RAVES Pervasive January 2005 Update

Charles E. Hughes
School of Computer Science
School of Film and Digital Media
Media Convergence Laboratory

Topics

- Algorithms
- Sensors
- Audio
- Authoring / Delivery
- PhD Student List
- Publications

Algorithms

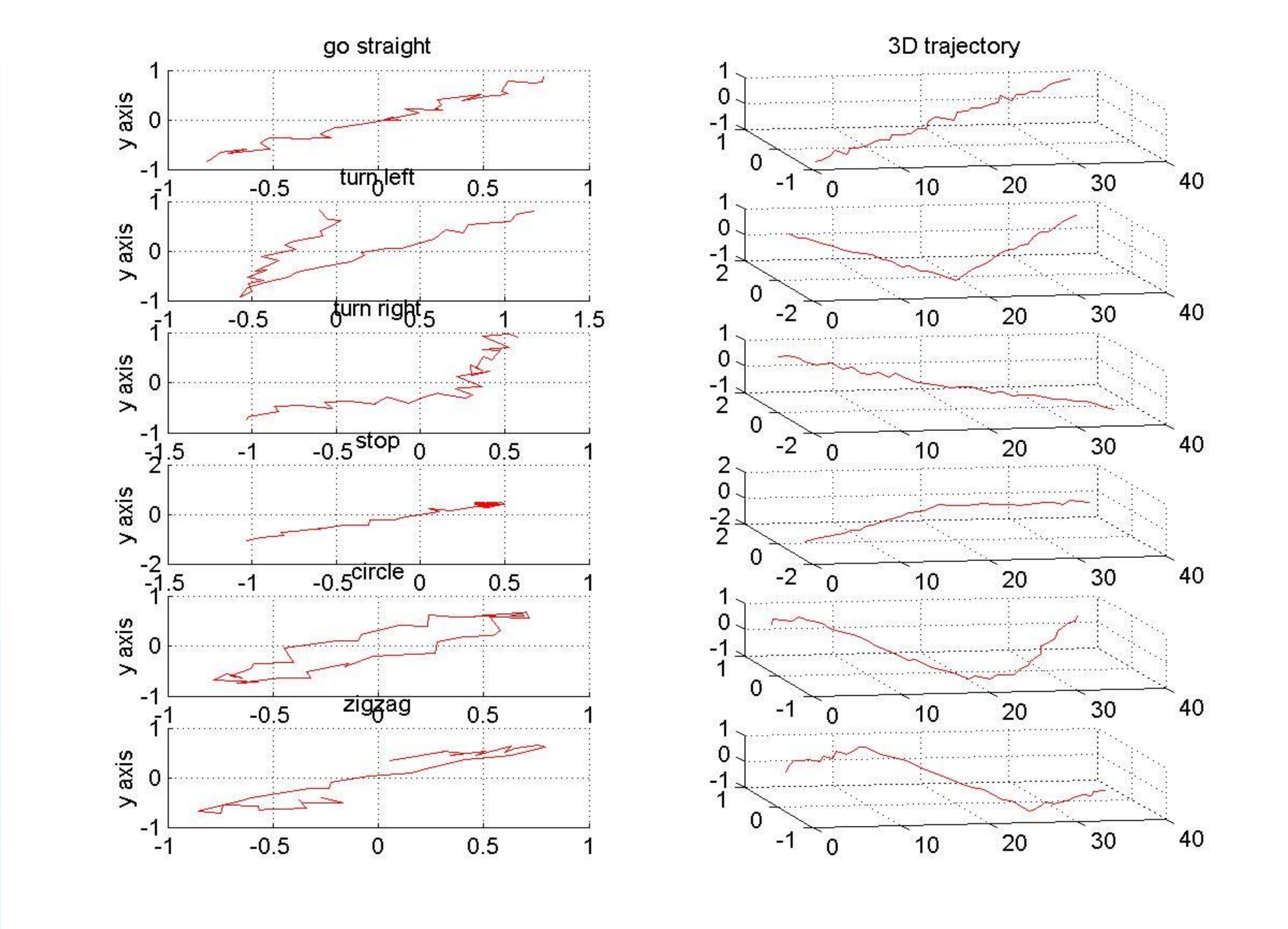
Deducing Behaviors from Primitive Movement Attributes

DanZhou Liu, CS PhD Candidate

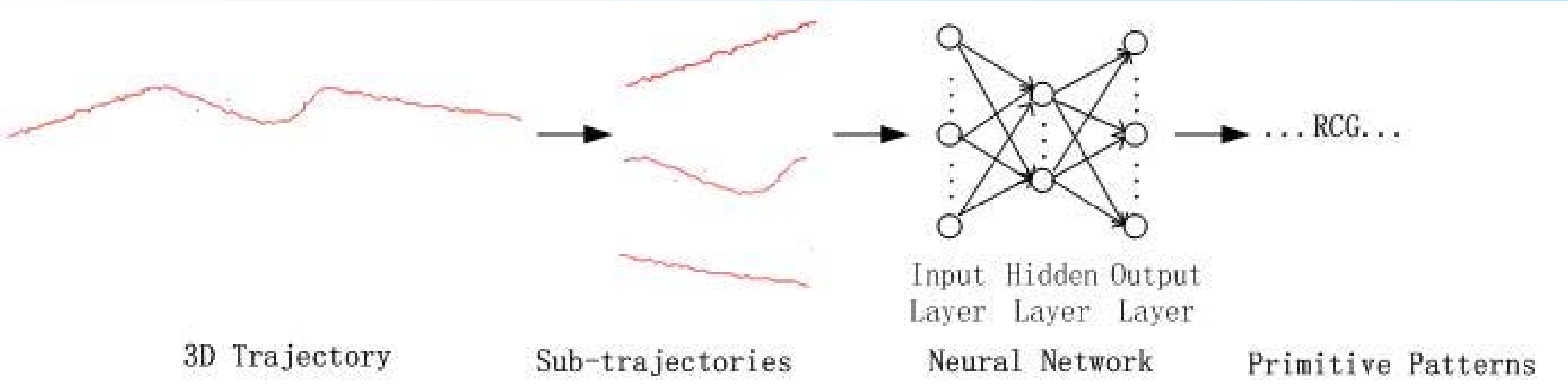
Raw to Primitive Motion

- Neural net removes body movement noise – 120,000 training sets used
- Record 5 values at each sample
  - (*interval_ID, body_movement, avg_speed, head_movement, places_passing_by*)
  - Body_movement:
    - **G**(go straight), **L**(turn left), **R**(turn right), **S**(stop), **C**(circle), **Z**(zigzag)
  - Head_Movement:
    - **F**(look forward), **L**(look left), **R**(look right), **U**(look up), **D**(look down)

Primitive Body Movement



Neural Network



Each sub-trajectory, *S,* is normalized as

$$p' = \frac{p - mean(S)}{std(S)}, \forall p \in S.$$

That is, each point is reduced by the mean of *S*, and then is scaled by the standard deviation of *S*. After normalization, each sub-trajectory is sequentially fed into the input layer, and the input layer size is determined by the number of points in *S*. The output layer size is 6 (**C**,**G**,**L**,**R**,**S**,**Z**)

Behaviors (Intentions)

- Described by regular expressions
- Two forms
  - (G+.*S+ , Φ, Φ, .*“CSB206”+)
    - walk for a while then stop; stand still near my office
  - (< G , . , . , .>+ < . , . , . , . >* < S , . , . , P >+)
    - expresses concurrency of stopping at my place

  - (G+.*S+ , Φ, .*[LRUD]+.* , .*“Records Room”+)
    - lots of head movement looks suspicious
  - (< G , . , . , .>+ < . , . , . , . >* < S , . , . , P >+) ∩ (Φ , Φ , .*[LRUD]+.* , Φ)
    - requires intersection to avoid unintended concurrency

Events

- Simple events are regular expressions
- Allow **and**, **or**, **negate**, **intersect**, etc.
- Support multiple tracked objects
- Add **concurrency**
  - Can express things like suspicious behavior due to a person changing behaviors as soon as someone else enters that person's line-of-sight.

