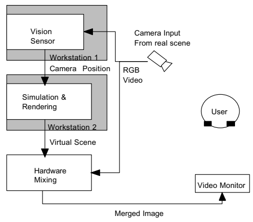
REAL AND VIRTUAL HUMANS

Real-time tracking of the Board
Detecting motions of the pieces
Animating the Virtual Human

SYSTEM OVERVIEW

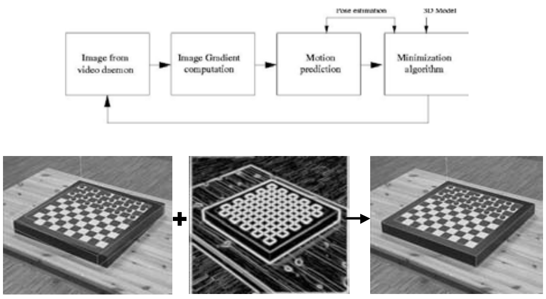


Single

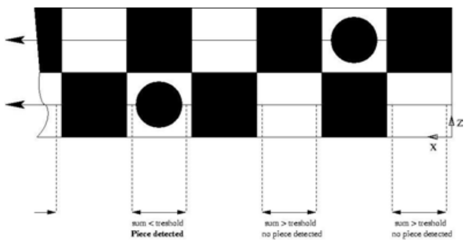


Dual

TRACKING



PIECE DETECTION



CONCLUSION

Augmented reality has numerous applications in many domains:

- Animation and special effects,
- Medicine,
- Industry,
- Architecture....

Must combine techniques from:

- Computer Graphics
- Computer Vision
- Photogrammetry

OUR AR PUBLICATIONS

1. V. Lepetit and M.O. Berger, "A Semi Automatic Method for Resolving Occlusions in Augmented Reality, IEEE Conference on Computer Vision and Pattern Recognition, Hilton Head Island, South Carolina (USA). June 2000.
2. Q. T. Luong, P. Fua, and Y. G. Leclerc. The Radiometry of Multiple Images. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(1):19--33, January 2002.
3. G. Sannier, S. Balcisoy, N. Magnenat-Thalmann, D. Thalmann. VHD: A System for Directing Real-Time Virtual Actors, The Visual Computer, Springer, Vol.15, No 7/8, 1999, pp.320-329.
4. A. Shahrokni, L. Vachetti, V. Lepetit and P. Fua. Extended version of Polyhedral object detection and pose estimation for augmented reality applications. International Conference on Computer Animation, Geneva, Switzerland, June 2002.
5. R. Torre, S. Balcisoy, P. Fua, M. Ponder, and D. Thalmann. Interaction Between Real and Virtual Humans: Playing Checkers, Interaction Between Real and Virtual Humans : Playing Checkers, Eurographics Workshop on Virtual Environments}, Amsterdam, Netherlands, June 2000.

See also http://ligwww.epfl.ch/vision/research/augm/