A Team Report

Obstacle Avoidance

Final Project Report
Presented to
The Department of Engineering & Computer Science
Concordia University

In Partial Fulfillment
of the Requirements of
COMP 471
Section F, FK

by

Vengadesa Sri Venkadanathan	**ID: 3950425**
Vincent Garreau	**ID: 6129706**
Thibault Lecat	**ID: 6130038**

December 11, 2006

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 Abstract

By Vengadesa Sri Venkadanathan

Obstacle avoidance is one of the key components in robotics. For a detecting mechanism, anything that is on its path will be considered as an obstacle. In order to avoid obstacles, counter measures such as finding a detour abiding the provided constraints, dodging, drifting, moving etc., are adapted. This type of collision avoidance technique will be considered obstacle avoidance. This feature could be interpreted and implemented in several ways according to the requirements and needs having different degrees of abstraction and complexity. This obstacle avoidance concept could be found in the vast range of applications from critical real-time life saving applications such as bio-medics, security, etc., to purely entertainment based realms such as amusement, computer games and so on and so forth.

We could analyze the concept of objects avoidance remarkably in three main categories. Firstly, there will be a detecting mechanism, the detector, in the motion which has to determine its path depending on the non-moving static objects that would be on the way during its travel or movement until it reaches the destination as shown in Figure 1.

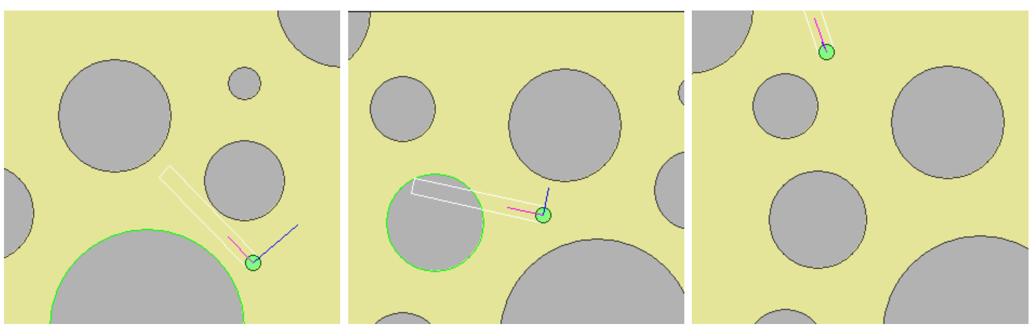

Figure 1: Obstacle Avoidance Steering Behavior [1]

This could also be called as steering behavior. In the second case, there will be a detector stationed in a certain location having a limited region for its movement which will be specified by the system whereas it will be making the movement within the certain range in order to avoid the randomly or deterministically moving objects. In the third category, both the obstacles and the detector would be in motion. Furthermore, we could also come up with other combinations for obstacles and detector according to the requirements or interest.

In the scope of our project, we have adapted real-time motion detection and a controlling system coupled with the human interaction that will be actuating and performing the obstacle avoidance. Not like any other complex robotic systems that are fully automated with the artificial intelligent capabilities, the system that we are building will be requiring

human interaction. Decision making will be solely made by the actor/player of the system as how and when to make a movement in order to avoid an obstacle. The collision detection and its consequent processing would be done by the computer system. Relative reaction and the current position rectification will be performed by the actor/player by the observation with the use of naked eyes upon the oncoming objects. This is a visually guided obstacle avoidance system. Depending on the actor/player’s movement, the system will change its state continuously.

2 Introduction

By Vengadesa Sri Venkadanathan

Our project has adapted the visually guided obstacle avoidance concept. Literally, this is an interactive game where the actor/player has to escape from the oncoming objects. There will be a hexagram plasma screen that will be displaying the road with its obstacles that will be output by a computer system as shown in Figure 2.

