Ciências ULisboa | Informática

Animação e Ambientes Virtuais 2018/2019

Mestrado

DI- FCUL

Guião das aulas teóricas

Ana Paula Cláudio

1. Behavioral Animation & Interaction with 3D Objects

2. Crowd Simulation

1. Behavioral Animation & Interaction with 3D Objects

Behavioral Animation	**Interaction with 3D Objects**
Definition and examples Perception Defining behaviors Planning Artificial Neural Networks Artificial Life	Smart Objects Interaction features Intrinsic object properties Interaction information Object behavior Expected actor behavior

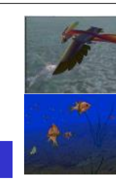
Behavioral Animation

The notion of **Behavioral Animation** was introduced by **Craig Reynolds** during the **Siggraph'87** conference:

> "Typical computer animation models only the shape and physical properties of the characters, whereas **behavior animation or character-based animation seeks to model the behavior of the character**."

Reynold's first application was to model the **flocking behaviors of birds and fish**, but the idea of behavioral animation has been applied since then to various domains.

https://www.youtube.com/watch?v=86iQiV3-3IA



Behavioral Animation

In **Behavioral Animation**, Virtual Characters acquire the ability to **perceive** their environment and are **able to react and take decisions**, depending on this input (it is important to note that **agents** need to be situated in a common environment: otherwise, no interaction could be possible).

"An **agent** is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors (actuators).

A **human agent** has eyes, ears, and other organs for sensors, and hands, legs, mouth, and other body parts for effectors.

A **robotic agent** substitutes cameras and infrared range finders for the sensors and various motors for the effectors.

A **software agent** has encoded bit strings as its percepts and actions."

In:
Artificial Intelligence: A Modern Approach
by Stuart Russell, Peter Norvig

http://www.cs.berkeley.edu/~russell/aima1e/chapter02.pdf

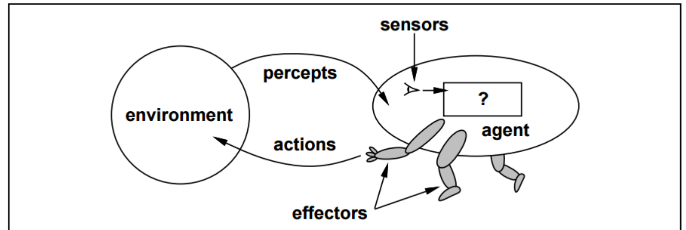


Figure 2.1 Agents interact with environments through sensors and effectors.

http://www.cs.berkeley.edu/~russell/aima1e/chapter02.pdf

Obs: effector ~ actuator

Behavioral Animation

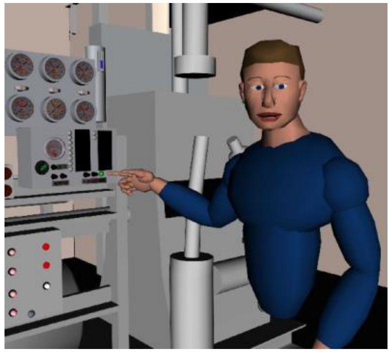
Applications of behavioral animation

Entertainment industry:
- games
- movies (e.g., "The Lord of the Rings")

Training environments:

Training situations in several environments (e.g. STEVE (**S**oar **T**raining **E**xpert for **V**irtual **E**nvironments)

Behavioral Animation

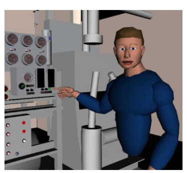


STEVE= Soar **T**raining **E**xpert for **V**irtual **E**nvironments

Steve's role is to teach students how to perform procedural tasks, such as operating or repairing complex devices.

http://www.isi.edu/isd/VET/vet.html

Behavioral Animation




STEVE = Soar **T**raining **E**xpert for **V**irtual **E**nvironments

Each Steve is assigned a role within the overall task. A Steve agent can monitor a human (or another agent) performing an assigned role.

http://www.isi.edu/isd/VET/steve-demo.html

Behavioral Animation

http://www.isi.edu/isd/VET/vet.html

STEVE = Soar **T**raining **E**xpert for **V**irtual **E**nvironments

Rickel, J., & Johnson, W.L., Task-Oriented Collaboration with Embodied Agents in Virtual Worlds. In J. Cassell, J. Sullivan, and S. Prevost (Eds.), ***Embodied Conversational Agents***. Boston: MIT Press, 2000.