Test Environment (Unreal Engine)

Experiment

- 4 students traversing environment 30 times
- Subjective view of whether action was to find professor’s office, be suspicious, or neither
- Used first form (independent dimensions)
- Got 80%-90% accuracy

Future

- Compare representation choices
- Extend algorithm to support multiple tracked objects with concurrency
- Use Genetic Algorithms to create populations of patterns describing intents (Annie Wu, CS Professor, plus graduate students)

Biologically Motivated String Expression Operators

Theoretical Results

- Shuffle operation (k-shuffle)
  - **$A \blacktriangleright_k B = \{ x_1y_1x_2y_2 \ldots x_ky_kx_{k+1} \mid y_1y_2 \ldots y_k \in A, x_1x_2 \ldots x_kx_{k+1} \in B, x_i, y_j \in \Sigma^*\}$**
- Undecidable Problems (new results)
  - **$L \blacktriangleright L = L$? L is a Context Free Language**
  - **$R \blacktriangleright_k L = R \blacktriangleright_{k+1} L$? R Regular, L a CFL**
  - **$L \blacktriangleright_k L = L \blacktriangleright_{k+1} L$? L a CFL**
  - and a few others

Future

- Determine complexity of reasoning about biologically motivated string operations

Color Transfer and Indoor/Outdoor Vision

James Burnett, CS PhD Candidate

Color Transfer in Lαβ Space

$$l_a^{scale} = \left(c_a^{scale} \cdot \frac{\sigma_b^l}{\sigma_a^l}\right) + \left(1 - c_a^{scale}\right)$$
$$\alpha_a^{scale} = \left(c_a^{scale} \cdot \frac{\sigma_b^\alpha}{\sigma_a^\alpha}\right) + \left(1 - c_a^{scale}\right)$$
$$\beta_a^{scale} = \left(c_a^{scale} \cdot \frac{\sigma_b^\beta}{\sigma_a^\beta}\right) + \left(1 - c_a^{scale}\right)$$

$$l_a^{boost} = c_a^{boost} \cdot \overline{l_b} + \left(1 - c_a^{boost}\right) \cdot \overline{l_a}$$
$$\alpha_a^{boost} = c_a^{boost} \cdot \overline{\alpha_b} + \left(1 - c_a^{boost}\right) \cdot \overline{\alpha_a}$$
$$\beta_a^{boost} = c_a^{boost} \cdot \overline{\beta_b} + \left(1 - c_a^{boost}\right) \cdot \overline{\beta_a}$$

$$l_a = \left(l_a - \overline{l_a}\right) \cdot l_a^{scale} + l_a^{boost}$$
$$\alpha_a = \left(\alpha_a - \overline{\alpha_a}\right) \cdot \alpha_a^{scale} + \alpha_a^{boost}$$
$$\beta_a = \left(\beta_a - \overline{\beta_a}\right) \cdot \beta_a^{scale} + \beta_a^{boost}$$

The scaling factor is computed by dividing the standard deviation of the other image by that of this one. The boosting value is the average calculated for the other image.

Discussion

- The coefficients have the effect of limiting the amount of luminance and color properties transferred from one image onto another.
- The *a* and *b* subscripts reference the images. *a* is the image being modified and *b* is the other image whose values are being “transferred” to *a*.

Variations on Color Transfer

Source Image Pair

Full Color Transfer

Half Color Transfer

Tuned Color Transfer

Overexposed Image

Underexposed Image

Adaptive Subdivision

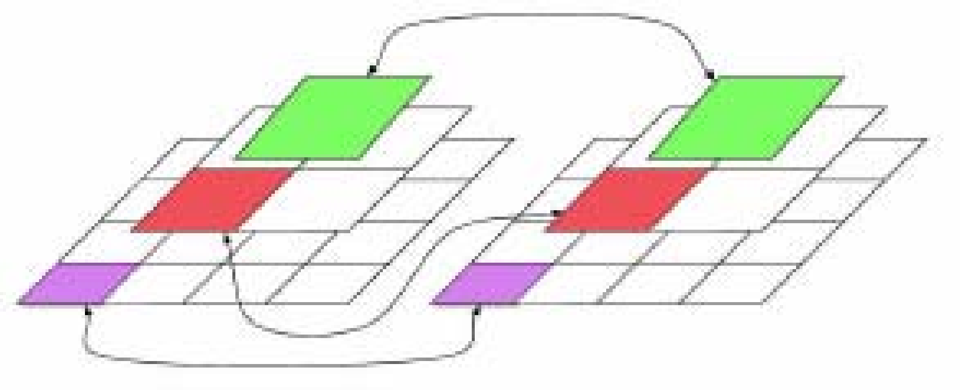


Image Pyramid

Each level has progressively lower resolution and fewer cells than the previous level. The color transfer algorithm looks at each level and each grid cell individually, viewing each cell as a separate image. In the diagram, the algorithm transfers image properties between the purple sections, the red sections, and the green sections individually. Finally, the levels of the image pyramid are combined using a simple up-scaling/ averaging algorithm to produce the final image. This gives a multi-scale view of the scene as well as preserving and incorporating global information in the resulting image.

Future

- Use an image rather than grid-based segmentation
- Slightly vary the exposure setting used for each frame. so each camera could capture a slightly larger bracket of dynamic range
- Use a third camera with automatic shutter control to provide a neutral point of view that slowly reacts to changes in environmental radiance
- Apply a filter to the algorithm that maps luminance and color information based on conformance to or divergence from this range humans observe
- Compute coefficients dynamically

Sensors

Sensor Servers

Felix Hamza-Lup
Matt O'Connor

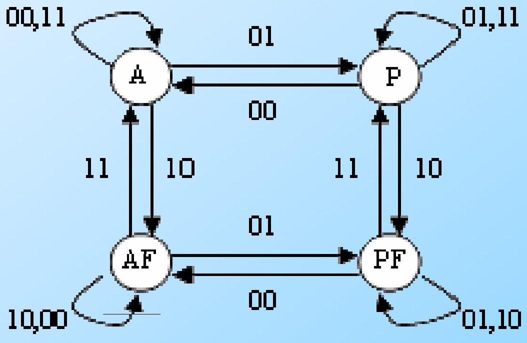
Hybrid Nodes with Sensors

Felix Hamza-Lup
CS PhD in Summer 2004

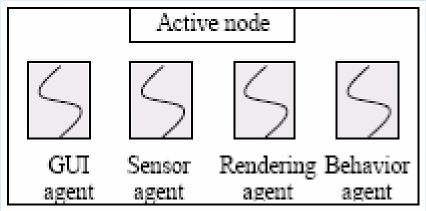
Hybrid Nodes Architecture

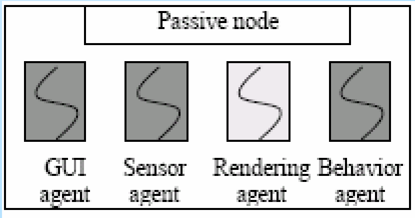
- *Passive & Active* -

- Hybrid Nodes
  - *active nodes* (A)
  - *passive nodes* (P)
  - *active forward node* (AF)
  - *passive forward node* (PF)



00 - the state changes from *inactive* to *active*
01 - the state changes from *active* to *inactive*
10 - the state change from forwarding *off* to *on*
11 - the state change from forwarding *on* to *off.*