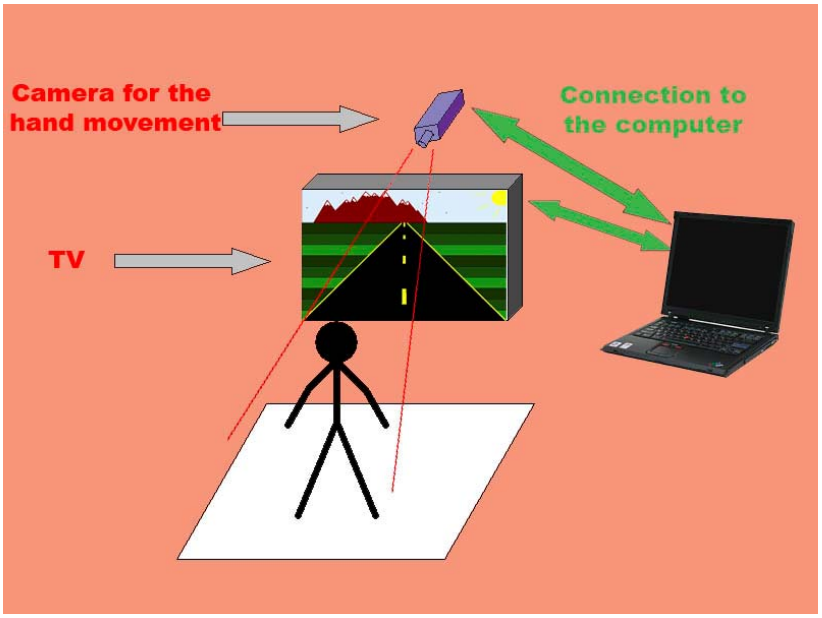

Figure 2: Actor/Player and the system setup

The actor/player is expected to stand 3 meters away from the TV and the camera is placed on the television. At the beginning of the game, in order to start playing, the player/actor has rise his right hand and reach the `start` button on the screen and tap on it as shown in Figure 3.

Figure 3: Starting the game

Once the game is started, the warm up messages such that the `Ready` screen is displayed to alert the player and then the `Go` message is given as shown in Figure 4.

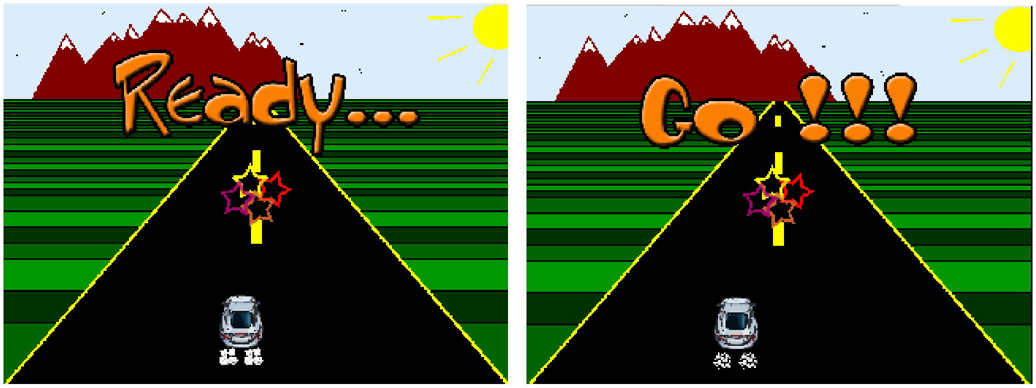

Figure 4: Welcome Screen

All the way from beginning to the end the proper audio streaming is played. Each game lasts 1minute and 6 seconds and each warm-up last 4 seconds. During the `Ready State` there will be three short beeps and one long final beep for the `Go State`. Afterwards, a portion of the Mario Song for the duration of 1minutes and 2 seconds will be played.

While playing the game, actor/player waves the left or the right hand; for the best performance, the hands have to be staying within the shoulder range in order to avoid the leaving out of the region of camera. Moreover, the background of the player has to be a white screen or rigid surface with a light unique color to avoid any noise interference. If the actor/player sees the obstacles coming on the left lane then the right hand has to be waved until the objects are avoided as shown in Figure 5.

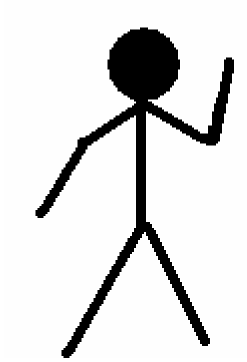

Figure 5: Moving to the right lane - right hand waving

On the contrary, if the objects are coming on the right lane then the left hand has to be waved as shown in Figure 4 in order to avoid the collision. Once the car reached the yellow solid line, for instance left side, any left hand movement will not do anything further on the car in regards to its left drifting.

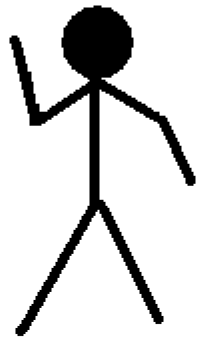

Figure 6: Moving to the left lane - left hand waving

For the player/actor, the display will look like as shown in Figure 7 in the television.

