# Computer graphics project

VISUALISATION OF FIRE AND FOREST FIRE

Jeon Hyun Soo
COSC3000 | 42865320

# Table of Contents

# 1 Project Overview

## 1.1 Introduction

Fire is a natural phenomenon that occurs due to heat and light energy emitted from the process of combustion which is a type of violent chain reaction of oxidation. Molecules in the fire system can pass the active energy to another nearby molecule if the heat produced by the oxidation is enough. This continuously produces more heat as the oxidation process continues.

As fire is a result of oxidation fire requires oxygen to burn. 3 main key elements that is required to start a fire are oxygenation, flammable combustible materials (gas, petroleum, dried leaves, paper) and source of heat that goes over ignition point (electrical sparks, friction energy, sunlight). Fire is composite of liquid or gas that interacts with oxygen, high heat gas that gets ionized and high heat plasma.

In natural environments fires usually occur due to heat from sunlight or sparks from thunder. As heat continues and without moisture from the rain, leaves, barks and grasses dry up and continuous heat to those dried materials will start to burn once the heat level goes over the ignition point.

There has been large number of natural occurring wild fires and house fires. It would be interesting to view the nature of fire and its movement with its key elements controlled would be a great way for people to understand the fire.

## 1.2 Aim and Purpose

This project aims to use basic computer graphic animations to simulate the movement of flames and fires. People usually know what fire is and how it looks like, but they do not understand the characteristics of the fire.

By simulating the fire and flames with computer graphics, fire can be seen more easier and by controlling the fire people can earn better understanding from it.

Also, by demonstrating how fires spread from a point to a huge space, people can get better understanding for wild fires and forest fires and how they spread. This can hopefully be able to grow better understanding to people about the fire and wild fires and be aware of them.

# 2 Method

## 2.1 Development Environment

The development environment of the program is important as it is used to build the codes that forms the computer graphics. In this project mixture of C++ and Python has been used with OpenGL technology to simulate the fire. OpenGL is Open Graphics Library API which is easily accessible and one of most used technology in computer graphics field.

### 2.1.1 PyOpenGL

PyOpenGL is a standardized python library that connects Python and OpenGL together. Without PyOpenGL, OpenGL cannot be run with Python codes, so it is important to install this library to run the OpenGL API in python. PyOpenGL also as acceleration code which speeds up the process.

```
python -m pip install PyOpenGL PyOpenGL_accelerate
```

### 2.1.2 PyGame

As a canvas, python's game module library PyGame was used to render the computer graphics on. PyGame is usually used to build games with python codes, but also it can be used as a great simulation tool. PyGame offers number of great graphics library to render the simulation easily.

```
python -m pip install pygame
```

## 2.2 Simulation

Fires, flames and explosion are frequently used in computer graphics field as it is largely used in videos and games. Simulating fire and flames with computer graphics is not an easy task as the system of fire is complex. Creating a simple none moving 2D fire graphics would not be such a difficult task, however simulating a live fire that moves is a difficult task. There are many numbers of aspects to consider.

To include every aspect and create a realistic live fire, it will require high technology and large system formed with high computing. However, there is no need to present high-quality simulation of realistic fire, so more simple simulation will be created.

## 2.3 Basic Fire

First graphics is basic fire modelling of fire and flames. It shows how the fire burns and how the movement changes by outer effects.

### 2.3.1 Particles

Particle system is formed with large number of small particles in a set sharing similar characteristics. These particles in a set shares similar velocity, position to one another, which is similar to characteristic of molecules of fire. Particles are usually a simple points or quadrangles with images and textures. Particle system is very simple but effective

There are number of different particle types, including billboard particles, 3d particles, point sprites and trail particles.

### 2.3.2 Rendering

There are number of methods to consider when rendering the fire after it is formed with particles. Rendering gives texture to the computer graphics.

Simple rendering methods includes billboarding and blending. Billboarding is to have fixed camera and rotating the image to always face the camera. Blending makes colours in the background to fade through to the foreground.

In this project billboarding was used to render the particles. Billboarding fixes the camera and does not simulate other thigs so it has very low computational requirement. As we focus on the fire itself and there is no need to rotate or shift around, billboarding is enough to be used.

### 2.3.3 Other methods

There are number of other methods or technologies to implement fire with computer graphics. In fields such as special effects used in videos, some methods like spring-mass modelling are used to model the kinematics of the fire which allows the gravity and wind to affect the fire.

Some other techniques are to simulate fire by rendering the fire over the surface of polygon shape mesh. Generating points over the surface of polygon and these individual fire points forming a big fire.