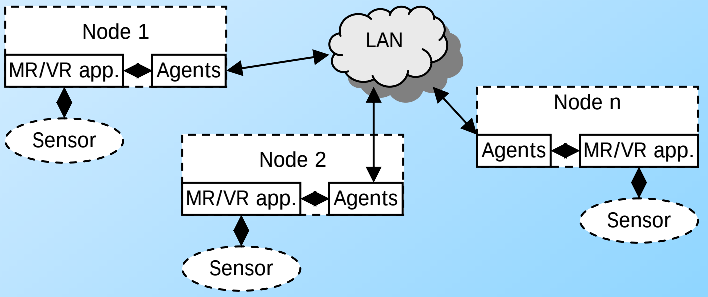




Hybrid Nodes Architecture

- System Perspective -

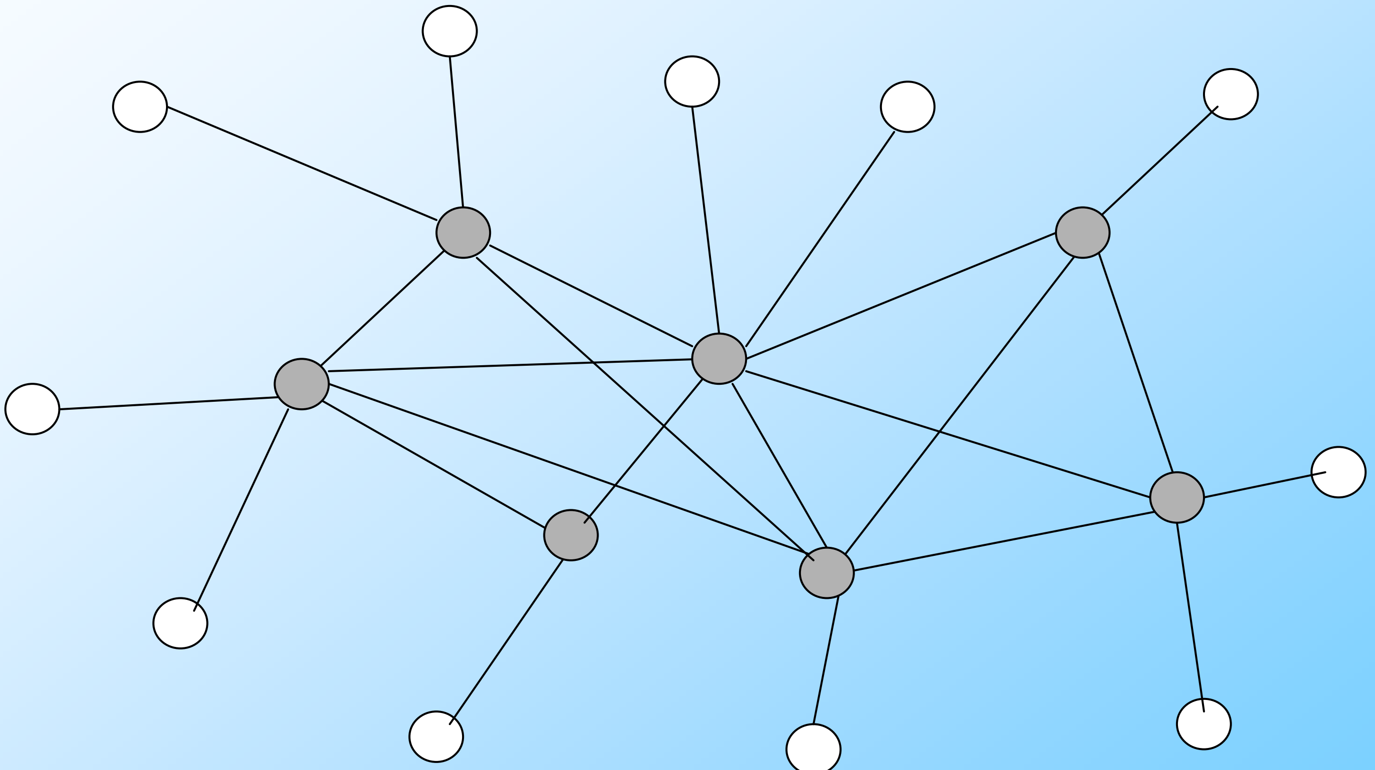
- A Distributed System Model
  - dynamic membership
  - graph representation



- Construction of a communication sub-graph
  - Core-Base Tree algorithm
    - guarantees that the path between any node and the core is the shortest
    - only one tree per multicast group

Hybrid Nodes Architecture

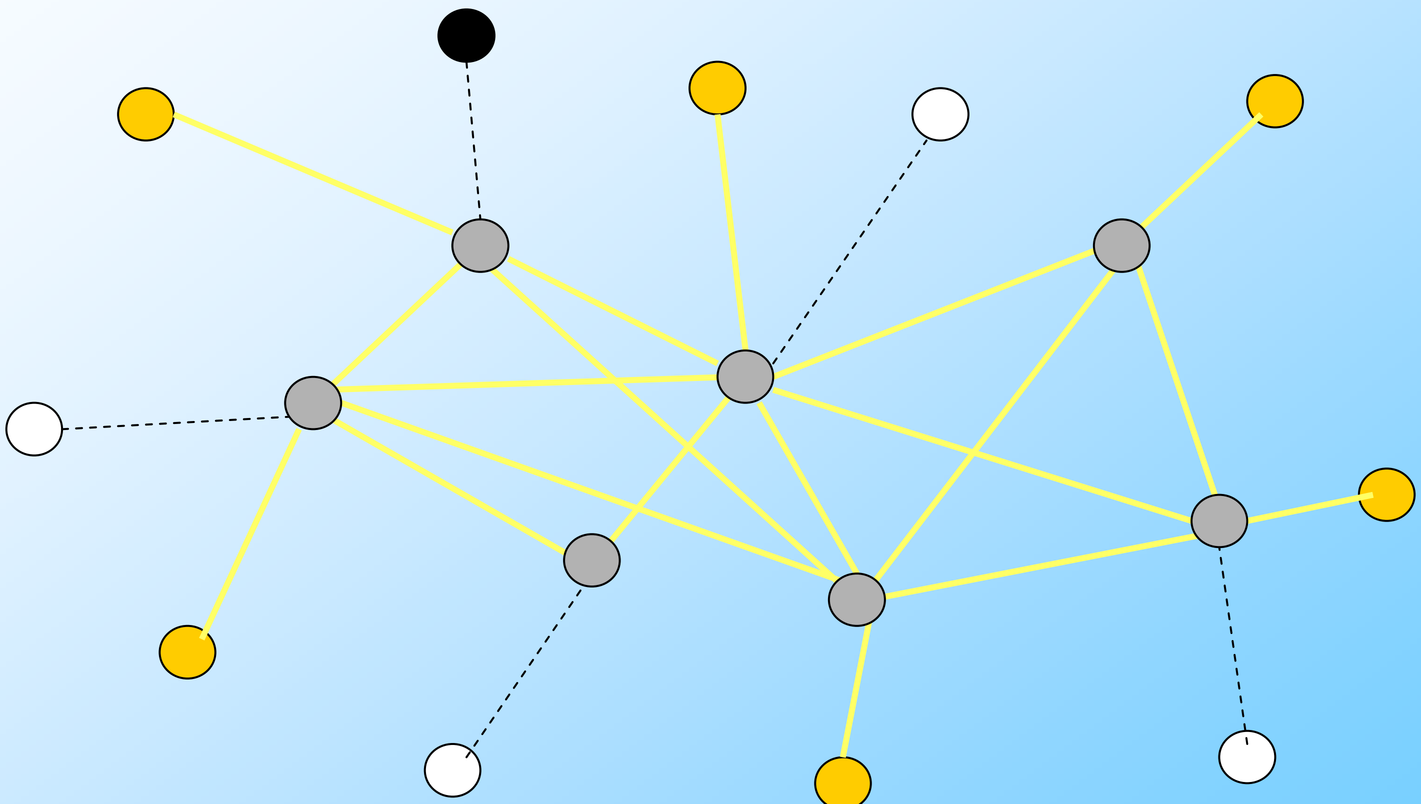
- Infrastructure Instance -



G=(V,E)