Embodied conversational agent (ECA):

It is a virtual human capable of carrying on conversations with humans by both understanding and producing speech, hand gesture and facial expressions.

Embodied Conversational Agents (ECA) are a type of **multimodal interface** where the modalities are the natural modalities of human conversation: **speech**, **facial displays**, **hand gestures**, **body stance**.

http://www.justinecassell.com/jc_research.htm

Rickel, J., & Johnson, W.L., Task-Oriented Collaboration with Embodied Agents in Virtual Worlds. In J. Cassell, J. Sullivan, and S. Prevost (Eds.), ***Embodied Conversational Agents***. Boston: MIT Press, 2000.

Behavioral Animation

What is **behavior?**

- The manner of conducting **oneself**;
- Anything that an organism does involving **action** and **response to stimulation**;
- The response of an **individual**, **group**, or **species** to its **environment**.

Behavioral Animation

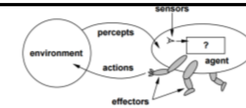
The **goal** of Behavioral Animation is not to add new animation capabilities to an agent, but **to add the capability of selecting the "best" action in a certain situation**, giving a set of all possible actions

In the **context of Behavior Animation**:

Action or **gesture** refer to the **most simple behaviors** that an agent is able to perform.

Behavior refers to more abstract capabilities such as executing a sequence of actions.

Behavioral Animation



1 Agents interact with environments through sensors and effectors.

Behavioral Animation requires various capabilities**:** the awareness of the agent starts with the simulated

- **perception of its environment** (and so the capability to **memorize** it); based on this input, the agent will:
  - **adapt its behavior**,
  - **take the proper decision** and then
  - **interact**, either with elements of the environment or with other virtual humans (when conversing, for example).

Of course, it is not easy to generate proper and convincing results without solving some animation problems which are closer to computer graphics than AI.

Behavioral Animation

Perception → **Incorporating sensors** that capture what is happening in the surrounding world

- Vision

Defining Behaviors

- Planning
- Artificial neural networks
- Artificial life
- …

Vision algorithms

There are two approaches:

- **omniscient**
  - Give information about **all the objects** in the 3D scene
  - Easier to implement
  - More efficient
  - Can produce unrealistic results
- **honest** – give information about the objects that are in the field of view of the agent
  - **Artificial vision - recognition from 2D images (image analysis)**
  - **Synthetic vision - identification of objects captured by a camera placed in the agent's head**

Synthetic vision algorithms

- False colors
- Image Recognition
- Filters
- Ray Casting

Image analysis

Image analysis is the extraction of meaningful information from images; mainly from digital images by means of digital image processing techniques.

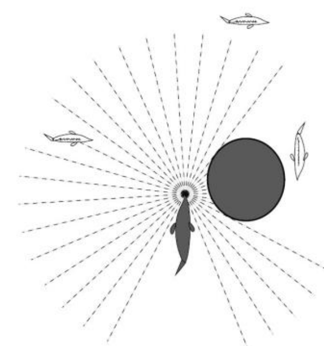
Image analysis tasks can be as simple as **reading bar coded** tags or as sophisticated as **identifying a person from their face**.

https://en.wikipedia.org/wiki/Image_analysis

Synthetic vision algorithms– ray casting

Vision of fish [Tu94]

The vision sensor covers an angle of 300° in a **spherical volume** with a predetermined radius



To check if a point P is visible:

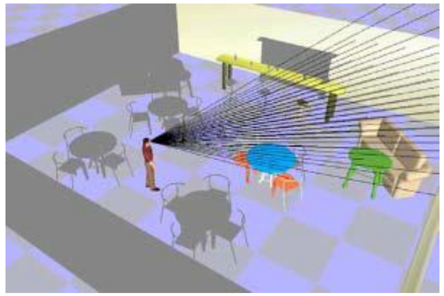
- Cast a ray from the origin of the referential towards point P
- If P is inside the view volume

and it is not occluded by other object, then,

P is visible

Synthetic vision algorithms– ray casting

SimHuman
[Vosinakis05]



- Rays are cast into the scene within the field of view of the agent
- For each ray is determined the position, the size and type of the intersected objects

Synthetic vision algorithms– ray casting

Vision algorithm in the **IViHumans Plataform**
[Semião06a]

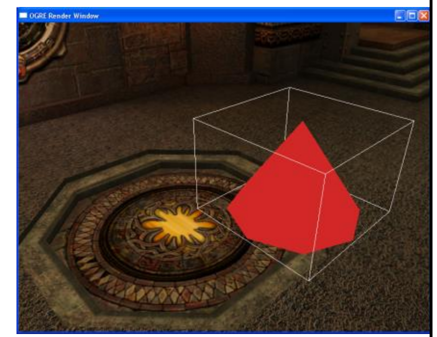
Ray cast

Detect the intersection of the rays with the bounding volumes of the objects

OpCode library– defines different bounding volumes, more or less tight to the objects' geometry.