In this project particles have been chosen to implement the fire for this project, as it is light and simple, not requiring high power computing. It can be considered to be enough to fit the purpose of this project.

Some other methods still use particles to generate fire, but use advanced rendering techniques to make the fires look more realistic. Using such rendering techniques as polygon surface model can create 3D models that are more realistic.

### 2.3.4 Algorithm

The algorithm for this simulation is quite simple. First it starts as creating and destroying particles. Each fire particles will be given lifetime value and generated in random areas, and when its lifetime is over, the particles will be destroyed.

### 2.3.5 Implementation

The fire is implemented by creating particles at random positions inside a circular area. Each particle will be given with lifetime of 1 and as will be reduces as time goes. Once the lifetime hits 0, that particle will be removed. Each of these particles will have colour of red-orange and as lifetime reduces, the colour changes to a darker colour becoming black one it reaches 0. The particles move around if wind blows by giving outer velocity to certain directions.

## 2.4 Forest Fire

To understand how the fire actually spreads around another graphics of fire's movement have been modelled in 2D shape.

### 2.4.1 Algorithm

The algorithm that is used in simulating the forest fire is a well-known mathematical theory by Drossel and Schwabi developed in 1989 and 1992. This algorithm is defined is defined by four rules which are executed simultaneously:

A burning cell turns into an empty cell

A tree will burn if at least one neighbor is burning

A tree ignites with probability *f* even if no neighbor is burning

An empty space fills with a tree with probability *p*

The controlling parameter of this model gives the average number of trees planted between two ignition points by lightning. The equation is:

$$p/f$$

$$f \ll p \ll T_{max}$$

where $T_{max}$ is the burn time of the largest cluster

### 2.4.2 Simulation

The simulation of this model was quite simple as it has been created in simple 2D model with 3 types of boxes. 3 types of boxes are black empty cell which is burnt area, green cell

which is tree and red cell which is fire. After the implementation of the forest fire algorithm, each cell was modeled.

### 2.4.3 Basic algorithm simulation

Before simulating this algorithm graphically, the algorithm has been tested in basic python. Normal forest of 400 cells starts with a forest with 251 trees.

```
Of    400 cells,    251 are trees of which      0 are currently burning.
%,  0.000%)
Print/Quit/<int>/<return>      0: p
TT..TTTT..TTTTT.T..T
...T.T.T..TT..TTTTTT
TTT..T.TTTTTTTT.T...
.T.TT.T.T.TTT.T..TT.
.TT....T.T.TTTTTTTT.
TTT.T.TTTT..TT..TTTT
TTT.TT..T.TTT.TT.TTT
T.T...T..TTT..TT.TTT
T..T.TTT.TT....T....
TTTT.TTTTTT....TT.T.
T..TTTTTT..TTTTTTTTT
.T.TTTT...T..TTT..T.
TTTT.TTTTTT.T..TT..T
...TT.TT.TTTTT.T....
TTTTTTTTTTTT..TTT.T
TTT..TT.TTTT.TT.TTTT
TTT.TTT.T.TTT.T.T...
..TTT.T.TTT..TTTTTT.
T.T.TT..T..TT.T.T.T.
T.T..TTTT..TTT.TTTTT
```

*Image 1 – First Stage*

There are no burning trees

```
TT..TTTTTTTTTT.T..T
...TTT.T..TT..TTTTTT
TTT..T.TTTTTTTT.T...
.T.TT.T.T.TTT.T..TT.
TTT...TT.T.TTTTTTTT.
TTTTT.TTTT..TT..TTTT
TTT.TT..T.TTT.TT.TTT
T.T...T..TTT..TT.TTT
TT.T.TTT.TTT...T....
TTTT.TTTTTT..T.TT.TT
T.TTTTTT..TTTTTTTTT
.T.TTTT...T..TTT..T.
TTTT.TTTTTT.T..TT..T
.T.TT.TT.TBBBB.T....
TTTTTTTTTB....TTT.T
TTT..TTTTB...B.TTTT
TTT.TTT.T.B...B.T...
..TTT.TTTTBB.BTTTTT.
T.T.TT..T..TT.T.T.T.
T.T..TTTTT.TTT.TTTTT
Of    400 cells,    251 are trees of which     14 are currently burning.
```