Figure 7: Output display – The car is sneaking through the obstacles

In the case of the collision, there will be stars flying in the display in order to denote such a collision state and after that the motion will continue as it was before as shown in Figure 8. A collision could occur by hitting an object by the back too. During a collision, a special audio stream is played to denote such state.

Figure 8: A Collision Instance

The obstacles will be randomly generated in the road and will be coming towards the player as shown above in Figure 8.

In this system, the player/actor is represented by a car which will be drifting left and right according to the hands' waving. The objective of this game would be that the player has to save the car from the oncoming objects (the balls) by moving it right or left using the right and left hands waving in an anticipated manner.

Once the game started, the objects will start flowing which will give the impression that the player is moving forwards on the road. The movement of the road is unidirectional. The actor/player could quit the game at any time and the game will end and come to the start state as it was in Figure 3.

3 Technical Analysis

By Vengadesa Sri Venkadanathan

Using Max/Jitter, with its graphical capabilities, the possibility of developing varieties of innovative software without any coding using any high level programming languages such as C++ and Java is in abundance. In our project, we have used the following software as shown in Table 1 for different image and video processing purposes.

	Name of the Software	**Functionality**
1	Painter 7	To create individual images and 3D effects
2	Microsoft Paint	`` `` ``
3	Adobe Photo CS2	`` `` ``
4	Jac Paint Shop Pro 7	`` `` ``
5	Jac Animation Shop 3	To animate the images and create movies
6	ULead COOL 3D 3.0	`` `` ``
7	Max/Jitter	To conduct and control the simulation

Table 1: List of video/image processing software

Jitter captures the live video using the "`jit.qt.grab`" object as shown in Figure 9. It will be possible to include several 3D-models that have been created using the above mentioned software in Table 1 concurrently using the "`jit.gl.model`" object.

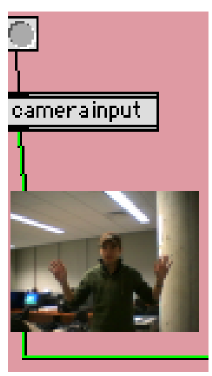

Figure 9: Capturing the real-time video

Once the video is captured, it is partitioned into nine regions and the left hand and the right hand portions are extracted as shown in Figure 10.

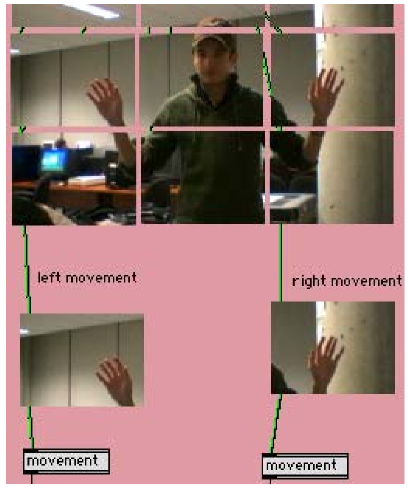

Figure 10: Partitions of the regions of the video

Once the video is captured, it is converted to grey scale using the "`jit.rgb2.luma`". Then the optical flow of the left and the right hands is processed based on the number of pixels using the horn-schunk technique enabled by the jitter object "`cv.jit.HSflow`" as shown in Figure 11.