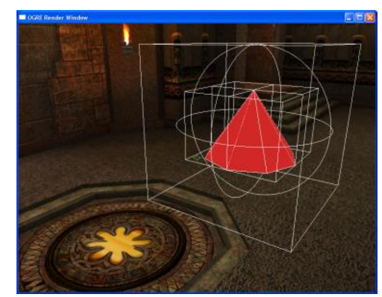
Synthetic vision algorithms– ray casting

OpCode library, example of bounding volume



Synthetic vision algorithms– ray casting

OpCode library, more examples of bounding volumes



Behavioral Animation

Perception

Defining Behaviors

- Planning
- Artificial neural networks
- Artificial life
- ...

Behavioral Animation

Defining Behaviors-
Planning

An agent determines its behavior by **reasoning** about what it knows to be true at a specific time. In the classical planning systems, **plans provide step-by-step instructions on how to satisfy some goals**; they basically require some **preconditions to be fulfilled** in order to achieve their effects. Giving a **goal** to the agent it will **trigger some plans** and generally activate some **sub-goals**.

Behavioral Animation

Defining Behaviors-
Planning

Ex:

Precondition: to be thirsty

Plane: drink a glass of water

Subplans:

- get a glass,
- fill the glass with water

Behavioral Animation

Defining Behaviors-
Artificial Neural Networks

Artificial Neural Networks (ANN) are often used for the specific case of **learning**. Inspiration for ANN comes from the desire to reproduce artificially a system into which computations are more or less similar to the **human brain**.

ANN are tightly linked to **training**: once the creature has been trained to learn something, it will tend to be able to solve this **particular** problem, but it will still **have problems dealing with new problems**.

Behavioral Animation

Defining Behaviors-
Artificial Life

Artificial Life is literally "**life made by Man rather by Nature**".

The idea is to first **model simple entities** and then, by **simulating virtual natural-life evolution** and complexifying both the entity and its environment, when time passes, **new behaviors will emerge**.
Since the entity always **repeates a sequence of three behaviors** (**perception of the environment**, **action selection** and **reaction**) it continously modifies its environment, which, in turn, influencies the entity.

1. Behavioral Animation & Interaction with 3D Objects

Behavioral Animation	Interaction with 3D Objects
Definition and examples Perception Defining behaviors Planning Artificial Neural Networks Artificial Life	Smart Objects Interaction features Intrinsic object properties Interaction information Object behavior Expected actor behavior

Behavioral Animation

The introduction of **emotional** states into the modelling of virtual humans is an important topic, which has received much attention in the behavioral animation field and has been successfully implemented in a number of systems.

However some social researchers have demonstrated that **emotions are not sufficient to model human behavior**; our **group belonging**, **gender** or **cultural origin** for example, are very important criteria to explain our behavior. Taking into account the sociological and social-psychological body of science is therefore an important challenge for the field of behavioral animation.

Interaction with 3D Objects

Among the several issues related to real-time animation of Virtual Human actors, the **ability to interact with virtual objects** require special attention in order to obtain **lifelike behaviors**.

Examples of **virtual objects**:

- a door or an automatic door
- a push button
- a lift
- a drawer

Interaction with 3D Objects

Some of the related **problems with this interaction:**

- Recognition of manipulation places
- Automatic arm and hand animation
- Motion synchronization between actors and objects

A human-like behavior would recognize a given object with the **vision** and **touch**, and then, based on **past experiences** and **knowledge**, the correct sequence of motions would be deduced and executed. Such an approach is still **too complex** to be handled in a general case, and **not suited for interactive systems** where **real-time execution is required**.

Interaction with 3D Objects- Smart objects

Smart object =

geometric information necessary for displaying it on the screen

+

semantic information useful for animation purposes

smart object (a concept proposed by Kallmann and Thalmann, 1999. SOMOD –Smart Object Modeler Application – was developed specifically to model smart objects)

Interaction with 3D Objects- Smart objects

A smart object stores all necessary interaction information within the object description. Interaction information is defined during the modeling phase, forming a complete "**user guide**" to the object.