*Image 2 – Second stage with fires start*

From a random point fire starts and burns nearby trees

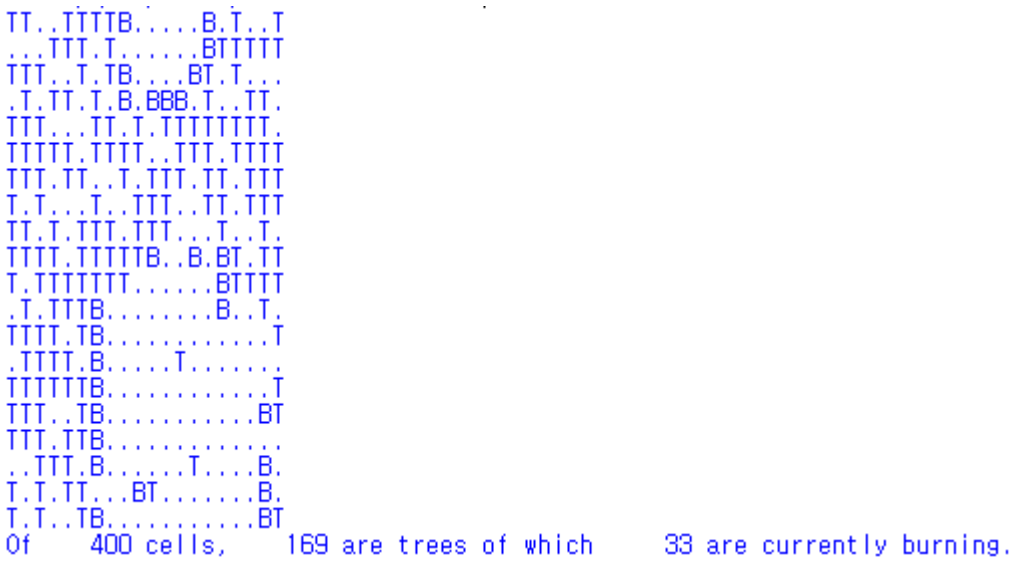


*Image 3 – Third Stage of the model with fire increasing*

As fire gets bigger it destroys and burns nearby fires and reaches further to burn the next trees.

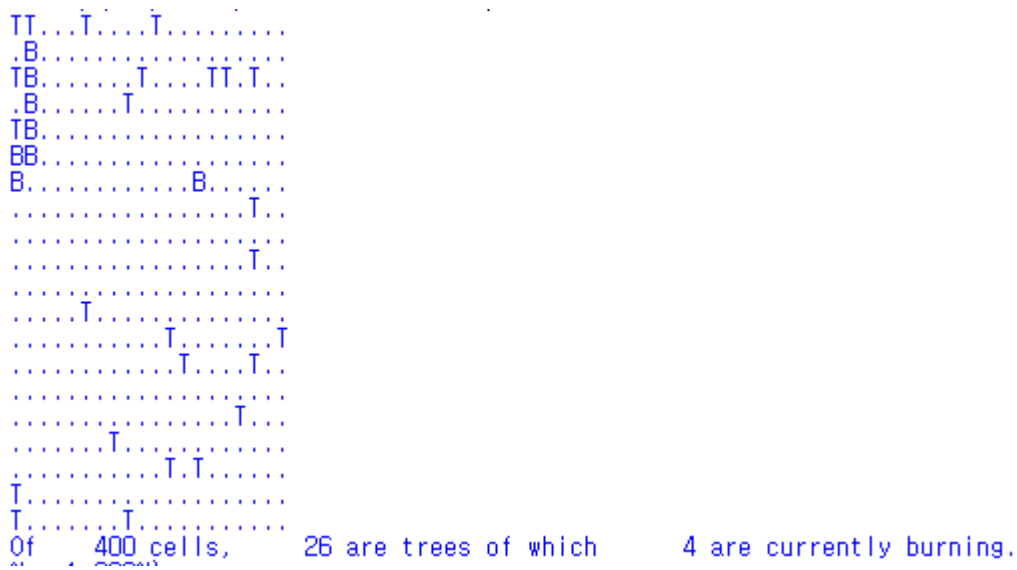


*Image 4 – Forth stage of the model with lots of trees burnt down*

As the fire continued most of the trees have been burnt down and destroyed the forest.

# 3 Result

After the development of computer graphics, the models have been created. Both of the models were put on PyGame to be rendered and simulated in it.

