A Mini-Project Report on

Go Smartwatch Animation

Submitted in 6th Semester

Bachelor of Engineering

In

Computer Science and Engineering

Submitted by

Name: Mohammed Ataaur Rahaman [1DS16CS721]

Name : KS Chandana [1DS16CS716]

Name : Arvind Babu P R [1DS16CS706]

Under the guidance of

PROF. Anitha M

Dept. of CSE, DSCE

2019

Department of Computer Science and Engineering,

DAYANANDA SAGAR COLLEGE OF ENGINEERING

BANGALORE – 560078

Abstract

Wearable computers are expected to become the next big thing but popular press is divided on whether they will be successful. In this paper we review the existing literature on one type of wearable - smartwatches - and extend their definition, in addition to highlighting the need to understand users' everyday appropriation of these technologies. We present initial findings from an on going interview study with ten early adopters that is, to the best of our knowledge, the first to investigate why and how people use smartwatches in real life. We describe everyday use of smartwatches, highlight the added value seen by users, and identify the limitations to mass adoption as expressed by current users.

The Problem

- There are lot of smart watches, but not many have an efficient way to manage its user's personal preferences and monitor its user's body functions.

The Solution

- A Smartwatch (Go Watch) which has exceedingly better user interface and incredible emergency situation handling.
- The animation presented helps in understanding the watch's capabilities during extreme scenarios.

Implementation

- C Language
- Libraries – freeglut.lib, glew32.lib, glew32.lib
- Visual Studio, Xcode

Table of Contents

Introduction

The Mini-Project aims to effectively provide consumers with the best possible Smartwatch with accurate cardiac-monitors adeptly procuring cardiac-graphs and efficiently reporting offbeat heart rates.

Story Line

```
1.      Frame  (TEXT) - Scene I
a.      2015


2.      Frame
a.      In Kumaraswamy Layout...


3.      Frame
a.      Time: 17:30HRS


4.      Frame  (Road and Buildings)
a.      Character X is seen jogging.
b.      Character X collapses suddenly.


5.      Frame  (TEXT)
a.      5 mins later...


6.      Frame  (Road and Buildings)
a.      Character F walks by and notices Character X.
b.      Character F uses his phone to call EMERGENCY.


7.      Frame(TEXT)
a.      15 mins later...


8.      Frame (Road and Buildings)
a.      AMBULANCE reaches the scene.
b.      Character X is declared dead.


9.      Frame  (TEXT) - Shifting to scene II
a.      Present Day...
b.      In Kumaraswamy Layout
c.      Time: 18:00HRS



10.     Frame (Road and Buildings)
a.      Character S is seen jogging.
b.      Character S collapses suddenly.
```

11. Frame (Focus shifts to Character S's hand with Smart watch(GO))
a. GO Watch detects the fall.
b. Sending Emergency Signal in 5secs.
c. Counter starts for 5secs.
d. 4
e. 3
f. 2
g. 1
h. Locating Nearby Hospital...
i. sending Live ECG to Hospital...

12. Frame (TEXT)
a. 5 mins later...

13. Frame
a. Cardio Ambulance reaches the scene.
b. Character S is saved.

14. Frame (TEXT)
a. GO watch is a life saving Band.

15. Frame (GO WATCH animation)
a. Design Show case in 3D animation.

16. Frame (TEXT)

Implementation (source Code)

Link to GitHub: https://github.com/Ataago/Go-Watch-Animation

Header.h

```
#ifdef
__APPLE_CC__
                #include <GLUT/glut.h>
                #else
                #include <GL/glut.h>
                #endif

                #pragma GCC diagnostic ignored "-Wdeprecated-declarations"  // Remove
                deprecation warnings

                #include <stdio.h>
                #include <stdlib.h>
                #include <string.h>
                #include <time.h>
                #include <math.h>


                #include "variables.h"
                #include "writeText.h"
                #include "Objects.h"
                #include "animate_textZoom.h"
                #include "animate_walk.h"
                #include "animate_fall.h"
                #include "animate_walk1.h"
                #include "animate_ambulance.h"
                #include "animate_walkCharS.h"
                #include "animate_fallCharS.h"

                #include "frame0.h"
                #include "frame1.h"
                #include "frame2.h"
                #include "frame3.h"
                #include "frame4.h"
                #include "frame5.h"
                #include "frame6.h"
                #include "frame7.h"
                #include "frame8.h"
                #include "frame9.h"
                #include "frame10.h"
```

```
#include "frame11.h"
#include "frame12.h"
#include "frame13.h"
#include "frame14.h"
#include "frame15.h"
#include "frame16.h"
#include "frame17.h"
#include "frameEnd.h"
```