In this way, <u>virtual actors can simply access and follow such interaction descriptions in order to accomplish a given task</u>.

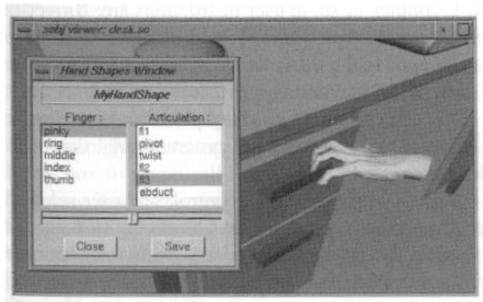
Interaction with 3D Objects - Smart objects

Interaction features can be seen as **all parts**, **movements and descriptions** of an object that have some important role when interacting with an actor.

Interaction features can be grouped into **four** different classes:

- **Intrinsic object properties**
- **Interaction information**
- **Object behavior**
- **Expected actor behavior**

Interaction with 3D Objects – Smart objects



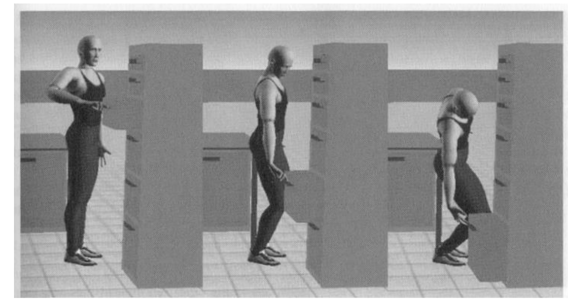
The image shows a hand shape being interactively defined in SOMOD. This hand shape will be suggested to the actor during the interaction.

Interaction with 3D Objects- Smart objects

Expected actor behavior is associated with each object behavior and it is useful to have a description of some **expected actors behaviors in order to accomplish the interaction**.

For example, **before opening a drawer, the actor is expected to be in a suitable position so that the drawer will not collide with the actor when opening**. Such a suitable position is then proposed to the actor during the interaction.

Interaction with 3D Objects- Smart Objects



http://vrlab.epfl.ch/HandbookVHumans/

Interaction with 3D Objects- Smart Objects

The **smart object** approach **introduces** the following main characteristics into a **simulation system:**

Decentralization of the animation control: Object interaction information is stored in the objects and can be loaded as plug-ins, so that **most object-specific computation is released from the main animation control**.

Reusability of designed smart objects: Not only by using the same smart object in different applications, but also by designing new objects by merging any desired feature from previously designed smart objects.

2. Crowd Simulation

Crowd Simulation

Psychologists and **sociologists** have studied the behavior of **groups of people** for several years. They have been mainly interested in the effects when **people with the same goal become only one entity**, namely a **crowd** or a mass. In this case, **persons can lose their individuality** and **adopt the behavior of the crowd entity**, behaving in a different way than if they were alone.

Some examples here:

https://bihc-fcul.weebly.com/videos-of-ai-and-crowds.html

https://bihc-fcul.weebly.com/videos.html

Crowd Simulation

Although **collective behavior** was studied as along ago as the end of the 19th century, attempts to simulate it by computer systems were mostly done only in the mid and late 1990s.

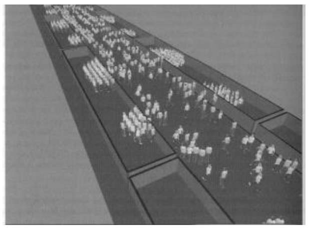
Application areas:

- Enternainment industry
- Architecture
- Digital Heritage
- Physics
- Robotics
- Safety science
- Trainning systems
- Sociology

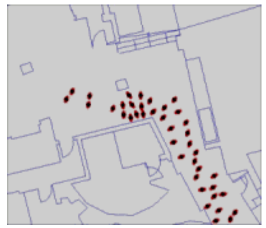
Crowd Simulation

Some exemples (safety & architecture):

These approaches are focusing on realism of behavioral aspects (sometimes using 2D visualizations).



EXODUS:
http://fseg.gre.ac.uk/exodus



Simulex:
http://www.crowddynamics.com/Egress/simulex.html

Crowd Simulation

Behavioral Simulation of Crowds

The goal is to provide **autonomous** or **semi- autonomous behaviors** that can be applied by **self-animating agents**, who form the crowd.