## 3.1 Live fire model

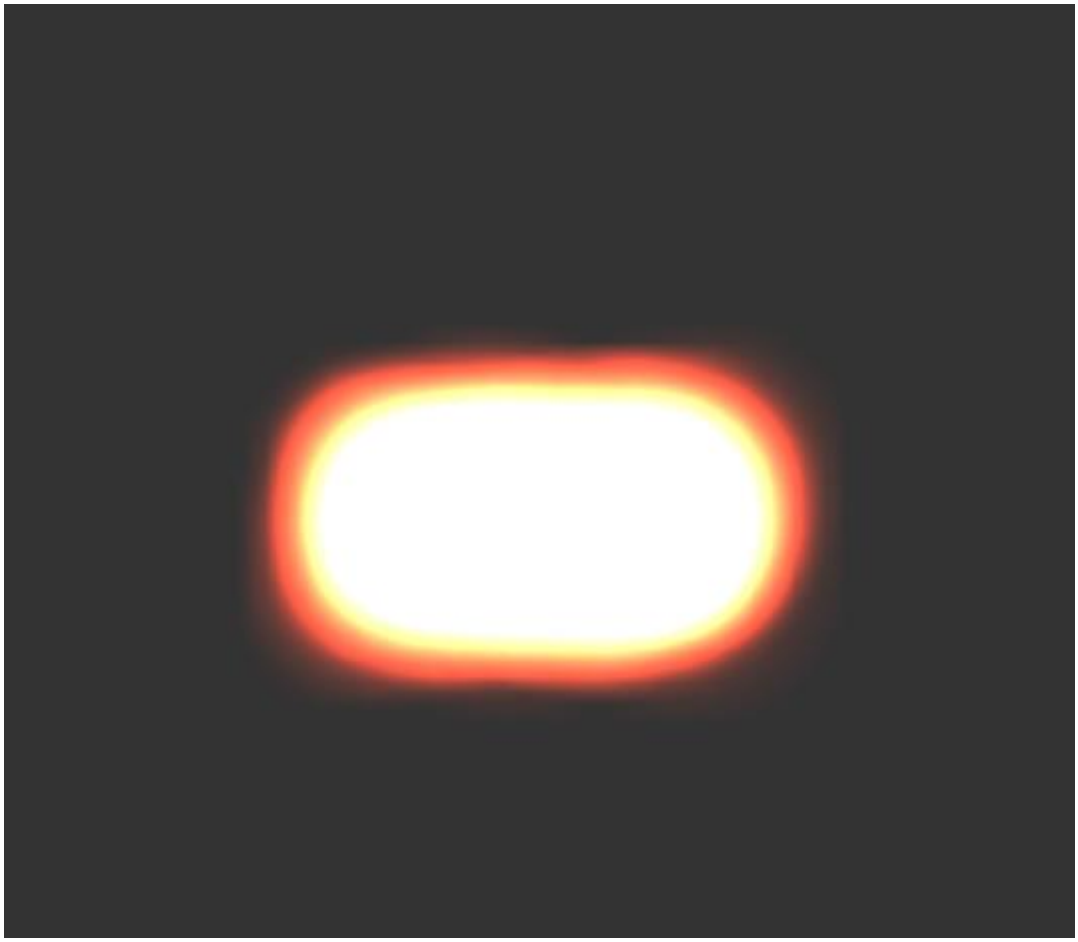

*Image 5 – The beginning of the fire*

In the initial stage, particles are created and is clustered in the center of the fire. Before it gets spreads and move toward upper direction most particles have same colour and positions.

