BOWIE STATE UNIVERSITY SPRING 2015 COSC 729 : VIRTUAL REALITY AND ITS APPLICATIONS

PROJECT REPORT: GAMING : ROBOT CAPTURE

PROFESSOR: Dr. SHARAD SHARMA STUDENTS:

Issiaka Kamagate Jamil Ramsey



OUTLINE

INTRODUCTION

- 1. GOAL AND OBJECTIVES
- 2. MODELING
- 3. PROBLEMS ENCOUNTERED
- 4. FUTURE IMPROVEMENTS
- 5. RESUME



INTRODUCTION

Video games have become an incredibly popular and pervasive form of entertainment. Video game use has increased steadily overtime (Rideout, Foehr, & Roberts, 2010) and today 9 out of 10 American children and teens play video games (Gentile, 2009; Gentile & Walsh, 2002). On average, youth play video games for two hours a day (Rideout, Foehr, & Roberts, 2010). However, a significant percentage of males report playing four hours a day or more (e.g., Bailey, West, & Anderson, 2010). This fact that such a large number of children and adolescents frequently play video games makes video game very important today.

1. GOAL AND OBJECTIVES

The objective of this game is to capture 5 escaped robots on a college campus. The game will simulate a crowd's reaction to danger while we try and capture the robots.

- 2. MODELING
- <u>Environment</u>: The game is taking place in a virtual college campus. The virtual campus is a unique blend of old and new. Boasting numerous cultivar of trees, shrubs, and flowers, the campus is dotted with structures of the early 1900s, as well as buildings exemplifying the most contemporary modes of architecture. There is also a soccer field.



Picture 1: screenshot of virtual campus



Picture 2: screenshot of virtual campus



- Light: We used a global illumination.
- <u>Sky</u>: We used a dynamic sky (sky_day.osgb) including in vizard.
- <u>Timers</u>: The game also integrated a timer of 60 second to find the robot.
- <u>Audio:</u> There are several sound in the game. First, there is a sound of a bell which allows the player to locate the robot. Next, there is a sound for catching or not the robot and a sound to alert the player when he has less than five second.



- Programming tools:

Design: We used SketchUp to design the environment.

Programming: Vizard software for building, rendering, and deploying the applications. Vizard uses Python for scripting.

8		Final_Project.py	- Vizard 5 Trial (64-bit)	- 8 ×
File Edit View Script Debug	Window Tools Help			vizand
D = # × 0 1 p + D				🔎 Search Open Documents 🔹 🛱 -
Resources 4 ×	A new release of Viza	d is available for download Downloa	lgnore	×
4 Audio	Getting Started Vizard2.py kama	_ links example.py links example.py	Final_Project.py • Final_Project_Virtual_Reality.py	Assignment4_VR2Client.py viz.py += →
BOING!.WAV	1 import viz			0
👌 bells.wav	2 import vizact			
👌 bounce.wav	3 import vizcam			
werewolf.wav	4 import vizinfo			
and the second sec	5 import viztask			
4 Avatar	6 import vizproximity			
1 boy1.cfg	7 import vizfx.postproc	ess	-	
🛉 man2.cfg	<pre>8 from vizfx.postproces</pre>	s.color import GrayscaleEf	fect	
🛉 vcc_female.cfg	9 Irom Vizix.postproces	s.composite import Blendki	IECT	
4 Model	11 import math			
🕥 campus.dae	12 import random			
🔾 robot.ive	13			
🕥 sky_day.osgb	15 TRIAL COUNT = 5	f Number of trials	ner dine	
	16 TRIAL DURATION = 60	# Amount of time al	loved for finding each robot (in secon	ds)
4 Texture	17 TRIAL DELAY = 4	# Delay time betwee	n trials	
📰 robotsa2.png	18 PROXIMITY RADIUS = 7.	0 # Radius for proxim	ity sensor around flag	
	19 FLASH_TIME = 3.0 20	# Time to flash scr	een at beginning of each trial	
	21 # List of hiding spot	s for flag		
	22 HIDING_SPOTS = [
	23 [13, 0.2, 13]			
	24 ,[0, 0.2, 25]			
	25 ,[20, 0.2, -10]			
	20 , [-9, 0.2, -0]			
	28 [11 0 2 13]			
	29 .[30, 0.2, -10]			
	30 , [-12.5, 0.2, 10]			
	31 , [6.8, 0.2, -1.4]			
🍕 Code Browser 🧕 Resources	<			>
Ready				🛒 Ln 16 Col 77 INS
🛋 🥝 🚞 🗅		🗧 🧕 🦌	💿 🏘 🖨 😰 🛃	60 A A A A A A A A A A A A A A A A A A A

Picture 3: vizard screenshot



 <u>Avatars</u>: We included some inbuilt avatars and agents from vizard with their own animations. We used 3DS Max and bip motion capture files to create and run animations for inbuilt avatars.

Avatars have multiple animations like talking, seating, walking and dancing.

8 custom avatars were created using 3D Studio Max. Each avatar has 4 animations. A stand, walk, sit, and run animation.

Custom Avatar creation procedure:

1) We started with a fully textured and rigged model in MAX, The cal3d export plugin should already be installed. For the cal3d export to work correctly you MUST export the Skeleton first (followed by the mesh(s), material(s), and animation(s) in any order you choose)



2) Select the bone rig





3) Select the motion menu and click the load file button.