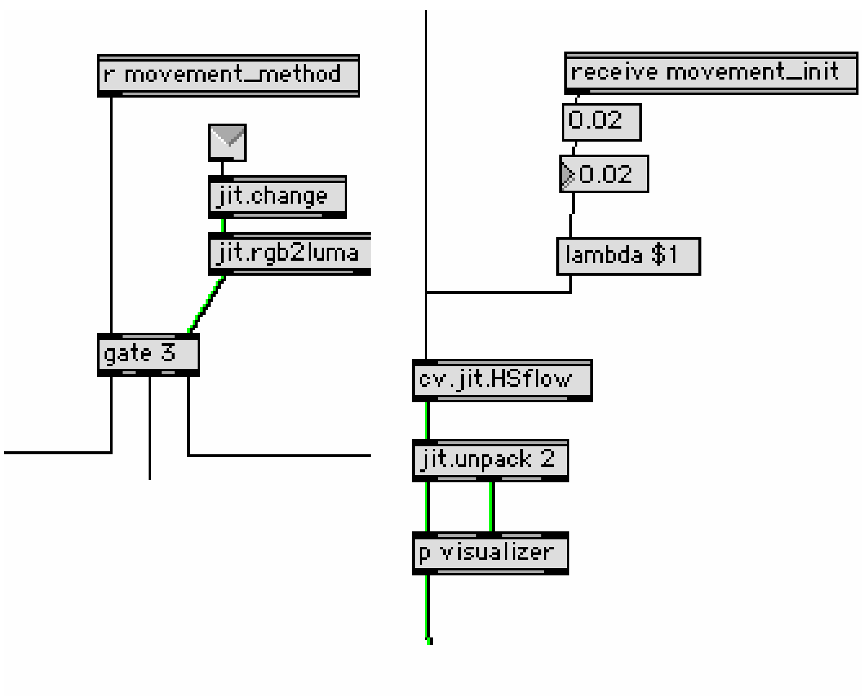

Figure 11: Grey scaling and optical flow using the horn-schunk technique

Using the grey scale images of the right and the left hands that we have obtained, a logical based a mathematical formulae is applied as shown in Figure 12 which in return produces the necessary amount of pixels for the car to shift left or right as shown in Figure 13 and Figure 14.