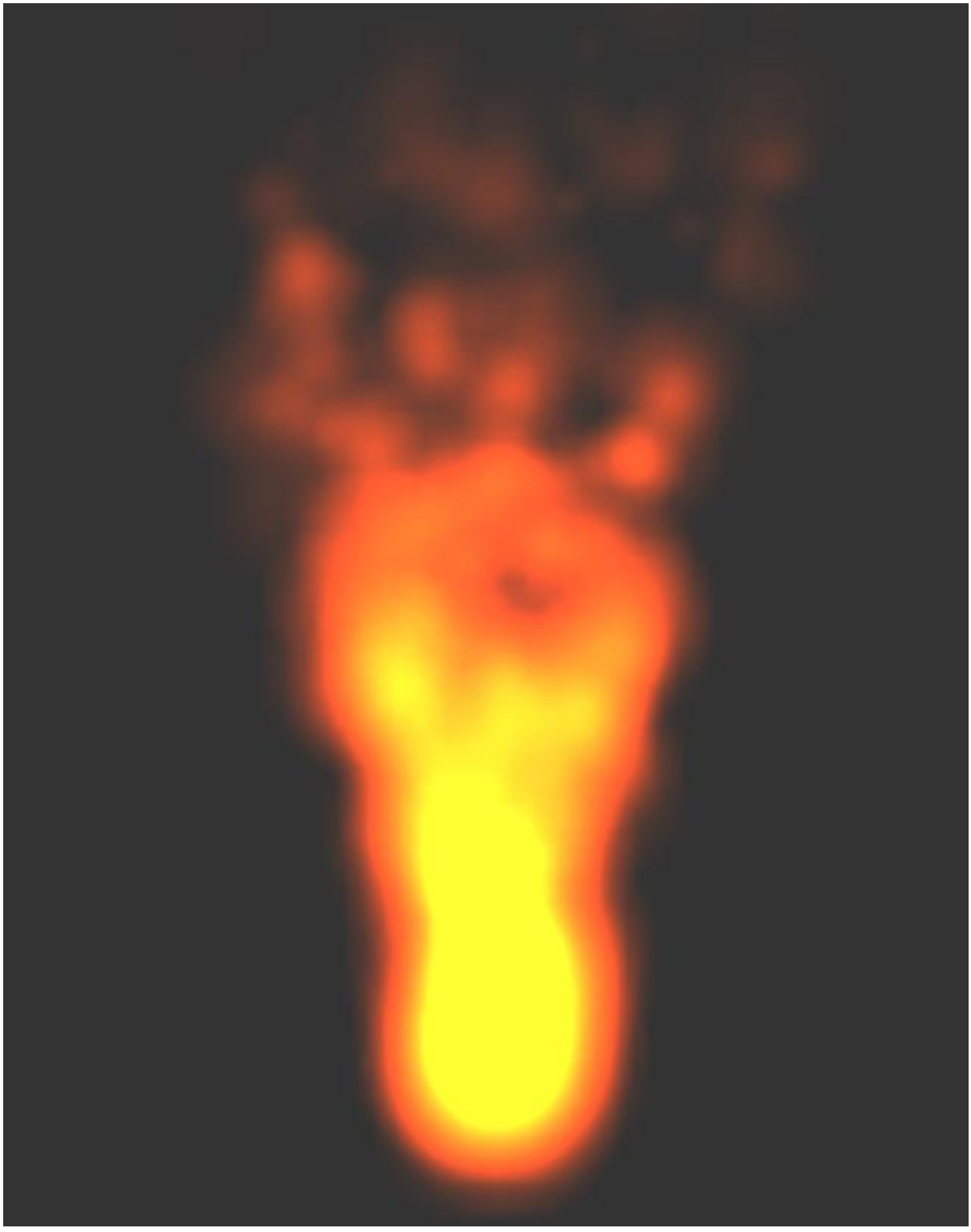

*Image 6 – Normal state of fire*

The normal state of the fire burning up. Fir particles can be seen.