Hybrid Nodes Architecture

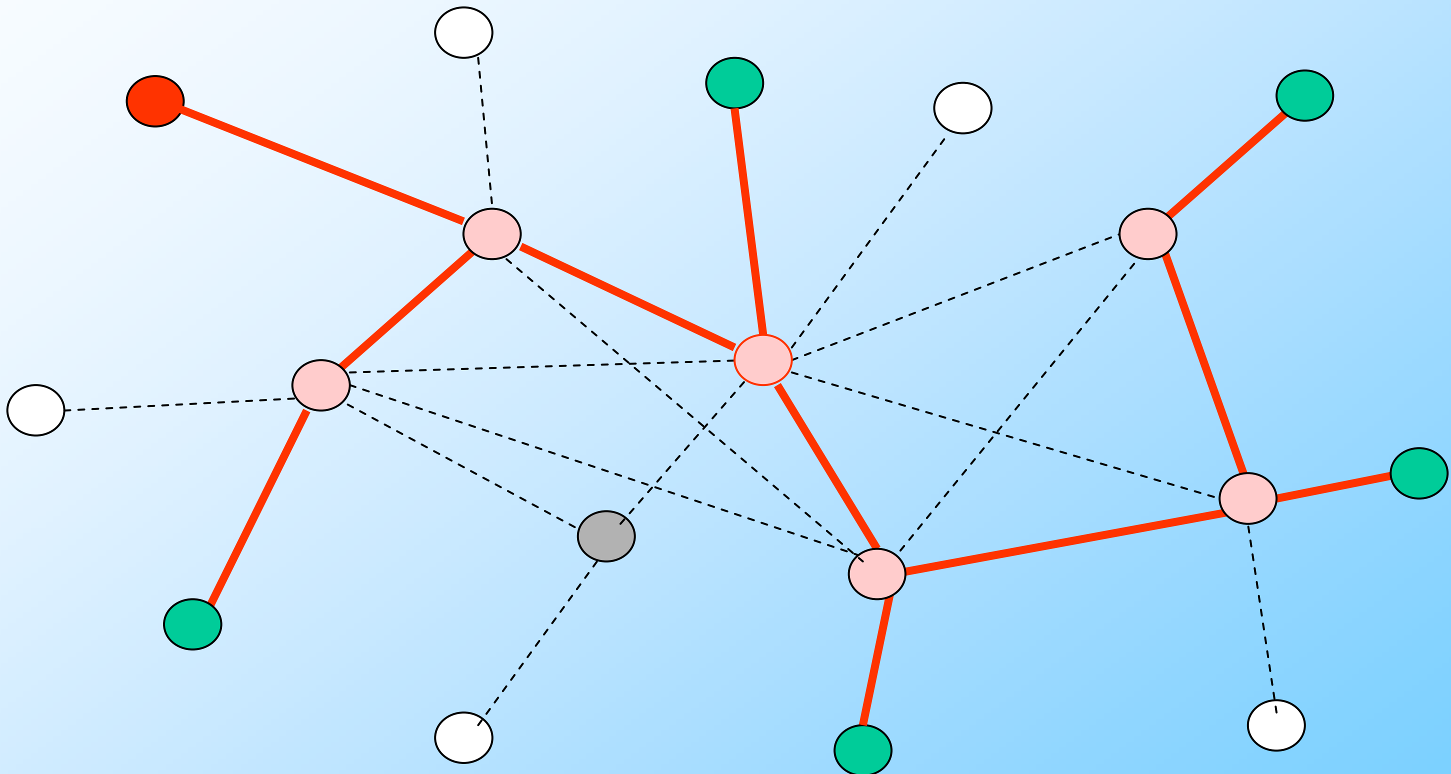
-Participants Sub-graph -



$G'(t) = (P(t),E') \subseteq G$

Hybrid Nodes Architecture

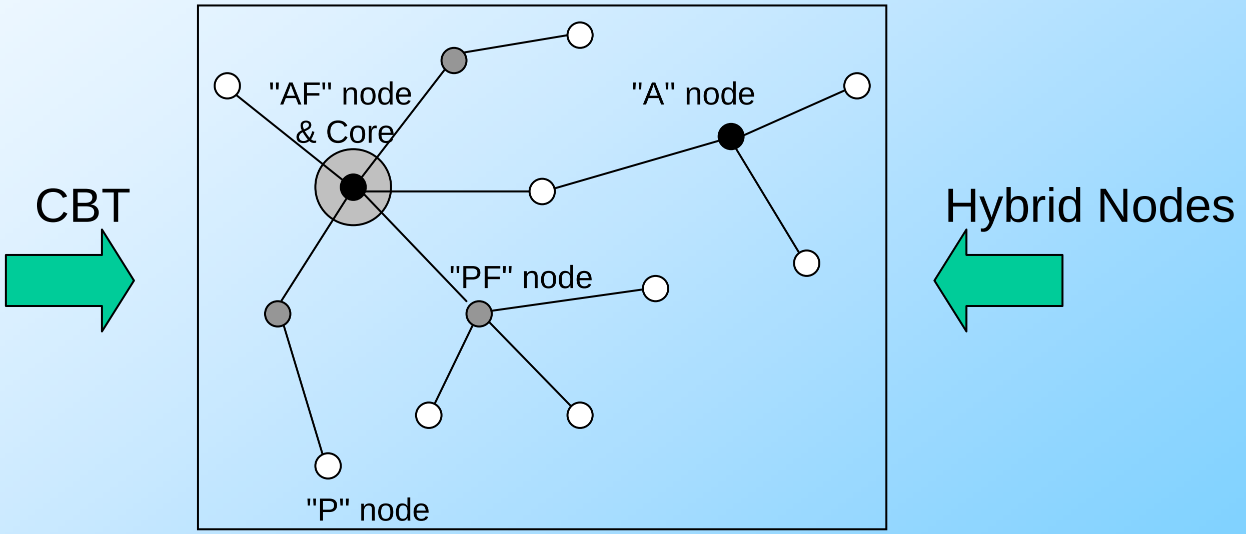
- Communication Sub-graph (CBT) -



Rule: *A node cannot join the multicast tree if the accumulated delay on the path to the core is greater than MaxValue*

Hybrid Nodes Architecture

- CBT, multicast tree -



Future

- Develop a monitoring system that will allow an objective assessment of the proposed architecture
- Refine the architecture and study its scalability.
  - If the number of remote participants increases we anticipate the need of some centralized control and additional dynamic shared state management techniques to maintain interactivity

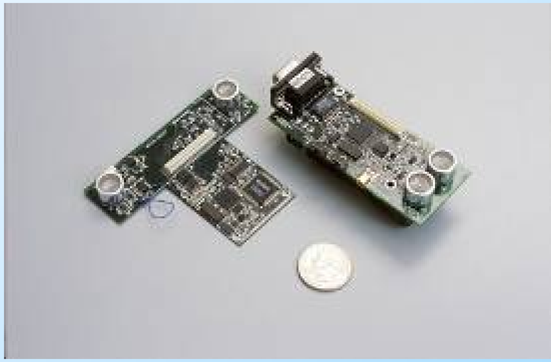
Crickets

Steve Teicher, DM Faculty Member

Summary

- Provides position sensing inside where GPS is blocked
  - Time stamps sent via RF from cricket beacons
  - Distance information comes from ultrasound signal synchronized to RF time stamps
  - Distance from beacon determined by time of flight of ultrasound
  - Position determined by triangulation of 3 or more beacon distances

Packaging

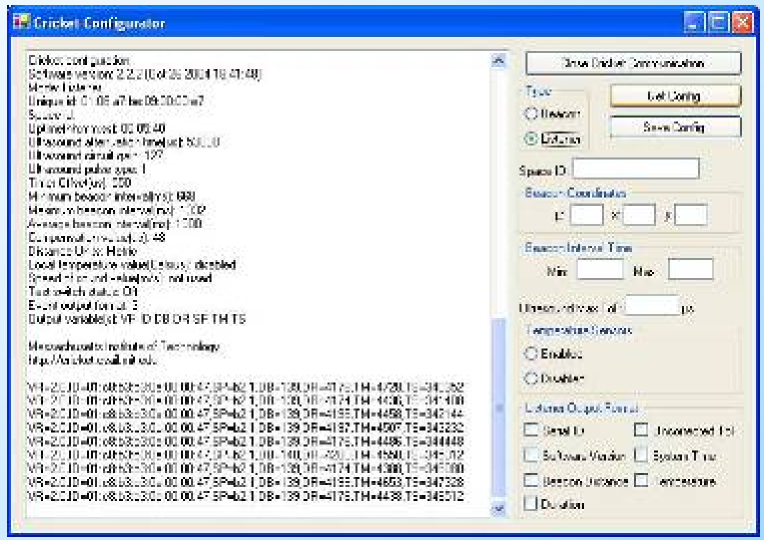


- Beacons and listeners are identical hardware packages
  - The difference is a command that tells the device to behave as a listener or beacon.
  - Current packaging is on ~ 2 by 5 inch pc card with serial i/o and battery or AC power shown on the right above
  - Later versions will have CF i/o and fit a compact flash slot as shown on the left.

Status

- Devices as shipped by Crossbow were not shipped with firmware. This was a surprise.
- Purchased a programmer from Crossbow and installed firmware from MIT website.
- Developed a utility to program beacons and to gather data from listeners.

Terminal Interface



Above is the terminal interface on the left and a utility that UCF’s Will Pereira built on the right. The utility determines the cricket’s function by choosing the appropriate radial button under Type. The listener output items are determined by check-boxes at the bottom. The Cricket Communication module can either talk to the interface shown or provide similar functions to any calling program

Future

- Build reusable classes that can be used in any program that needs location info
- Beacon class will have methods that return info such distance from beacons or xyz coordinates relative to building structures
- Acquire sufficient beacons to instrument a facility
- Experiment in determining object orientation using multiple listeners
- Experiment with techniques for understanding & improving location accuracy

PDA Connection

Various DM and CS Students

Partially Done and Future

- ARToolkit on PDA (working)
- Crickets talking to PDA
- Integrate with behavior work
- Display images on wall based on location (crickets and markers), orientation (markers, multiple crickets and perhaps motion vector)
- Choose content of images based on user behavior, or behaviors of others in building

