INF01047 - Fundamentals Computer Graphics - 2011/01

Group: Diogo Raphael Cravo, João Luiz Grave Gross

Registry: 181051, 180171

Professor: Marcelo Walter

Fun-flight: The user's manual

A *software* by students Diogo Raphael Cravo and João Grave Gross.

The *software*: fun-flight is a flight simulator application developed on linux with the use of languages c/c++ (focused on Opengl and glut libraries) and a semi object oriented paradigm. It's the final work on 2011's first semester for INF01047 (fundamentos de computação gráfica), on the Inf (Instituto de Informática) from UFRGS (Universidade Federal do Rio Grande do Sul).

What is it?

Fun-flight is a simple flight-simulator. It has an airplane (a boeing), two citys (which are the same and are located near eachother), an airport (with no texture, just a white geometry), a grass ground and a sky (this one is a big sphere).

The *software* also has a light that spins around the ground, attempting to simulate a sun (it can be seen as the bright part on the sky in the next image).

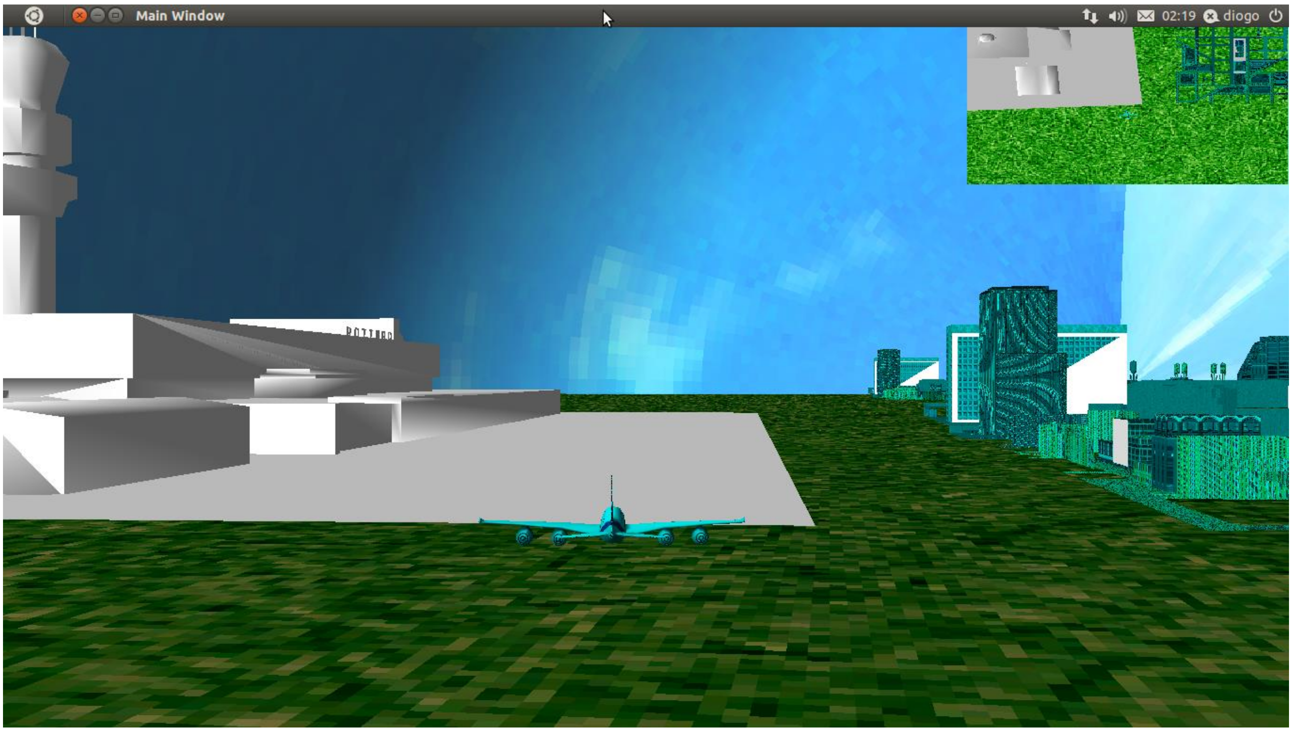


Image showing the first screen seen by user.

This image shows also both viewports that fun-flight offers: a main big one – where the playing happens – and a small upper right one – that gives the player a bigger vision of what's around the plane.

When playing, the user is able to switch between 4 viewmodes, they are summarized in the next image.