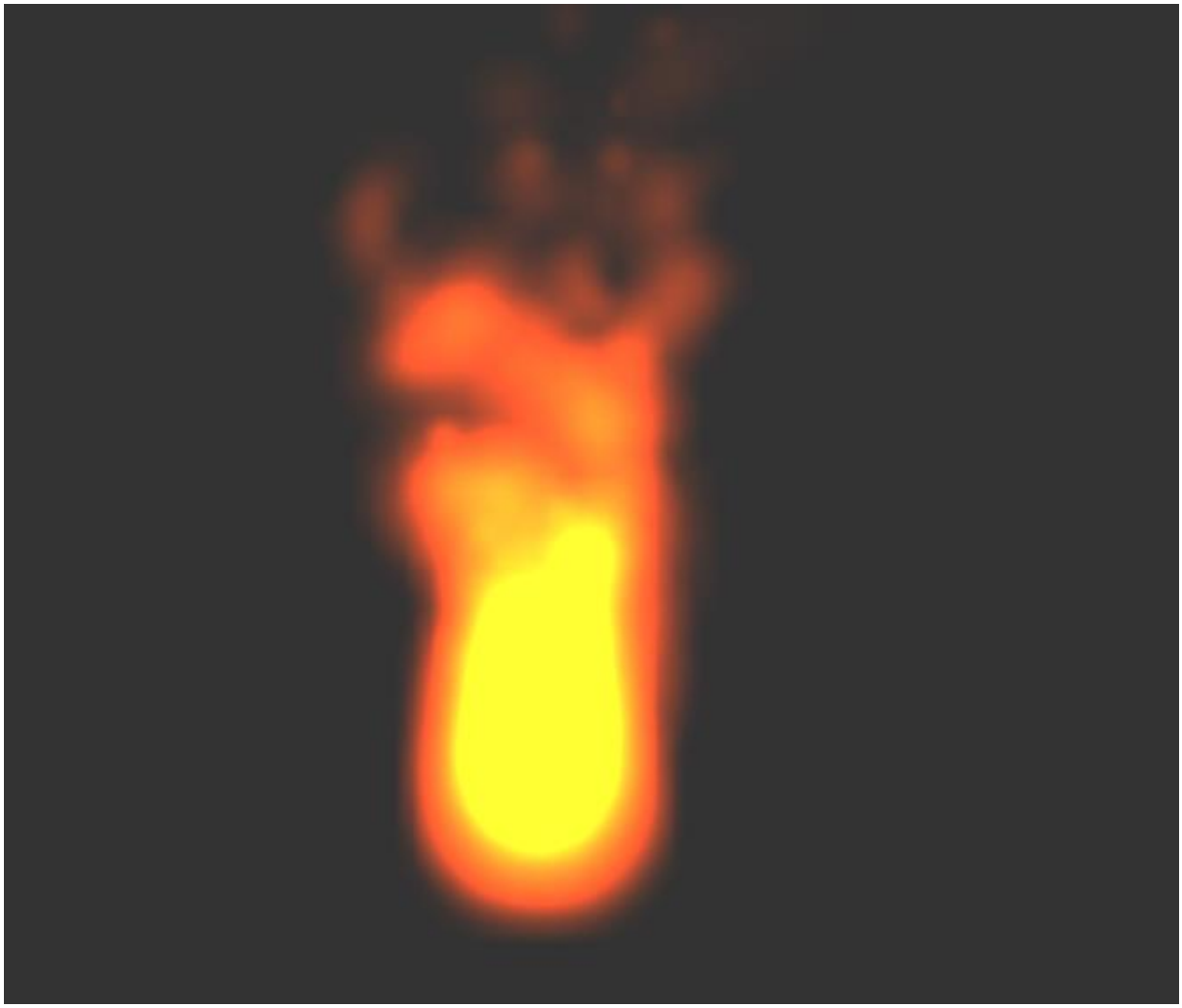

*Image 7 – Fire in reduced size*

The size of the fire is reduced as the flammable material in the center reduced. Less material to burn makes the size of the fire to shrink.