Pervasive Audio

Darin Hughes, M&S PhD Candidate

Also Research Associate IST/MCL

Done in his presentation

Authoring / Delivering

Includes MR Backlot

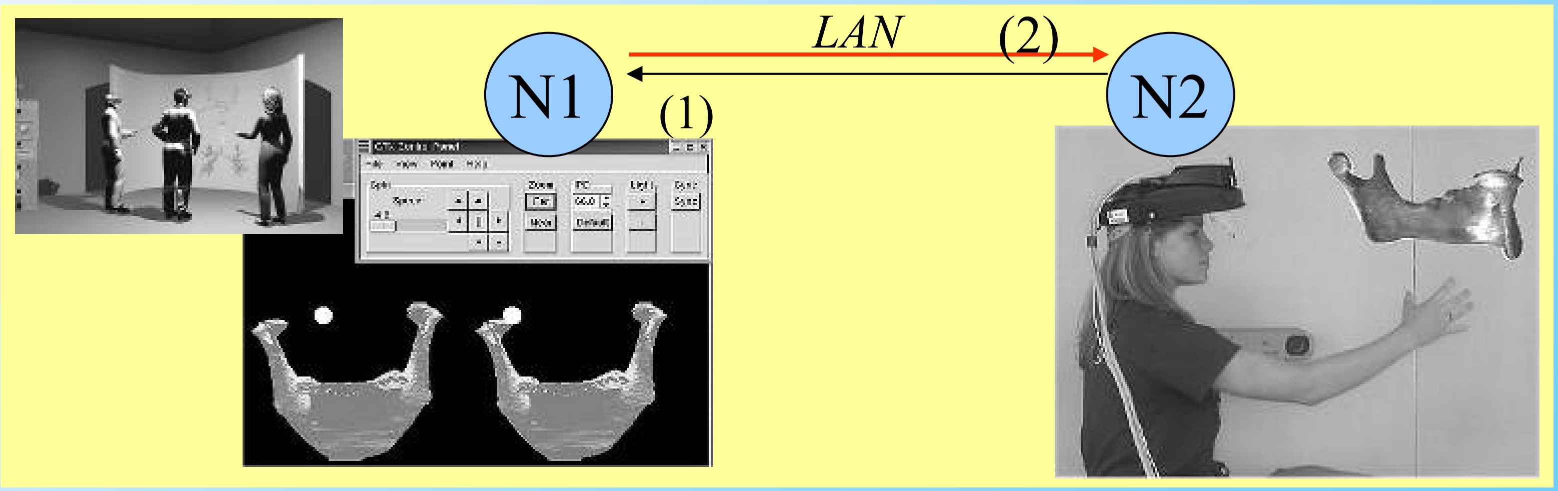
Adaptive Scene Synchronization

Felix Hamza-Lup

CS PhD in Summer 2004

Adaptive Scene Synchronization Algorithm

- Motivation
  - distributed algorithm for shared state maintenance that compensates for the ***network latency***
  - takes into account the network infrastructure behavior
  - provides distributed computation combined with distributed system monitoring



Adaptive Scene Synchronization Algorithm
- Overview

- DCE is seen as
  - A distributed system of "n" nodes
  - Each node:
    - runs a set of threads: rendering, interaction, monitoring.
    - has access to a local library of 3D models
    - data is exchanged through software objects (each shared virtual 3D object has a software object associated)

- Two types of nodes
  - "server" nodes (produce/broadcast interaction data, software objects)
  - "client" nodes (consume interaction data, compute delay)

- Each node adjusts the local scene attributes based on
  - delay (between each producer and consumer)
  - information carried in the software objects (e.g. interaction speed)

Adaptive Scene Synchronization Algorithm
- Drift Value, Drift Matrix

- *Drift value* at *(N2)* - is the product between the action velocity and the network delay

- For a DCE of "N" nodes sharing "M" virtual objects
  - Velocities matrix, $S = [s_i]$, where $i \in [1, M_\tau]$
  - Delays matrix, $T = [t_j]$, where $j \in [1, N_\tau]$

- *Drift matrix*
  - $\boldsymbol{D}(M_\tau, N_\tau) = \boldsymbol{ST}^t$

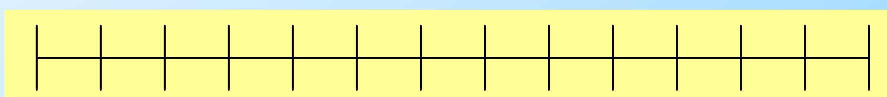
Adaptive Scene Synchronization Algorithm

<u>Client side</u>:

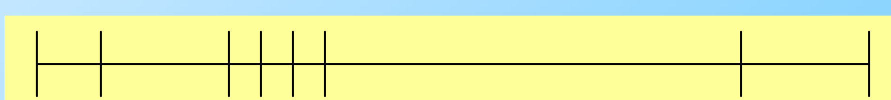
```
Client side:
        Initialization:
                Tn ← ComputeNodeDelay()
                Sn ← UpdateAction();
                Dn ← UpdateDrift()
                UpdateLocalScene();
        Main:
                if (trigger)
                        Tn ← ComputeNodeDelay()
                        Dn ← UpdateDrift()
                end if
                if (changedScene)
                        Sn← ReceiveChanges()
                        Dn ← UpdateDrift()
                end if
Server side:
        for ever listen
                if (newClientRequest)
                        SendToClient(Sn);
                end if
                if (changedScene)
                        BroadcastChanges();
                end if
        end for
```

Delay Measurements
- Fixed vs. Adaptive Threshold

- When do we trigger the delay computation ?
- Delay measurements must be triggered whenever significant variations in the network delay appear
  - Fixed Threshold – delay measurements are triggered at regular intervals

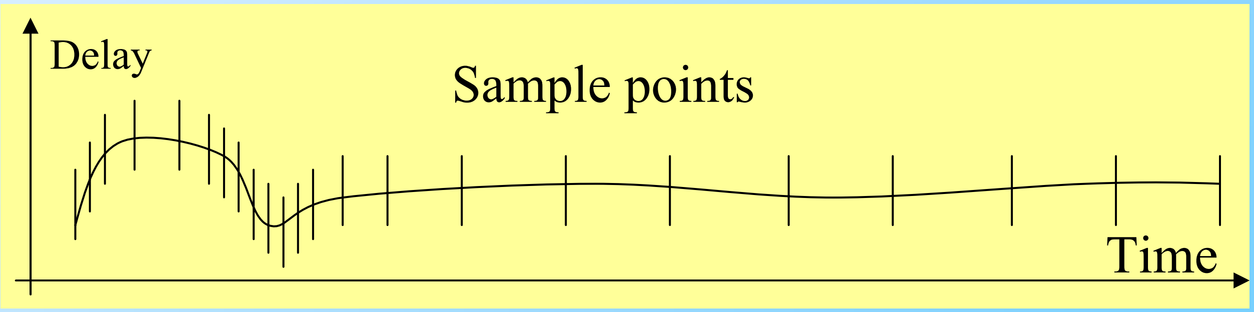


  - We propose an *Adaptive Threshold* - delay measurements are triggered based on the delay history - better characterizes the network jitter and the users interaction



Delay Measurements
- Adaptive Threshold

- Let:
  - H_p the delay history
  - σ and h_{mean} be the standard dev. and the mean of H_p
  - h_0 be the most recent delay, i.e. the last entry in H_p
  - γ_0 the current frequency of delay measurements, (*expressed as the number of measurements per second*)
- Adaptive approach:
  - decrease γ_0 , if $h_0 \in [\, h_{mean} - \sigma, h_{mean} + \sigma \,]$
  - increase γ_0 , if $h_0 \notin [\, h_{mean} - \sigma, h_{mean} + \sigma \,]$



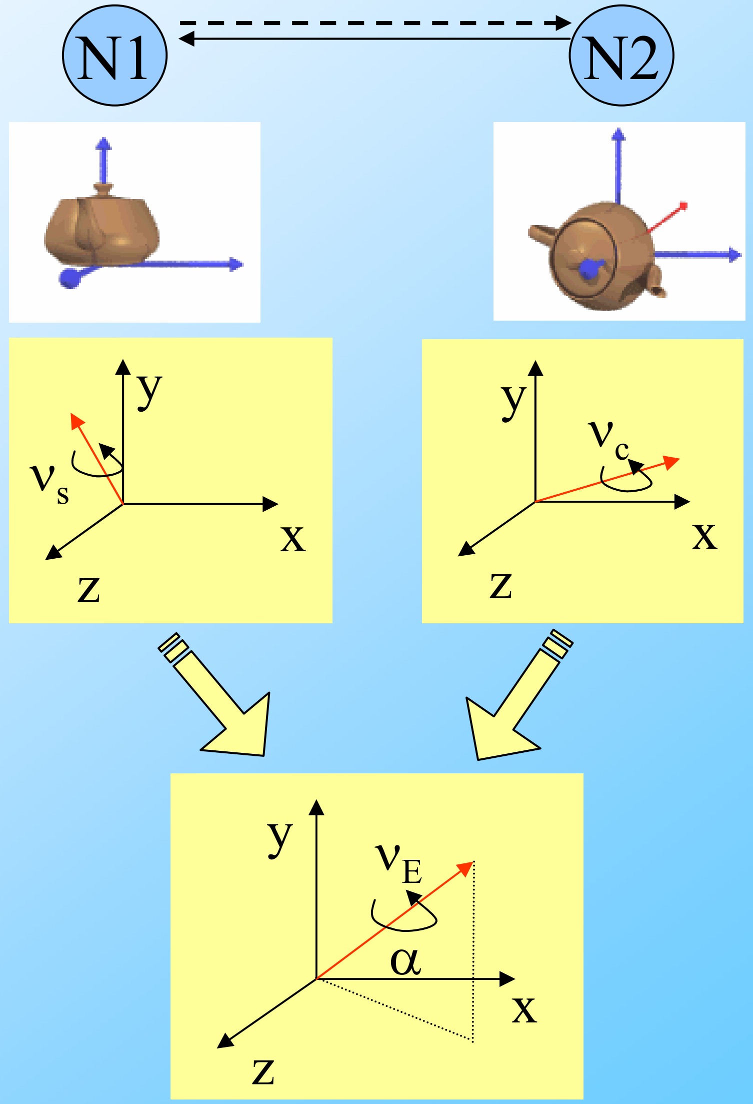
Quantitative Assessment

- Assess *orientation drift* of a shared 3D virtual object
- Let
  - q_s - rotation of an object at N1
  - q_c - rotation of the same object at N2
- Correction quaternion (q_E) - expresses the error between the actual orientation of the object and the desired orientation