In order to deal with the constraints of having a **large number of actors** (hundreds or thousands), this kind of application presents the following challenges:

- **description of sociological and social aspects that arise in crowds** and
- **connection with the virtual environment**.

Also, necessary optimizations and simplifications are required in order to model and simulate **crowd intelligence** and **decision ability**.

Crowd Simulation

Crowd Motion Simulation aims at evaluating the **motion of several people in a constrained environment**.

The main challenge in this case includes a **realistic method of collision avoidance** and a **strong connection with the environment.**

Crowd Simulation

The **starting point in crowd simulation** by computer systems:

Reynolds (1987) described a distributed behavioral model to simulate the aggregate motion of a **flock** of birds. The flock is simulated as a **particle system**, with the simulated birds (called **boids**) being the particles.

Remember that **Reynolds introduced the behavioral animation concept.**

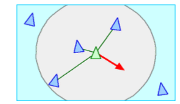
Crowd Simulation

Behavioral animation!

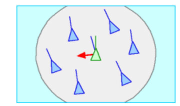
In the area of **Computer Graphics**, the goal of behavioral animation is to simplify designers' work by letting **virtual characters perform autonomously** or **semi- autonomously** complicated motions which otherwise would require large amount of human animators' work.

Basic Flocking Model

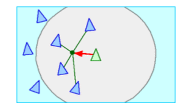
Steering behaviors:



Separation: steer to avoid crowding local flockmates



Alignment: steer towards the average heading of local flockmates



Cohesion: steer to move toward the average position of local flockmates

Each boid has direct access to the whole scene's geometric description, **but flocking requires that it reacts only to flockmates within a certain small neighborhood around itself**.

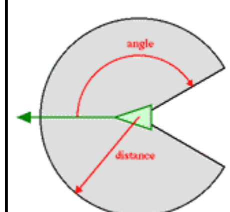
Steering Behaviors

Each boid is implemented as **an independent actor** that navigates according to its **local perception** of the **dynamics of the environment**, the **laws of simulated physics** and a set of **behaviors.**

The **basic flocking model** consists of **three simple steering behaviors** which describe **how an individual boid maneuvers based on the positions and velocities of its nearby flockmates** (next page)

http://www.red3d.com/cwr/boids/

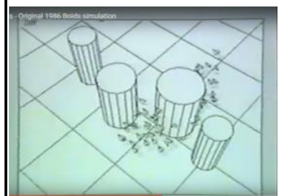
Steering Behaviors



The **neighborhood** is characterized by a *distance* (measured from the center of the boid) and an *angle*, measured from the boid's direction of flight. **Flockmates outside this local neighborhood are ignored**.

The neighborhood could be considered a model of limited perception (as by fish in murky water) but it is probably more correct to think of it as defining the region in which flockmates influence a boids steering.

Steering Behaviors



Used in the case of a **single autonomous agent**, these behaviors are fairly simple to understand and implement.

In addition, by building a system of **multiple characters** that steer themselves according to simple, locally based rules, surprising levels of complexity emerge.

Reynolds's "boids" model for "flocking/swarming" : https://www.youtube.com/watch?v=86iQiV3-3IA

Steering behaviors

Steering behaviors aim to help autonomous characters move in a realistic manner, by **using simple forces** that are combined to produce life-like, improvisational navigation around the characters' environment.

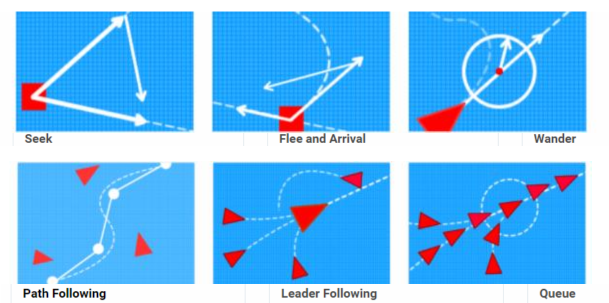
http://gamedevelopment.tutsplus.com/series/understanding-steering-behaviors--gamedev-12732

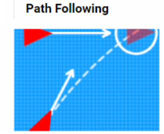
Steering behaviors

Steering behaviors are not based on complex strategies involving path planning or global calculations, but instead use local information, such as **neighbors' forces**.

http://gamedevelopment.tutsplus.com/series/understanding-steering-behaviors--gamedev-12732

Other Steering behaviors





Pursuit and Evade

http://gamedevelopment.tutsplus.com/series/understanding-steering-behaviors--gamedev-12732

Crowd Simulation- Aspects concerning Visualization

The key feature of a realistic crowd is its **diversity.**

One approach:

Example avatars. On the left, an avatar rendered with ray tracing in 3D Studio Max. In the middle, an avatar with alpha channels to identify parts to modify. On the right, an avatar rendered with multipasses regarding the alpha channel. Notice how these four avatars look different although they are built from the same 3D model.