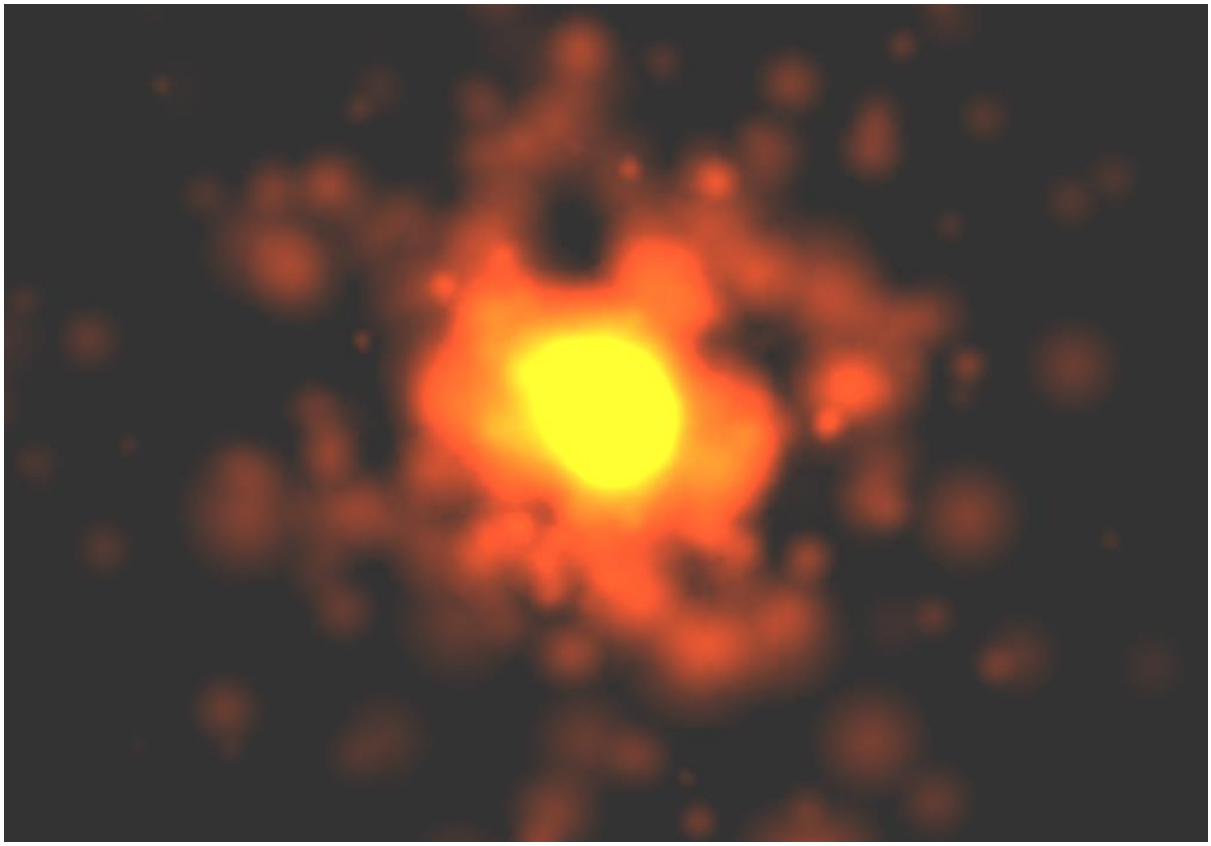

*Image 8 – Bird's eye view of the fire*

Fire that has been taken from the above. The center is hot with yellow and particles spreads around from the center point.

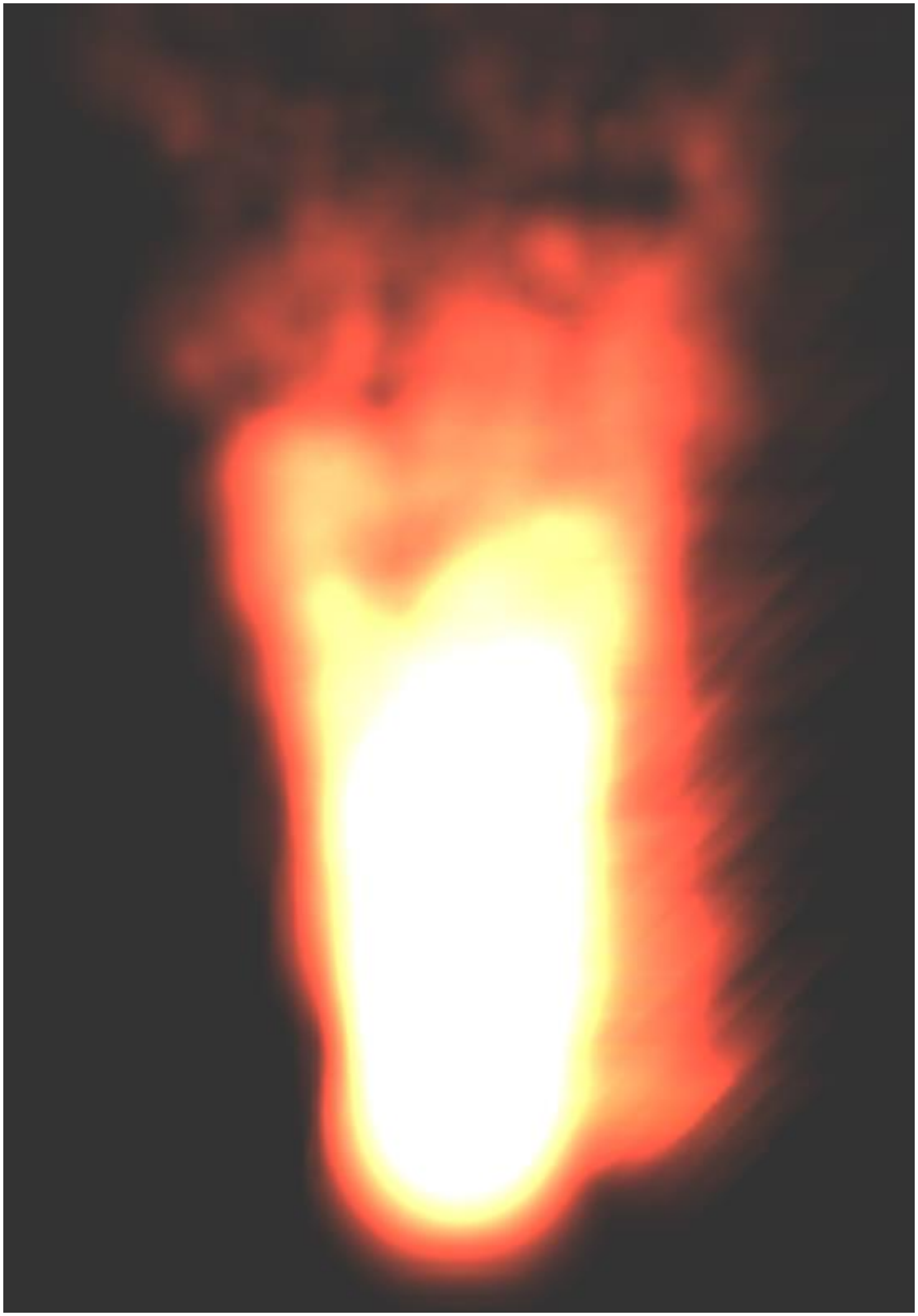

*Image 9 – Wind blowing from left*

As the wind blows from the left particle that forms the fire move to the left spreading the particles and fire more.