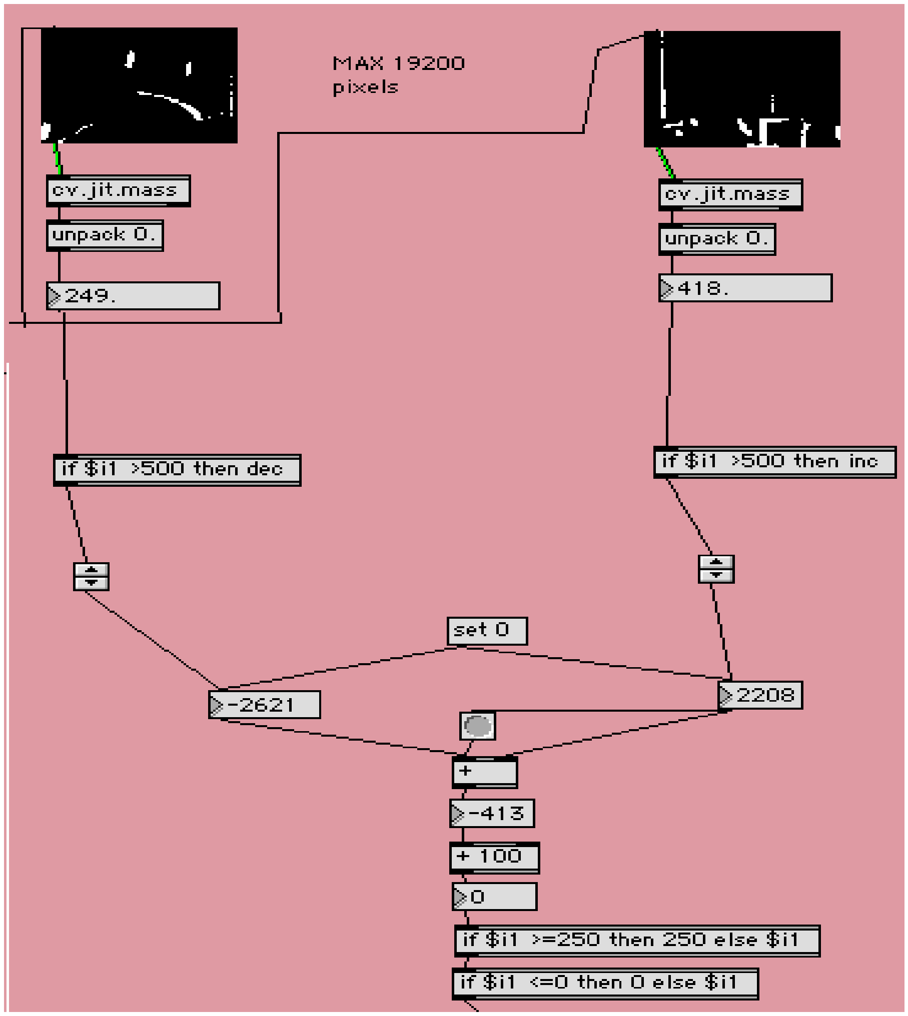

Figure 12: Pixel density and logical formulae

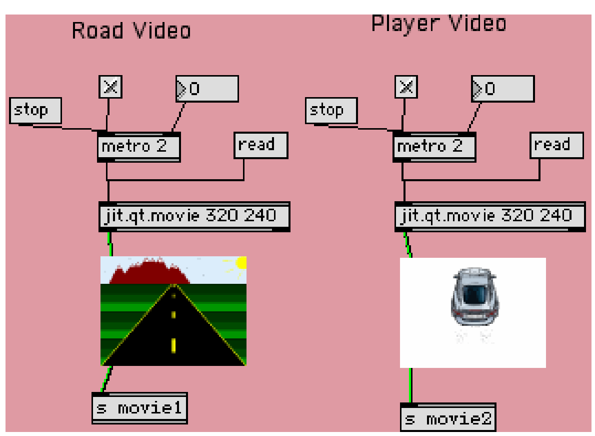

Figure 13: Two separate movies before the merger

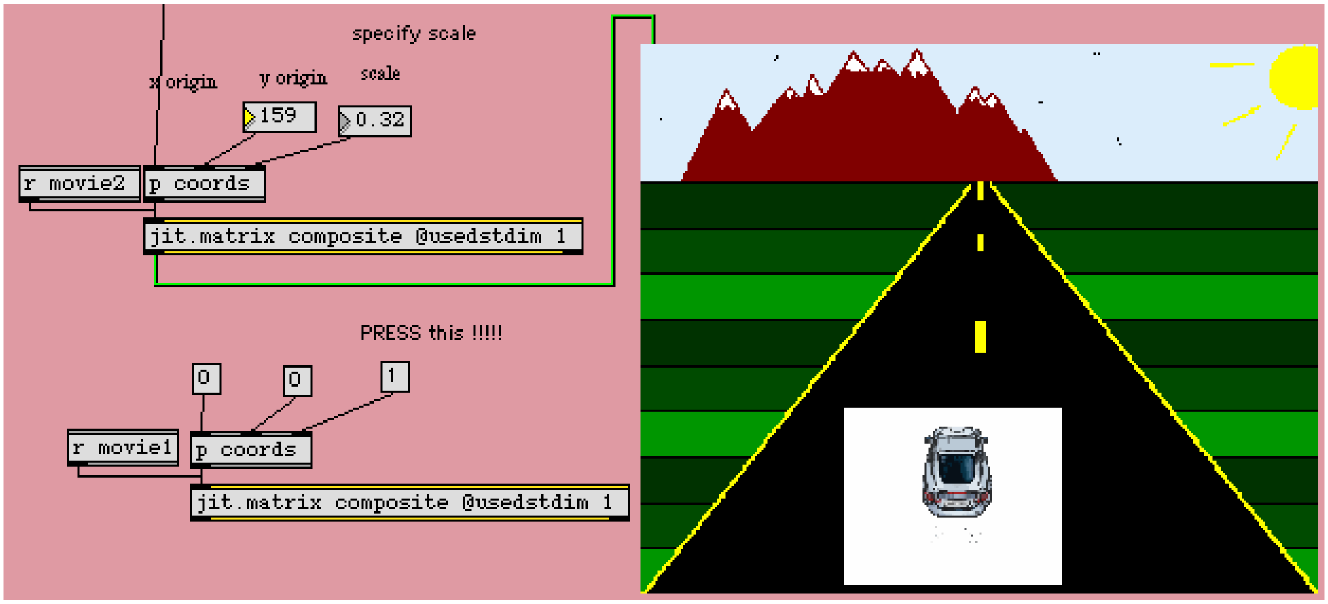

Figure 14: Composition of two movies – road and the symbolized player by car

Here, the "`p coords`" function is used to control the dimension and the location of the car set initially. Finally, the collision and its detection are implemented as shown in Figure 15. The colored obstacles are motion detected and converted into black and white pixels. Using the mathematical operations the collision detection performed.