$$q_s = q_E q_c$$

$$q_E = (\omega_E, v_E)$$

$$\alpha = 2\cos^{-1}(\omega_E)$$



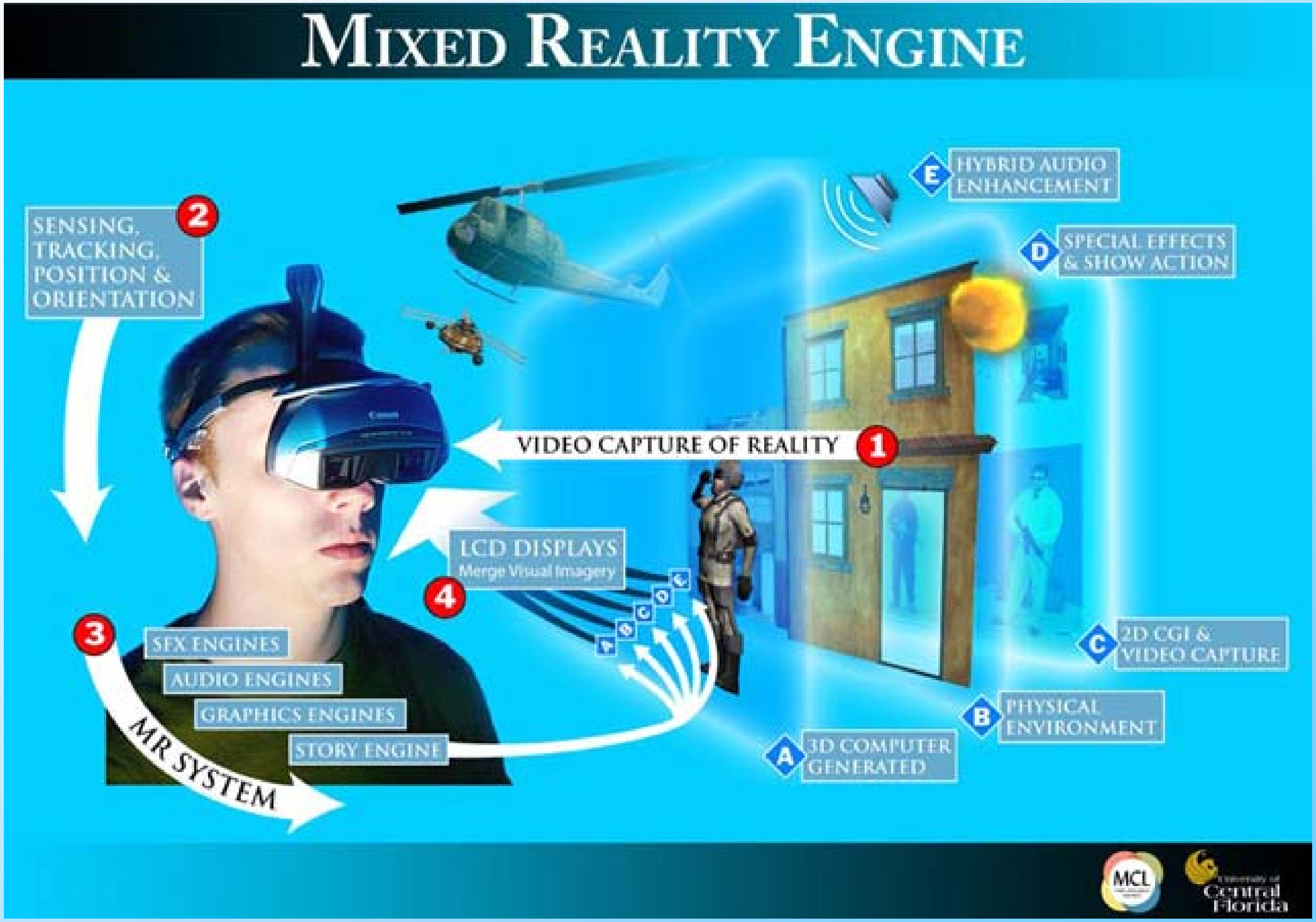
Future

- Extend the system infrastructure to multiple interacting nodes

MR Story Engine

Matt O’Connor, CS PhD Candidate
Also Research Associate IST/MCL

Integration of MR Engines



Story / Scripting / Master

- Agent architecture
- XML-based scripting
  - behaviors with guarded cases, triggers, guarded reflexes
- AI Plug-ins
- Multiple physics engines
- Pluggable-interface protocol
- Procedural scripting

Graphics Engine

- OSG
- Cal3D
- Agent peers
- Occlusion
  - Impostors
  - Matte support

Audio Engine

- Ambient, point, dynamic (3d moving)
- Constrained based on speaker placement
- Agent peers

Network Protocol

- Simple, efficient
- Command stream
- Control stream
  - Story can attach control stream to trackers, physics engine, other agents

Future

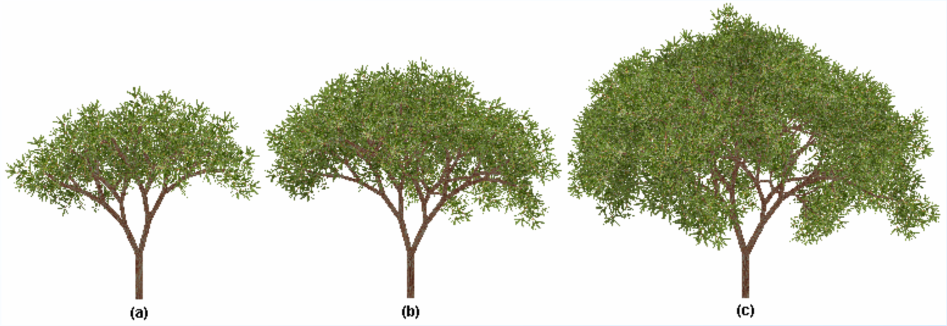
- Experiment with AI plug-ins
- Develop visual authoring interface
- Continue work on network protocols
- Integrate shadow/illumination work
- Run stories on DPAs
- Go outdoors

MR Audio Engine

Darin Hughes, M&S PhD Candidate

Also Research Associate, IST/MCL

In separate presentation



Virtual Landscapes

Julien Perret, IRISA PhD Candidate
Paulius Micikevicius, CS PostDoc
Valerie Sims, Psych Prof
Hana Smith, Psych PhD Candidate

Goals

- Model complex environments
- Use real data
  - Height maps
  - Water maps
- Model the evolution of the environment
- Simulate emergent phenomena
- Visualize the generated environments

Challenges

- Computationally intensive
  - "growing" the forest and cities
    - L-System for biological correctness and variations
    - plant interaction
    - fires, droughts, *etc.*
    - constraints (e.g., existing building over new branch, streets)
  - rendering the forest
    - each tree is unique and has complex geometry
    - a large number of trees can overwhelm current graphics hardware
  - rendering the buildings
    - constrained by human resources and street networks

L-systems generation of trees

ω $FA(1)$

p_1 $A(k) : \min\{1, (2k+1)/k^2\}$

$\rightarrow \quad /(\varphi)\ [+(\alpha)\ FA(k+1)]\ -(\beta)\ FA(k+1)$

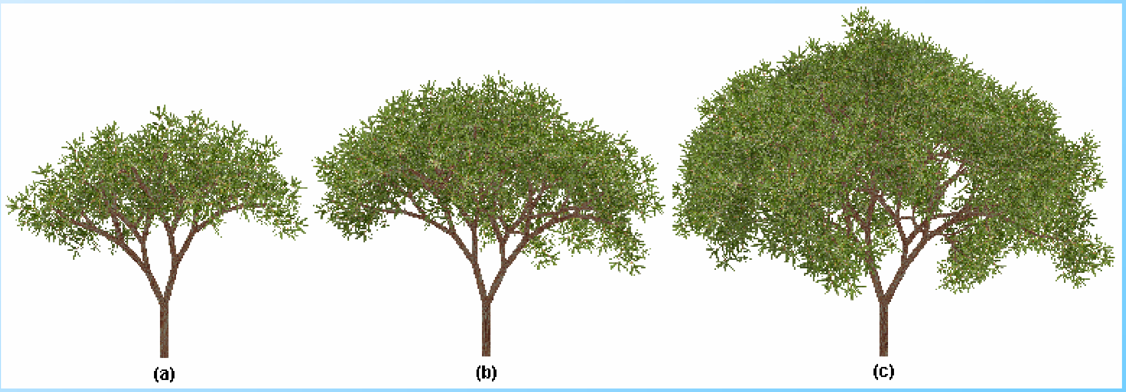
p_2 $A(k) : \max\{0, 1-(2k+1)/k^2\}$

$\rightarrow \quad /(\varphi)\ -(\beta)\ FA(k+1)$

Modules +, – denotes rotation around the z-axis (up)

Module / denotes rotation around the y-axis

The angles for the rotations are specified for a given class of trees ($\alpha = 32^\circ$, $\beta = 20^\circ$, $\varphi = 90^\circ$).