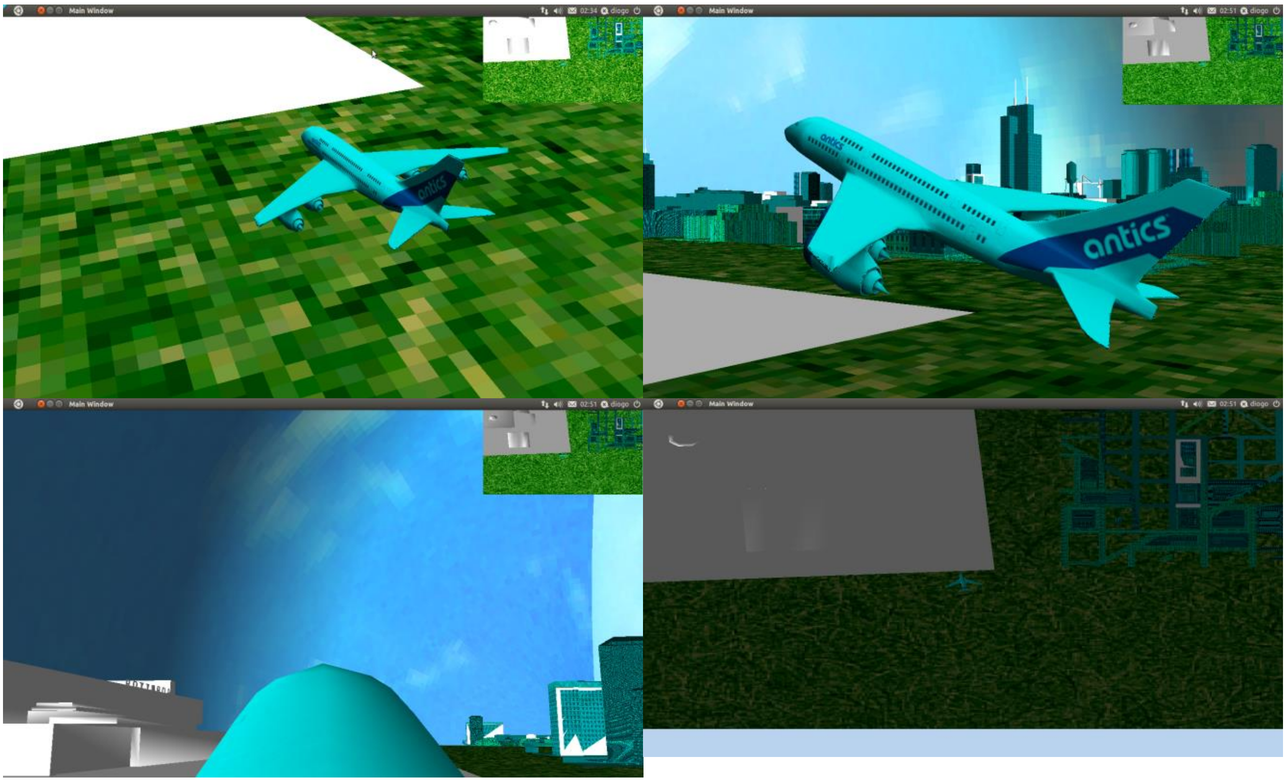


Image showing possible viewports. From top to bottom, left to right: viewmode 1, 2, 3 and 4.

The four possible viewmodes are 1 (for camera constrols), 2 (for plane control, camera not attached to it), 3 (for a camera attached to the plane, moving with it) and 4 for a camera up in the sky.

Get to action!

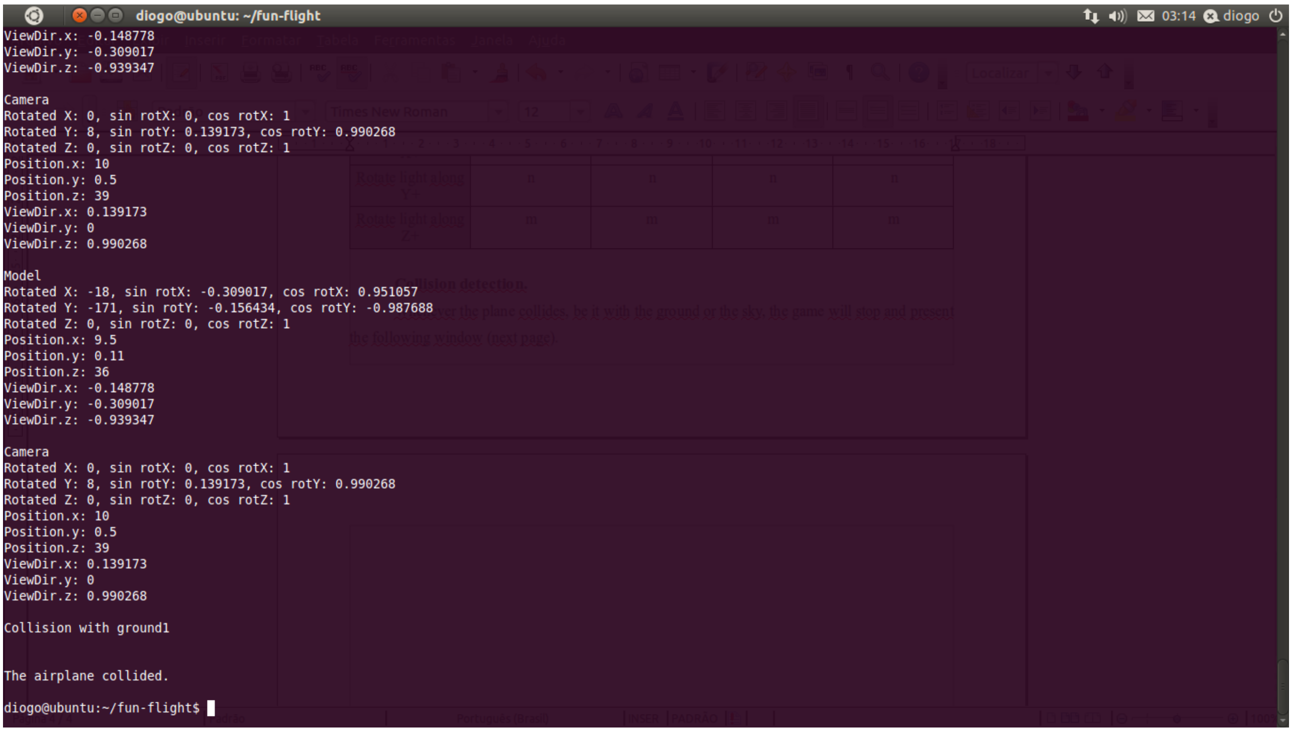
For the airplane and the cameras, there are many controls that make motion possible. The following table lists them all.