Objects.h

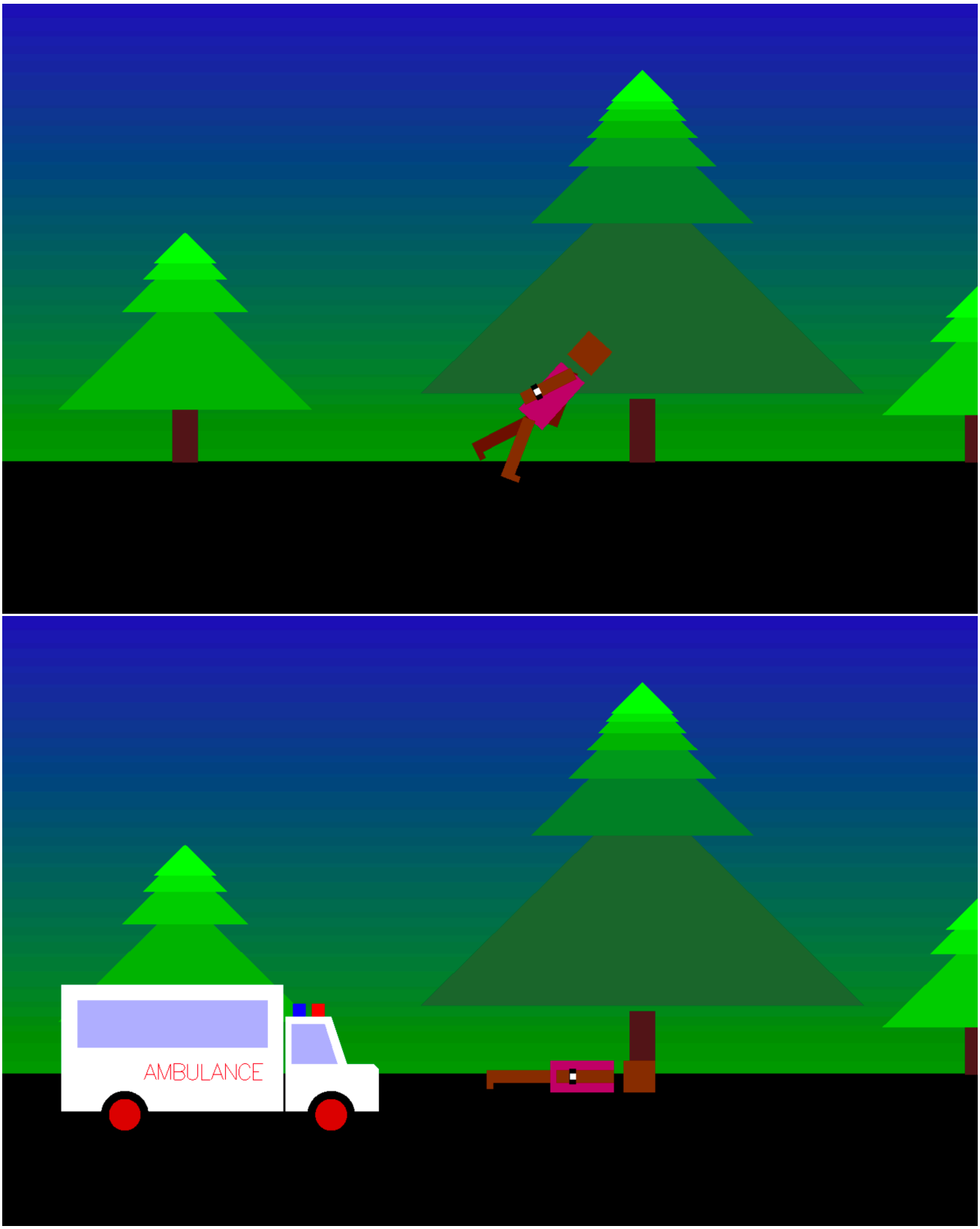
```
Void  square(float x)
{
	glColor3d(0.8, 0, 0);

	glBegin(GL_POLYGON);
	glVertex3f(1 * x, 1 * x, 0);
	glVertex3f(-1 * x, x, 0);
	glVertex3f(-1 * x, -1 * x, 0);
	glVertex3f(x, -1 * x, 0);
	glEnd();
}

void square0()
{
	glColor3d(0.8, 0.0, 0.0);

	glBegin(GL_POLYGON);
	glVertex3f(1, 1, 0);
	glVertex3f(-1, 1, 0);
	glVertex3f(-1, -1, 0);
	glVertex3f(1, -1, 0);
	glEnd();
}

void circleD(double radius, double depth)
{
	glBegin(GL_POLYGON);
	for (double theta = 0; theta < 2 * PI; theta += 0.1)
	{
		glVertex3d(radius * cos(theta), radius * sin(theta), depth);
	}
	glEnd();
}
```