Perception Experiments

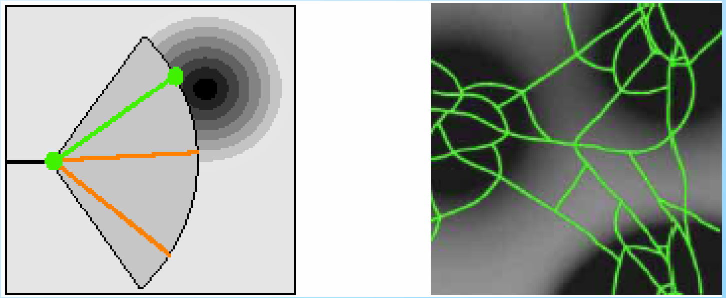
- Goal is to determine when people perceive a pop in display (common problem with discrete LODs)
- Even with visibility LOD, distance is useful metric as people perceive pops less readily at distances
- If subject is moving fast, pops are not as readily observed; detail may be able to be reduced, compensating for larger area traversed
- LODs in high mid range are less vulnerable to observed popping

Applications

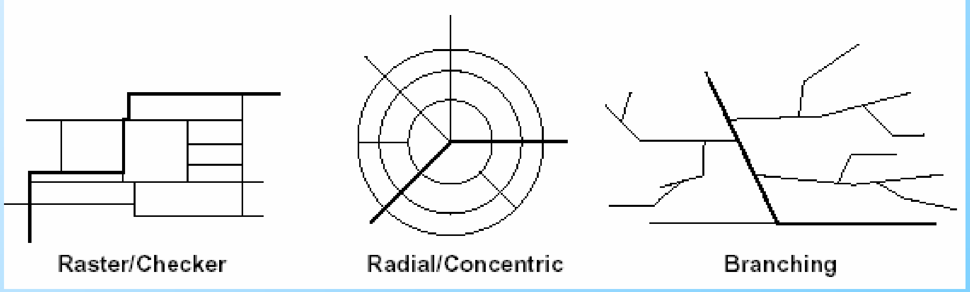
- Biological models
  - Cells
  - Plants
  - Trees
- More recently
  - Street networks
  - buildings

Street network using L-systems

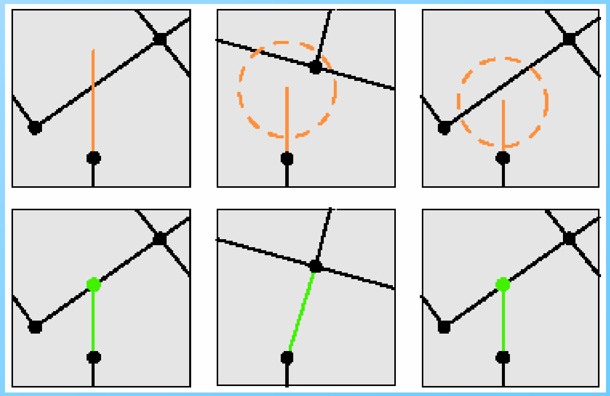
- Global constraints
  - Population density
    - For highways



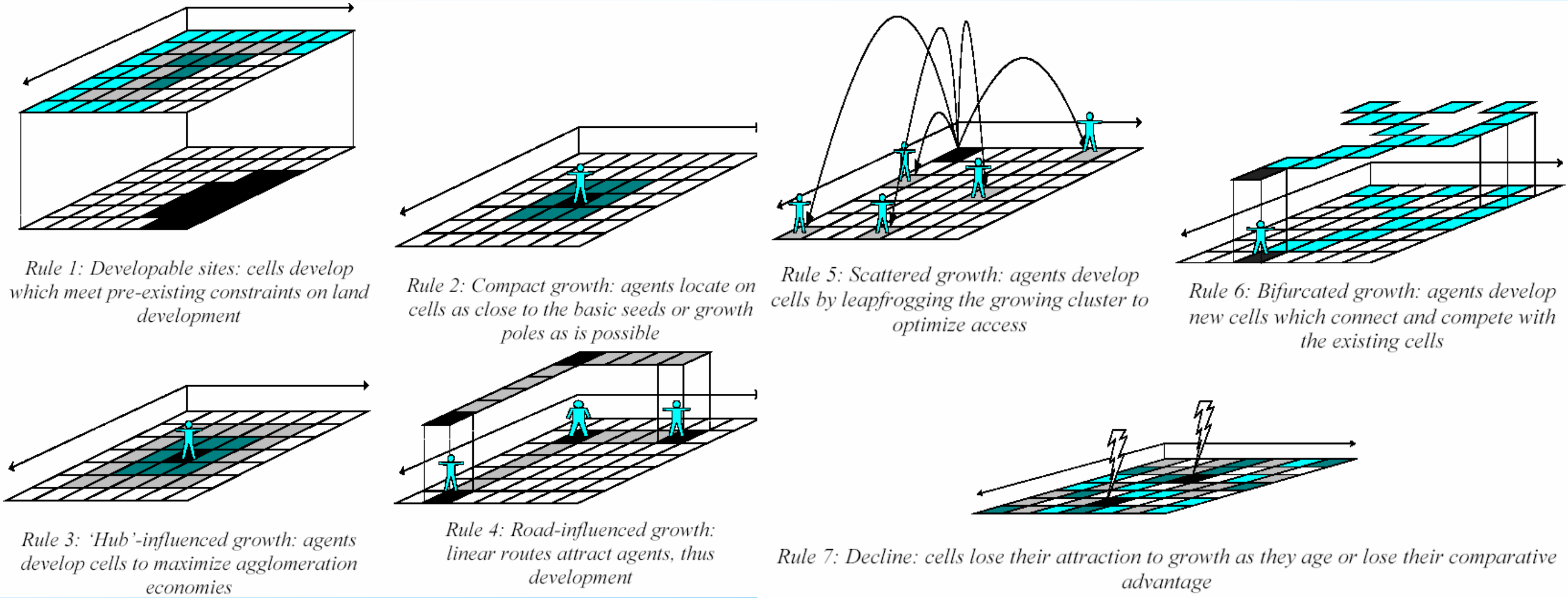
  - Road patterns
  - For streets



- Local constraints



Urban growth and suburban sprawl: the rules



Rule 1: Developable sites: cells develop which meet pre-existing constraints on land development

Rule 2: Compact growth: agents locate on cells as close to the basic seeds or growth poles as is possible

Rule 5: Scattered growth: agents develop cells by leapfrogging the growing cluster to optimize access

Rule 6: Bifurcated growth: agents develop new cells which connect and compete with the existing cells

Rule 3: 'Hub'-influenced growth: agents develop cells to maximize agglomeration economies

Rule 4: Road-influenced growth: linear routes attract agents, thus development

Rule 7: Decline: cells lose their attraction to growth as they age or lose their comparative advantage

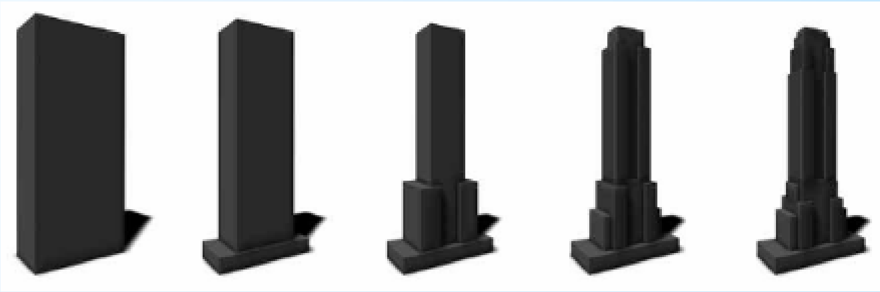
Urban growth and suburban sprawl: the results