Function/Key	Viewmode 1	Viewmode 2	Viewmode 3	Viewmode 4
Zoom in	Left mouse button	Left mouse button	Left mouse button	Left mouse button
Zoom out	Right mouse button	Right mouse button	Right mouse button	Right mouse button
Accelerate plane forward	-	w	w	w
Slow plane	-	s	s	s
Move plane up	-	r	r	r
Move plane down	-	f	f	f
M ove plane to right	-	d	d	d
Move plane to left	-	a	a	a
Rotate plane along X+	-	r	r	r
Rotate plane along X-	-	f	f	f
Rotate plane along Y+	-	q	q	q
Rotate plane along Y-	-	e	e	e
Rotate plane along Z+	-	x	x	x
Rotate plane along Z-	-	z	z	z
Move camera forward	w	-	w	-
Move camera backward	s	-	s	-
Move camera up	r	-	r	-
Move camera down	f	-	f	-
Move camera to right	d	-	d	-
Move camera to left	a	-	a	-

Rotate camera along X+	t	-	t	-
Rotate camera along X-	g	-	g	-
Rotate camera along Y+	q	-	q	-
Rotate camera along Y-	e	-	e	-
Rotate camera along Z+	z	-	z	-
Rotate camera along Z-	x	-	x	-
Rotate light along X+	b	b	b	b
Rotate light along Y+	n	n	n	n
Rotate light along Z+	m	m	m	m

It detecs collisions!

Whenever the plane collides, be it with the ground or the sky, the game will stop and present the following window (next page).



```
ViewDir.x: -0.148778
ViewDir.y: -0.309017
ViewDir.z: -0.939347

Camera
Rotated X: 0, sin rotX: 0, cos rotX: 1
Rotated Y: 8, sin rotY: 0.139173, cos rotY: 0.990268
Rotated Z: 0, sin rotZ: 0, cos rotZ: 1
Position.x: 10
Position.y: 0.5
Position.z: 39
ViewDir.x: 0.139173
ViewDir.y: 0
ViewDir.z: 0.990268

Model
Rotated X: -18, sin rotX: -0.309017, cos rotX: 0.951057
Rotated Y: -171, sin rotY: -0.156434, cos rotY: -0.987688
Rotated Z: 0, sin rotZ: 0, cos rotZ: 1
Position.x: 9.5
Position.y: 0.11
Position.z: 36
ViewDir.x: -0.148778
ViewDir.y: -0.309017
ViewDir.z: -0.939347

Camera
Rotated X: 0, sin rotX: 0, cos rotX: 1
Rotated Y: 8, sin rotY: 0.139173, cos rotY: 0.990268
Rotated Z: 0, sin rotZ: 0, cos rotZ: 1
Position.x: 10
Position.y: 0.5
Position.z: 39
ViewDir.x: 0.139173
ViewDir.y: 0
ViewDir.z: 0.990268

Collision with ground1


The airplane collided.

diogo@ubuntu:~/fun-flight$ 
```

It's also possible to note in this print some of the prints the *software* does. It shows every rotation/translation done to the plane and/or the cameras. On the beginning, it tells the user all files that are being read and images that are loaded.

Final thanks...

This is what fun-flight is capable of doing. We hope you enjoy using it. Thank you.