[TLC02]

One approach:

Example avatars. On the left, an avatar rendered with ray tracing in 3D Studio Max. In the middle, an avatar with alpha channels to identify parts to modify. On the right, an avatar rendered with multipasses regarding the alpha channel. Notice how these four avatars look different although they are built from the same 3D model.

The **alpha channel** is really a mask. It specifies how the pixel's colors should be merged with another pixel when the two are overlaid, one on top of the other.

http://www.webopedia.com/TERM/A/alpha_channel.html

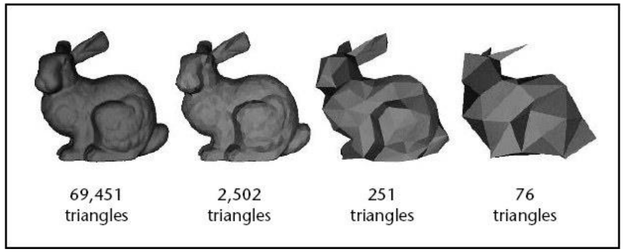
Crowd Simulation- Aspects concerning Visualization

Level of Detail (LOD)

Geometric level of detail attempts to reduce the number of rendered polygons by using several representations of decreasing complexity of an object. At each frame the appropriate model or resolution is selected.

Typically the selection criterion is the distance to the camera.

Crowd Simulation- Aspects concerning Visualization



Same model with different levels of detail (LOD)

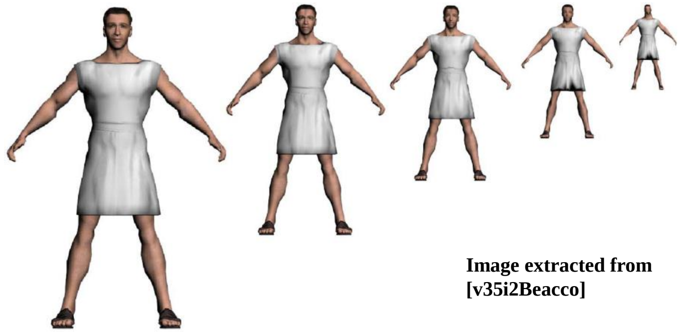
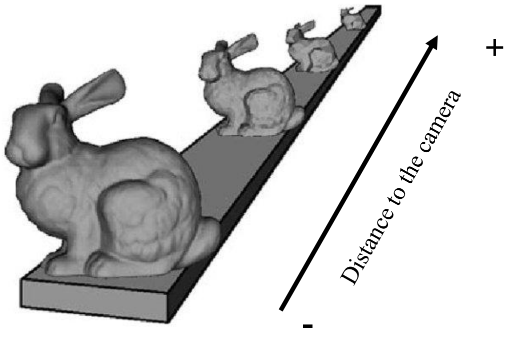


Image extracted from [v35i2Beacco]

Figure 4: *Five models of the same avatar with decreasing number of polygons as they are placed further away.*

Crowd Simulation- Aspects concerning Visualization



Example:

The recent Assassin's Creed Unity … claims to show up to 12 000 agents in real-time, although just 120 of them are rendered using high resolution.

[v35i2Beacco]

Examples (cont.):

Moving to strategy games, we can find massive armies of up to 100 000 soldiers animated in real-time, as in Creative Assembly's Total War: Rome 2. **Distant soldiers are not required to have individual appearance, animation and behaviour**, which makes it easier to reach such a high performance.

In sport games such as Electronic Arts' FIFA 15, stadiums can also be filled with up to 120 000 animated virtual spectators, although **they remain seated in the same place with no navigation or collision avoidance**.

[v35i2Beacco]

Crowd Simulation- Aspects concerning Visualization

Animated impostors

An impostor (or billboard) is in essence some very simple geometry that manages to fool the viewer.

An impostor may be a **simple textured plan that rotates to face continuously the viewer**. The image or **texture that is mapped onto this plane** is merely a **snapshot** of the virtual human.