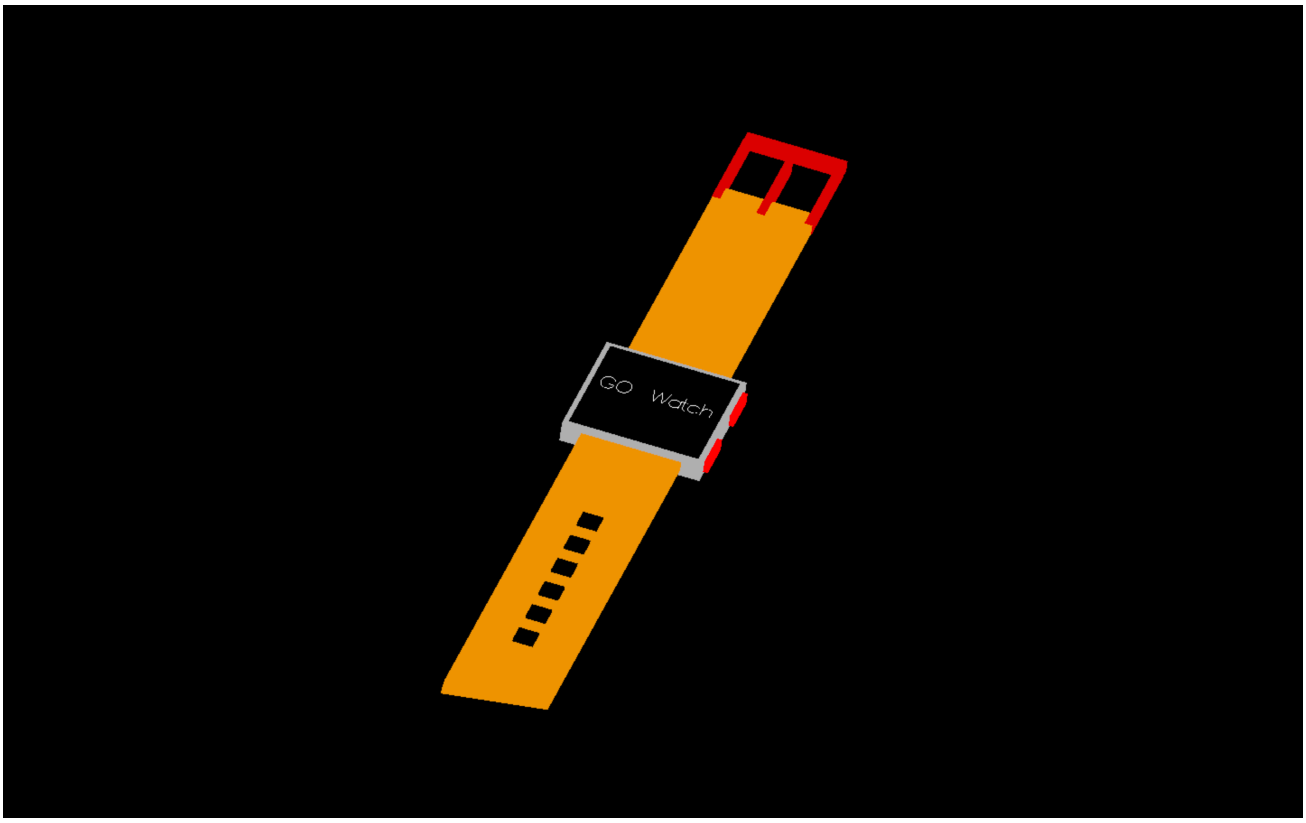
```
void rectangleD(double x1, double y1, double x2, double y2, double z)
{
        glBegin(GL_POLYGON);
        glVertex3d(x1, y1, z);
        glVertex3d(x2, y1, z);
        glVertex3d(x2, y2, z);
        glVertex3d(x1, y2, z);
        glEnd();
}

void ambulance()
{
        char stringO_1[] = "AMBULANCE";

        // Display Text
        glPushMatrix();
        glTranslatef(-90, -100, 3);
        glScaled(0.5, 0.5, 0);
        glColor3f(1, 0, 0);
        drawText(stringO_1);
        glPopMatrix();

        // Ambulance Body
        glColor3d(1, 1, 1);
        rectangleD(-350, -200, 350, 200, 2);

        // Ambulance Engine
        glBegin(GL_POLYGON);
        glColor3d(1, 1, 1);
        glVertex3d(355, -200, 2);
        glVertex3d(650, -200, 2);
        glVertex3d(650, -65, 2);
        glVertex3d(635, -50, 2);
        glVertex3d(550, -50, 2);
        glVertex3d(500, 100, 2);
        glVertex3d(355, 100, 2);
        glEnd();

        // Lights
        glColor3d(0, 0, 1);
        rectangleD(380, 100, 420, 140, 2);
        glColor3d(1, 0, 0);
        rectangleD(440, 100, 480, 140, 2);

        // Back Tire
        glPushMatrix();
        glTranslated(-150, -210, 0);
        glColor3d(0, 0, 0);          // Tire
```

```
        circleD(75, 3);
        glColor3d(0.8, 0, 0); // Rim
        circleD(50, 4);
        glPopMatrix();

        //Front Tire
        glPushMatrix();
        glTranslated(500, -210, 0);
        glColor3d(0, 0, 0);          // Tire
        circleD(75, 3);
        glColor3d(0.8, 0, 0); // Rim
        circleD(50, 4);
        glPopMatrix();

        //Window Back
        glColor3d(0.7, 0.7, 1);
        rectangleD(-300, 0, 300, 150, 3);

        // Window Front
        glBegin(GL_POLYGON);
        glVertex3d(375, -50, 3);
        glVertex3d(520, -50, 3);
        glVertex3d(480, 75, 3);
        glVertex3d(375, 75, 3);
        glEnd();
}


void leaf(double x, double z)
{
        glBegin(GL_POLYGON);
        glVertex3d(-x, -x / 3, z);
        glVertex3d(x, -x / 3, z);
        glVertex3d(0, x * 0.65, z);
        glEnd();
}

void tree(double h, double z)
{
        green = 1;

        glPushMatrix();
        glTranslated(0, h * 170, 0);
        for (int i = h; i > 0; i--)
        {
                glPushMatrix();
                glColor3d(0.2, green, 0.2);
                glTranslated(0, -h * 120 / i, 0);
                leaf(h * 100 / i, z);
```

```
                glPopMatrix();

                green -= 0.1;
            }
            glPopMatrix();

            glColor3d(0.3, 0.1, 0.1);
            rectangleD(-40, -100, 40, 100, z);

        }
```