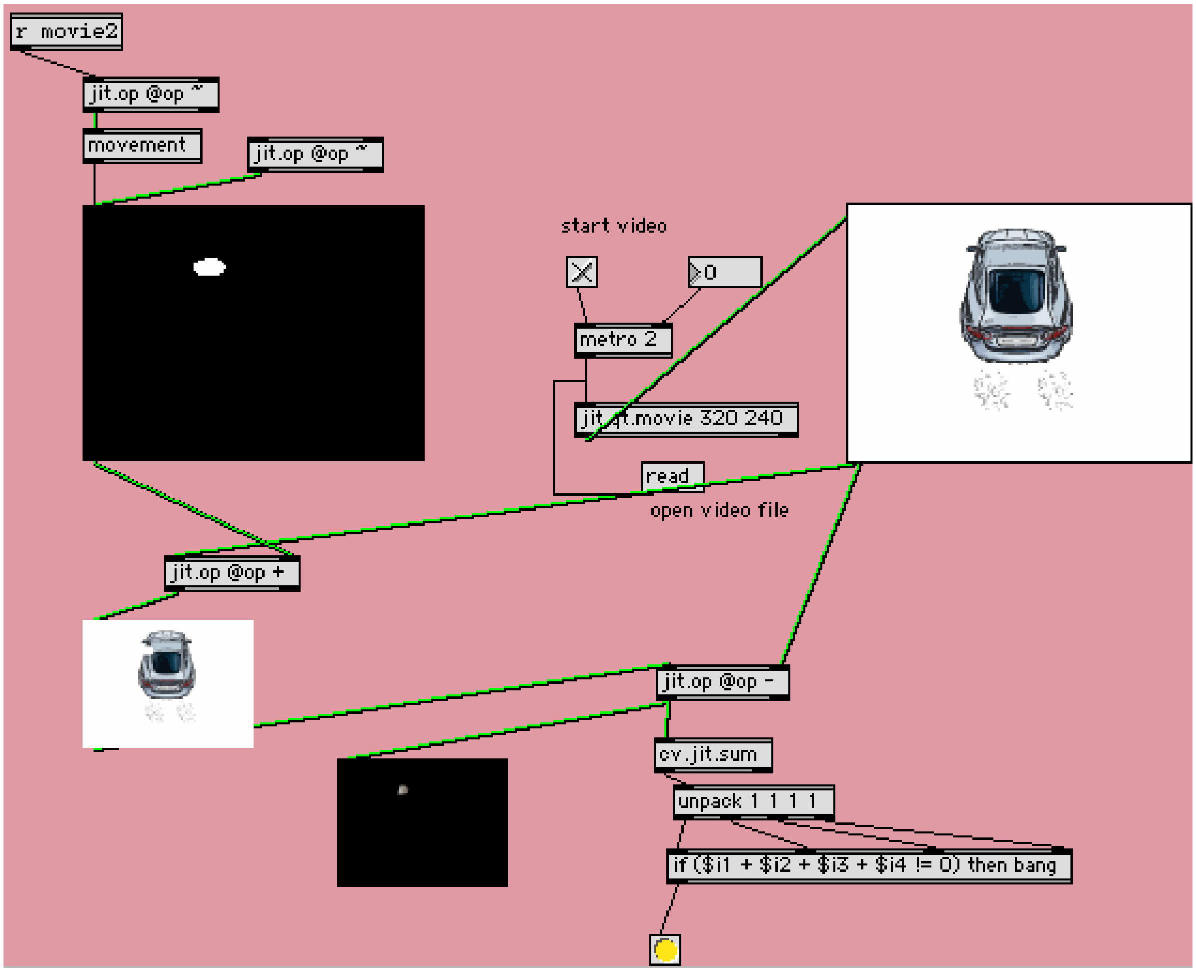

Figure 15: Collision detection

Here the motion of the optical flow is used. The colored input real-time video is converted into black and white grey scale matrix using the `jit.rgb2.luma jitter` object. The volume density of the white pixels is measured. This is input through a jitter object "`cv.jitt.HSflow`" and "`cv.jitt.LKflow`" in order to compute the optical flow of the incoming grey-scale matrix. Here, Horn-Schunk technique is used. Then using the matrix differences between the present and the present matrix frames, the motioned determined.

4 Installation

By Vengadesa Sri Venkadanathan

In order to install our project we required several resources. We could categorize the needed resources in terms of software and hardware.

We needed the following elements of resources in order to demonstrate our gaming system as shown in Table 1

Quantity	Resource	Attributes
1	Computer	Mac OS | Having MAX/MSP Jitter | 2 GB RAM | 10 GB free space
1	Monitor	No constraints
1	Eyesight	Minimum 2 Feet USB Firewire connection
1	Table & chair	Table to set the computer up | chair - for the conductor of the project
1	TV	At least 40'' width hexagram plasma screen display
1	TV stand	
1	S-Video	S-Video cable
1	Pair of speakers	Compatible with Mac OS
1	Space	5x5 square meter area | one side has to be non-transparent – wall
2	Power cords	One 3 meters extension | 1 power-bar

Table 2: Resources needed

As mentioned in Table 1 above, the installation has been set up as shown in Figure 16.