If this **snapshot can be re-used** over several frames then the polygon complexity of the virtual human is minor.

Crowd Simulation- Aspects concerning Visualization

Animated impostors (billboards)

Impostors: simple textured-mapped primitives that replace 3D characters

Crowd Simulation- Aspects concerning Visualization

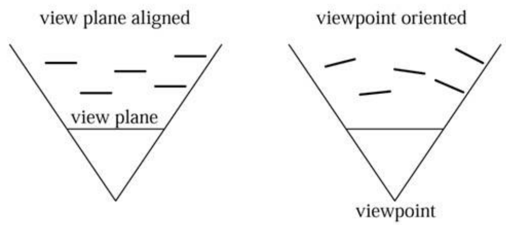
The main use of billboards is to replace complex three-dimensional models (e.g. grass, bushes, or even trees) by two-dimensional images.

(…) It is crucial that the billboard is always aligned parallel to the view plane in order **to keep up the illusion of a three-dimensional shape** although only a two-dimensional image is rendered.

https://en.wikibooks.org/wiki/Cg_Programming/Unity/Billboards

Crowd Simulation- Aspects concerning Visualization

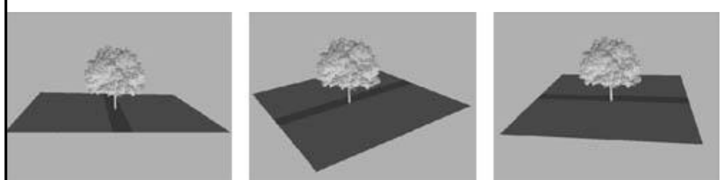
Impostors



http://www.flipcode.com/archives/Billboarding-Excerpt_From_iReal-Time_Renderingi_2E.shtml

Crowd Simulation- Aspects concerning Visualization

Impostors



http://www.flipcode.com/archives/Billboarding-Excerpt_From_iReal-Time_Renderingi_2E.shtml

Crowd Simulation- Aspects concerning Visualization

Impostors- a different approach

The plane commonly used in literature as the projection plane for an impostor is usually the one perpendicular to the view direction from which the sample image was taken.
This plane does not take into account the shape of the object nor any kind of special occlusion that could be present in the image. We then decided to try a different approach [TLC02]

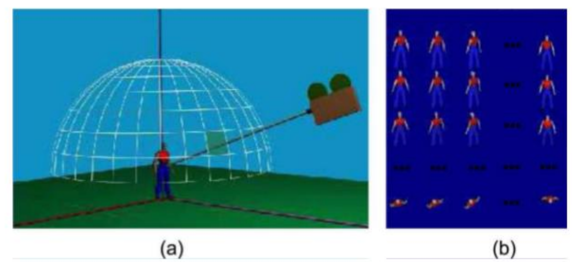
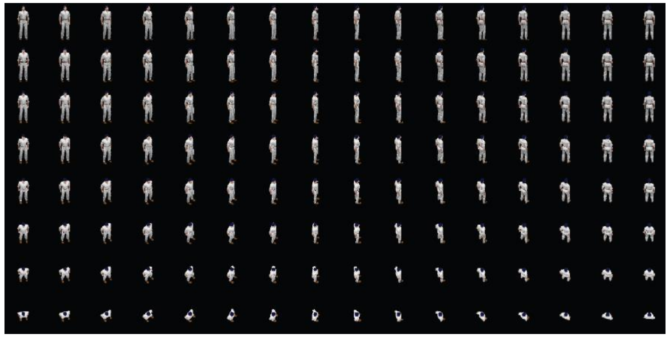


Figure 9: *Discretizing the view direction between the object and the viewpoint (a) allows to generate a texture with all the captured directions for one frame of one animation (b). The process must be repeated for every animation frame [TC00] (images courtesy of Franco Tecchia).*

[v35i2Beacco]



Impostor texture for an animation frame

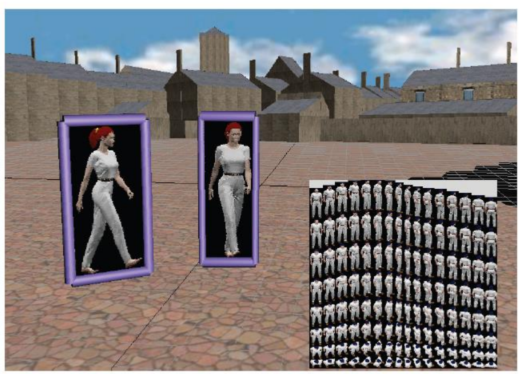
[TLC02]

Image extracted from [v35i2Beacco]

Figure 11: *Geopostors. Far agents are rendered with impostors while closer ones are rendered with geometry [DHOO05] (image courtesy of Carol O'Sullivan).*

https://www.youtube.com/watch?v=srj4liqWbik

Rendering impostors using a compressed texture. The bottom right shows the texture after compression.