Results (Screen shots of the Program)


AMBULANCE


GO Watch

Benefits

These are some of the points how Education can benefit with Smart Watches.

- **Get Real time notifications about classes and exams:** Android Wear powered smart watches can be connected with school management apps that will notify students of classes in a day and changes in schedules in real time.

- **Assist teachers in marking attendance:** Apps for the school management systems can integrate with Android Wear and assist teachers to mark student attendance by tracking student presence in a classroom.

- **Plan daily study routines during class**: With the easy access that these devices provide students can immediately note down and plan activities & assignments created by teachers in classes.

- **Keep track of fitness during exercise/games:** With inbuilt sensors in the devices, students can keep track of their workout in school games/gym and school physical instructors can plan specific regimens for their students using this data.

- **Voice Recording:** Students can record lectures right from the Android Wear without fumbling with recording devices and later transfer to their phones or PCs for future reference.

Future Scope

While some may argue that further integrating technology into our bodies could distract from our day-to-day interactions with other people, wearable technology can actually enhance our lives.

There is definitely a future for the smartwatch. The market is still extremely young. Users start realizing the actual potential. There is enough criticism to create realism."

Alternatively, never believe the hype, because smartwatches won’t change the world in the ways many people promised. SmartWatches of today simply are the external output/input devices for our phones. They notify us about the most relevant information from our phone and serve as minimalistic input for some of the interactions we have with our devices. Input through active interactions like selecting a response or passive input by letting it monitor the environment of the user with one of the sensors.

As it turns out most smartwatches still are most valued for doing what smartwatches are best at: showing the time.

Conclusion

A SmartWatch is being able to receive unobtrusive notifications in social situations, hence reducing mobile phone dependency.

Despite a general appreciation for inconspicuous notifications, early adopters from this study are still confused as to what the real benefit of a SW is. When compared to smartphones, they still do not offer enough additional functionality in order for them to take off for mass adoption.

Aesthetics desirability of SmartWatches highly depends on personal preferences. As our participants point out, SmartWatches have not entirely replaced traditional watches and users have different opinions about what a SW should look like.

Smartwatches carry a lot of potential, but to call the current devices the future...? We think that the question is not if it becomes the future, but how big a role the Smart Watch will play in our near future. There are a few key variables that will define the size of this role. One of these drivers is the evolution of the components that are needed to improve the experience. If these components keep improving rapidly, this could have a serious impact on the potential role of the watch.

References

1. Trew, James. "Sony SmartWatch 2 review". Engadget.com. Retrieved 2014-07-22.

2. Cooper, Daniel. "Garmin's new app turns Sony's Smartwatch 2 into a tiny sat-nav". Engadget.com. Retrieved 2014-07-22.

3. Doensen, Pieter. "Watches with Memory and Database".
WATCH. History of the modern wrist watch. Pieter Doensen. Retrieved 17 September 2010.