Simulation up to time 1850
t=50

Simulation up to time 1950
t=150

Simulation up to time 2000
t=200

Buildings using L-systems



Future

- Large scenes of biologically accurate vegetation
  - forests, fields, *etc.*
  - thousands to millions of unique trees
- Buildings, streets, urban landscape
- Realistic rendering; Interactive frame rates
- Interaction
  - Interaction of buildings, streets and trees
- Applications
  - rapid creation of virtual assets
  - informal science education
  - urban planning
  - search and rescue training

3d Tour from 2d Photos

James Burnett, CS PhD Student
at IRISA, Rennes, France

High-Level

- Create tours from memories and friends with only tools being a digital cameras and laptop
- Produce with little manual intervention
- Pass on reconnaissance / planning information in a form that includes relation of images to each other

Current Goals (2 months)

- Specify vanishing points manually – existing automated tour method do this poorly
- Link important spaces in related camera views via user specified markers
- Plan is to alpha blend the scenes based on their distance from the user
- How should the geometry be glued together
  - At any given point in space, there exists a list of tours being displayed, and a selected current tour which is dominant
  - World space is taken to be the camera space of the current tour, so it is the only tour that is not warped in any way
  - All other tours are transformed so as to glue their related point together

Future

- Integrate with mobile devices:
  - PDAs, phones
- Many things we do not yet know about

PhD Students

- Dr. Felix Hamza-Lup finished July 2004
  - Adaptive scene synchronization
- DanZhou Liu working with Kien Hua
  - Deducing intent from primitive movement
- James Burnett
  - Color mapping for indoor/outdoor
  - 3d photorealistic tours from 2d image sequences (IRISA)
- Matt O'Connor
  - MR Story Engine, network protocols, evolution of reg. expressions
- Nick Beato
  - MR Graphics Engine, network protocols
- Darin Hughes
  - MR Audio Engine (with S. Vogelpohl), expectation, 3d audio
- Jason Gauci and Sean Szumlanski
  - AI behaviors (dynamic path planning / genetic algorithms)
- Gordon Worley
  - Visual programming for MR Story Engine
- Julien Perret (from IRISA in Rennes, France)
  - Rapid creation of virtual assets via L-systems (forests, villages, cities, etc.)

Papers -- Experiences

- Christopher B. Stapleton and Charles E. Hughes, “Mixed Reality and Experiential Movie Trailers: Combining Emotions and Immersion to Innovate Entertainment Marketing,” *Proc. of 2005 International Conference on Human-Computer Interface Advances in Modeling and Simulation (SIMCHI’05)*, New Orleans, January 23-27, 2005.
- Felix G. Hamza-Lup, Charles E. Hughes and Jannick P. Rolland, “A Distributed Augmented Reality System for Medical Training and Simulation,” *Energy, Simulation-Training, Ocean Engineering and Instrumentation: Research Papers of the Link Foundation Fellows, Vol. 4*, Rochester Press, 2004.
- Scott Malo, Christopher B. Stapleton and Charles E. Hughes, “Going beyond Reality: Creating Extreme Multi-Modal Mixed Reality for Training Simulation,” *Proceedings of I/ITSEC 2004*, Orlando, December 6-9, 2004.
- Frank Dean, Pat Garrity and Christopher B. Stapleton, “Mixed Reality: A Tool for Integrating Live, Virtual & Constructive Domains to Support Training Transformation,” *Proceedings of I/ITSEC 2004*, Orlando, December 6-9, 2004.
- Charles E. Hughes, Eileen Smith, Christopher B. Stapleton and Darin E. Hughes, “Augmenting Museum Experiences with Mixed Reality,” *Proceedings of KSCE 2004*, St. Thomas, V.I., November 22-24, 2004.
- Christopher B. Stapleton and Charles E. Hughes, “Interactive Imagination: Tapping the Emotions through Interactive Story for Compelling Simulations,” *IEEE Computer Graphics and Applications* 24(5), pp. 11-15, September/October 2003.

Papers -- Sensors

- DanZhou Liu and Charles E. Hughes, "Deducing Behaviors from Primitive Movement Attributes," *SPIE Defense and Security Symposium*, Orlando, FL, March 28 - April 1, 2005.
- Felix G. Hamza-Lup, Charles E. Hughes and Jannick P. Rolland, "Hybrid Nodes with Sensors – Architecture for Interactive Distributed Mixed and Virtual Reality Environments," *SCI 2004*, Orlando, FL, July 18-21, 2004.

Papers -- Authoring/Delivery

- Darin E. Hughes, Scott Vogelpohl and Charles E. Hughes, "Designing a System for Effective Use of Immersive Audio in Mixed Reality," *Proc. of 2005 Int'l Conference on Human-Computer Interface Advances in Modeling and Simulation (SIMCHI'05)*, New Orleans, January 23-27, 2005.
- Matthew O'Connor and Charles E. Hughes, "Authoring and Delivering Mixed Reality Experiences," *Proceedings of 2005 Int'l Conference on Human-Computer Interface Advances in Modeling and Simulation (SIMCHI'05)*, New Orleans, January 23-27, 2005.
- Felix G. Hamza-Lup, Jannick P. Rolland and Charles E. Hughes, "A Distributed Augmented Reality System for Medical Training and Simulation," *Energy,Simulation-Training, Ocean Engineering and Instrumentation: Research Papers of the Link Foundation Fellows, Vol. 4, Rochester Press, 2004*
- Felix G. Hamza-Lup, Charles E. Hughes and Jannick P. Rolland, "Distributed Consistency Maintenance Scheme for Interactive Mixed Reality Environments," *CITSA 2004*, Orlando, FL, July 21-26, 2004.
- Felix G. Hamza-Lup and Jannick P. Rolland, "Adaptive Scene Synchronization for Virtual and Mixed Reality Environments," *Virtual Reality IEEE VR 2004*
- Charles E. Hughes, Christopher B. Stapleton, Paulius Micikevicius, Darin E. Hughes, Scott Malo and Matthew O'Connor, "Mixed Fantasy: An Integrated System for Delivering MR Experiences," *VR Usability Workshop: Designing and Evaluating VR Systems*, Nottingham, England, January 22-23, 2004.

Papers -- Evaluation

- Darin E. Hughes, Jennifer Thropp, John Holmquist, J. Michael Moshell, “Spatial Perception and Expectation: Factors in Acoustical Awareness for MOUT Training,” *Proceedings of Army Science Conference (ASC) 2004*, Orlando, FL, November 29-December 2, 2004.
- Charles E. Hughes, Glenn Harrison, Steve Fiore, Elisabet Rutstrom, Eileen Smith and Christopher B. Stapleton, “Cognition in Natural Environments: Using Simulated Scenarios in Complex Decision Making,” *Proceedings of Army Science Conference (ASC) 2004*, Orlando, FL, November 29-December 2, 2004. (Included in Proceedings but not Presented)

Papers -- Graphics

- Charles E. Hughes, Jaakko Konttinen and Sumanta N. Pattanaik, “The Future of Mixed Reality: Issues in Illumination and Shadows,” *Proceedings of I/ITSEC 2004*, Orlando, December 6-9, 2004.
- Erik Reinhard, Ahmet O. Akyuz, Mark Colbert, Matthew O’Connor and Charles E. Hughes, “Real-Time Color Blending of Rendered and Captured Video,” *Proceedings of I/ITSEC 2004*, Orlando, December 6-9, 2004.
- Charles E. Hughes, Erik Reinhard, Jaakko Konttinen and Sumanta N. Pattanaik, “Achieving Interactive-Time Realistic Illumination in Mixed reality,” *Proceedings of Army Science Conference (ASC) 2004*, Orlando, FL, November 29-December 2, 2004. (Poster Presentation Only)

Tutorials/Invited Presentations

- Frank Dean and Christopher B. Stapleton, “Augmented Reality,” *I/ITSEC 2004*, Orlando, FL, December 7, 2004.
- Christopher B. Stapleton, Charles E. Hughes and Scott Malo, “Extreme MR: Going beyond Reality to Create Extreme Multi-Modal Mixed Reality for Entertainment, Training and Education,” *ISMAR 2004*, Washington, D.C., November 2, 2004.
- Charles E. Hughes, “Mixed Reality: Trompe l’oëil in the 21st Century,” University of Rennes, Rennes, France June 2004.
- Christopher B. Stapleton, Charles E. Hughes and Erik Reinhard, “Trompe l’oëil” Past, Present and Future,” Appleton Museum of Art, Ocala, FL, May 2004.
- Charles E. Hughes, “Merging the Real and Virtual: The Science Behind the Curtain,” *2003 I/ITSEC Conference*, Orlando, FL, December 2003.