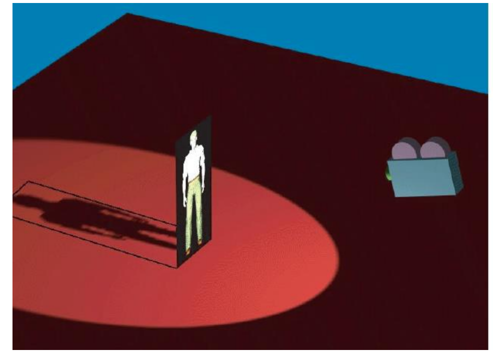
[TLC02]

Image extracted from [v35i2Beacco]

Notice the shadows

Figure 10: *A scene with shadowing pre-generated impostors [TLC02] (image courtesy of Yiorgos Chrysanthou).*

How important are the shadows?



7 A virtual human's shadow is the projection onto the ground of its silhouette as seen from a single light source.

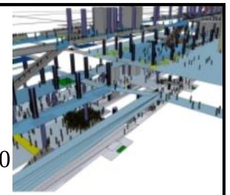
[TLC02]

How important are the shadows?

Crowd Simulation- Main Software Tools

- MassMotion
- Massive
- Miarmy

MassMotion



http://www.oasys-software.com/products/engineering/massmotion.html

https://www.youtube.com/watch?v=5EUYzDJyuQ0

"What is MassMotion used for?
There are many applications for MassMotion, it is highly adaptable to suit a wide variety of pedestrian issues at the planning stage or detailed design phase of a project.
Use it to simulate **transit operations**, **venues/special events**, and **evacuation scenarios**; **stations, airport terminals, healthcare facilities, office towers and arenas/stadiums**.
Airport Terminal Design & Planning – including curbside with vehicles, passenger movement and processing, baggage handling, groundside operations modeling
Rail & Transit Terminal Planning – large high density crowds and schedule based activity, neighbourhood dispersion including signalled road crossings
District Modelling – unparalleled scalability will enable the complete simulation of large swaths of urban areas
Fire & Evacuation Planning – validated multi-floor evacuation by stairs, evacuations including elevators"

Massive

- http://www.massivesoftware.com/feature.php
- https://www.youtube.com/watch?v=cr5Cwz-5Wsw

What is Massive?
Massive is the premier simulation and visualization solution system for generating and visualizing realistic crowd behaviors and autonomous agent driven animation for a variety of industries, including film, games, television, architecture, transportation, engineering, and robotics. Using Massive, an animator, engineer or robot developer designs characters with a set of actions and reactions to what is going on around them.

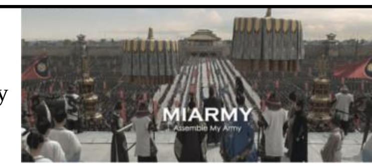
The reactions of the characters determine what they do and how they do it. Their **reactions can even simulate emotive qualities such as bravery, weariness, or joy.** The agent reactions can control key–framed or motion captured animation clips called actions.

Massive

http://www.massivesoftware.com/engineering.html

"Simulation and visualization for the real world: Architecture, civil engineering, pedestrian planning, transportation, life safety, life science – and much more."

Miarmy



http://www.basefount.com/miarmy

https://www.youtube.com/user/Basefount

Miarmy (named "My Army") is a **Human Logic Engine based Maya plugin for crowd simulation, AI, behavioral animation, creature physical simulation and rendering.** It already has been utilized by many film projects and companies. It able to help you:
•Build human fuzzy logic network without any programming and node connecting
•Adopt standard production pipeline, referencing, Human IK, motion builder support etc.
•Integrated with PhysX: ragdoll, RBD emitters, force field, cloth, wind, fluid, etc.
•Create Stunning hybrid crowd VFX, Easily combined with Maya VFX tool, particles, field and fluid
•Support all renderers, like Renderman, Mental Ray, V-Ray and is able to run on Render Farm

Note: VFX (visual effects)