Figure 16: Installation outlook overview

5 Scope of the application

By Vengadesa Sri Venkadanathan

This obstacle avoidance game, although it resembles an amusement game, has symbolic significance such that it could be applied in places whereas a motion has to be detected and collision avoided. As a utility feature, this technique could be implemented in a treadmill as shown in Figure 17 whereas the users have to watch a monitor screen in front of them and carry on with their workout with no worries about stumbling on the edges.

Figure 17: Computer managed obstacle avoidance on a treadmill [2]

If the user is going off the side of the treadmill during their training then it will display a collision state with the beep either on the left or right side in the monitor in order to alert them to rectify their motion naturally.

Moreover, our design techniques could be also be used in some of the real world critical applications. For instance, security motion detector system could adapt our project design techniques. As shown in Figure 18, a motion could be detected and the results could be used in several critical applications such as security camera, alarm system and etc.

Figure 18: Motion detection [3]

6 People and Role

By Vengadesa Sri Venkadanathan

After one of our members of the group, Ahmad Shakhtour, dropping the course, we are currently three people in our team and we have reassessed our responsibilities based on our abilities, strengths and timing constraints.

#	**Responsibilities**	**Performer**
1	Concept Leader	Vengadesa Sri
2	Animation - Images & movies	Vengadesa Sri
3	Animation – assisting	Vincent Garreau
4	Logistic	Vengadesa Sri
5	Integration of all the artifacts	Vengadesa Sri
6	Presentation preparation & delivering	Vengadesa Sri
7	Documentations Assignment 2 Assignment 3 Final Project Report	Vengadesa Sri
8	Web design	Vengadesa Sri
9	Jitter programming – suggestions & modifications	Vengadesa Sri
10	Jitter Programming – planning	Vincent Garreau
11	Jitter Programming - Mathematics	Vincent Garreau
12	Jitter Programming - Collision	Thibault Lecat
13	Sound Programming	Vincent Garreau Thibault Lecat
14	Installation	Vengadesa Sri Vincent Garreau Thibault Lecat

Table 3: Activity distribution

7 Milestones

By Vengadesa Sri Venkadanathan

The members of our team have agreed to operate according to the following scheme as shown in Table 4. We have met schedule as shown. The final presentation due deadlines were met successfully.

Event	Completion
Team project proposal Due	October 23, 2006 √
Documentation – proposal	October 23, 2006 √
Presentation – proposal	October 23, 2006 √
Artifacts – Making movie – road/random objects	October 26, 2006 √
Artifacts – calculations / math involving	October 27, 2006 √
Artifacts – motion detection and video analysis by jitter / coding	October 30, 2006 √
Building the system – trouble shooting of individual components	October 31, 2006 √
Integration – all the design work together	November 06, 2006 √
Calculations on installation – positions/ angles/ background	November 10, 2006 √
Trouble shooting	November 17, 2006 √
Amendments	November 20, 2006 √
Team project – 02	November 20, 2006 √
Team project – 03	November 29, 2006 √
Team project – Final presentation	December 04, 2006 √
Team project – Repetition of Final presentation	December 05, 2006 √

Table 4: Milestones

8 Conclusion

By Vengadesa Sri Venkadanathan

Having visual and audio effects coupled with the human interaction, this game system has been built. This project was liked by many of the players during the demonstration. It ended as an interesting and public adoring project. The choice of this project is appropriate for COMP471 since we are utilizing most of the techniques that we have learnt in the course such as motion detection, real-time video processing, optical flow techniques, Max/Jitter, etc.,

There are several technical interests exist in our project. Mainly, maneuvering and communicating the collision detection via an intermediate object, a car, which symbolically represents the player. This was one great challenge. Another challenging part was about associating the sound streams in the right time during the state changes. Ultimately, the project has been completed fulfilling the scope of the course.

9 References

By Vengadesa Sri Venkadanathan

[1] Obstacle avoidance – A steering behavior
<http://www.red3d.com/cwr/steer/Obstacle.html>

[2] Obstacle Avoidance - training with computer simulated environments,
<http://guide.stanford.edu/96reports/96dev1.html>

[3] Motion Detection Algorithms
<http://www.codeproject.com/cs/media/Motion_Detection.asp>