4) Select "In place mode"

	🍬 📶 📠 💿 🛄 🥕
	CMan0012-Bip
FRONT	Selection Level:
	Sub-Object Trajectories V
	Parameters Trajectories
	+ Assign Controller
	- Biped Apps
	Mixer Workbench
	- Biped
	2 00 7 4
	Pg 📂 🗏 🤔 8
	- Modes and Display
	Modes
	e w = o,
	Display
	2,88, 🚥 🖙 🔨 📾
	Name: CMan0012-Bip
	+ Track Selection
·	+ Quaternion / Euler
	+ Twist Poses
	+ Bend Links
	+ Key Info
	+ Keyframing Tools
	+ Copy/Paste
	+ Layers
	Motion Capture
	Dynamics & Adaptation



5) Select Export > Cal3d Skeleton. All bones should be selected. Click finish.

Step 1 of 1 The exporter has found the nodes below that match the criteria for Cal3D bones. Select the ones you want to export. Select the nodes you want to export: Select the nodes you want to export: Select the nodes you want to export: CMan0012-Bpine CMan0012-Spine 1 CMan0012-Spine 1 CMan0012-Spine 1 CMan0012-Spine 1 CMan0012-Spine 2 CMan0012-Spine 1 CMan0012-Spine 2 CMan0012-Spine 1 CMan0012-Spine 1 CMan0012-Spine 1 CMan0012-Spine 2 CMan0012-Spine 1 CMan0012-Spine 2 CMan0012-Spine 2 CMan0012-Spine 1 CMan0012-Spine 2 CMan0012-Spine 3 CMan0012-Spine 4 CMan0012-Spine 4 <	•	Cal3D Export - Skeleton Hierarchy	×
About Cal3D	C/ 30	Step 1 of 1 The exporter has found the nodes below that match the criteria for Cal3D bones. Select the ones want to export. Keep in mind that only the selected ones can be used later in animation and mesh Select the nodes you want to export:	you exports.
	About Cal3D	CMan0012-Bip Bip01 Footsteps CMan0012-Spine CMan0012-Spine1 CMan0012-Spine2 CMan0012-Spine3 CMan0012-Head CMan0012-Head CMan0012-Lead CMan0012-Lead CMan0012-Loavide CMan0012-Loavide CMan0012-L Finger41 CMan0012-L Finger41 CMan0012-L Finger42 CMan0012-L Finger31 CMan0012-L Finger31	<

6) Select the mesh in the object explorer and then Export > **Cal3d Mesh**. Select the defaults and finish.

2	Cal3D Export - Level of Detail ×
	Step 3 of 4
C/30	Cal3D supports progressive meshes for level-of-detail. The exporter can automatically create them now. This will take a while for complex models.
	1 - Set the LOD properties
	Automatically create progressive meshes
About Cal3D	
	< <u>B</u> ack <u>N</u> ext > Cancel



7) Select the material editor. Right click the textured node and rename. Material must follow a specific format. They must be ordered starting with 0. My model only had a single full body texture so I renamed it body[0]. If you textures are broken up you may have to do leg[0], arms[1], body[2]....

After your textures are named properly select the mesh in the object browser the **Export > Cal3d Material** and select the material to export. The texture file will be listed click next. At this point you need to copy the listed texture file to the same folder as the exported cal3d files.

2	Cal3D Export - Material Selection
	Step 1 of 2
C/30	The exporter has found the materials below that match the criteria for Cal3D materials. Select the one you want to export.
	Select the material you want to export:
	02 - Default 03 - Default 04 - Default 05 - Default 06 - Default 07 - Default 09 - Default 09 - Default 10 - Default 11 - Default 11 - Default 12 - Default 13 - Default 14 - Default 15 - Default 16 - Default 17 - Default 17 - Default 19 - Default 19 - Default 19 - Default 19 - Default 20 - Default 21 - Default 23 - Default 24 - Default 23 - Default 23 - Default 23 - Default
About Cal3D	body[0]
	< Back Next > Cancel

2		Cal3D Export - I	Material Maps		×
	Step 2 of 2	material contains the mans helow	M		
	Edit the maps	of this material:			
	Subfolder:			Texture format:	
	Map	Filename			_
	Dimuse	CMan0012.tf			
					_
					_
					_
About Cal3D					
			< <u>B</u> ack	Finish	Cancel



8) Next select Export > Cal3d Animation. By default it'll select the full animation range. If you want to export a partial animation enter the start frame and end frame.



2	Cal3D Export - Animation Time
About Cal3D	Step 3 of 3 The exporter has tried to query the animation time values. Adjust them carefully. Make sure that the start and end frame are correctly set for animation cycles, and use the displacement to synchronize different ones. 1 - Set the start and end frame of the animation Start End Frame: 2 - Set the displacement of the keyframes within the animation Displacement: Frames 3 - Set the frame rate (fps) of the animation Frame Rate: 30
	< <u>B</u> ack Finish Cancel

9) Create config file using the format below.

#

Cal3d cfg File

#

path=cwom0012/



scale=0.025

skeleton=cwom0012.csf

mesh=cwom0012.cmf

animation=stand.caf

animation=sit.caf

animation=walk.caf

animation=run.caf

material=cwom0012.crf

3. PROBLEMS ENCOUNTERED

The campus computers did not have the exporter for Cal3d. The instructions for exporting avatars were not clear and took a lot of trail and error to complete.

4. FUTURE IMPROVEMENTS

- Adding client server functionality
- Incorporating hardware support (Oculus and Data Glove).

5. RESUME

Catch The Robot is a capture the flag style game. The objective is to capture 5 missing robots. A proximity sensor is attached to the robot and when you get close the robot is considered captured. The game uses the WASD control scheme. You have a 60 second timer to capture each robot. The robots make a sound that gives you a clue to where it's hidden. Also the player can simulate an evacuation (by pressing 'm' button) to isolate the robot and make easier its capture. It triggers the run animation for the avatar. Simulating crowd behavior when confronted with a dangerous situation.