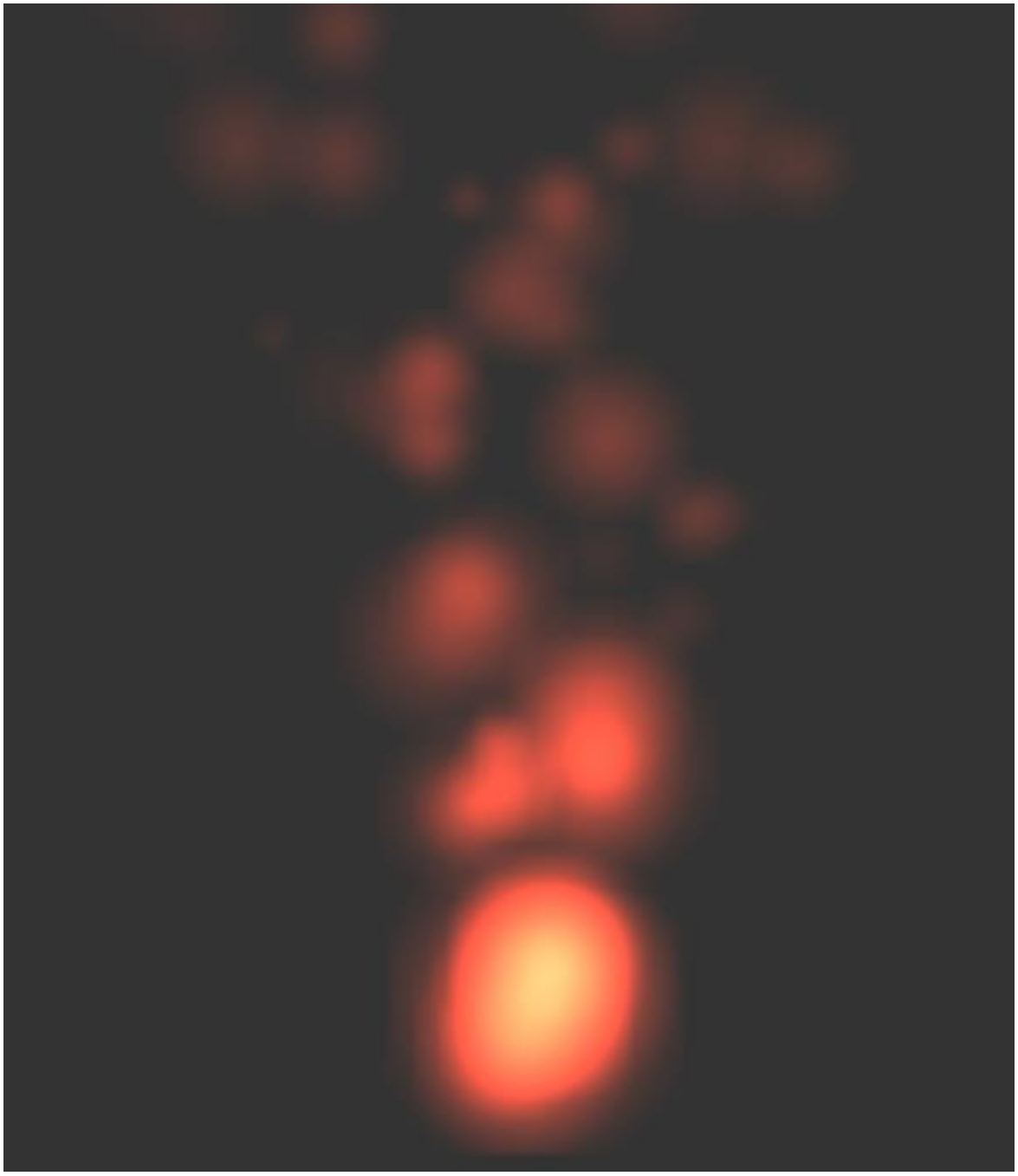

*Image 10 – Model built with 50 paticles*

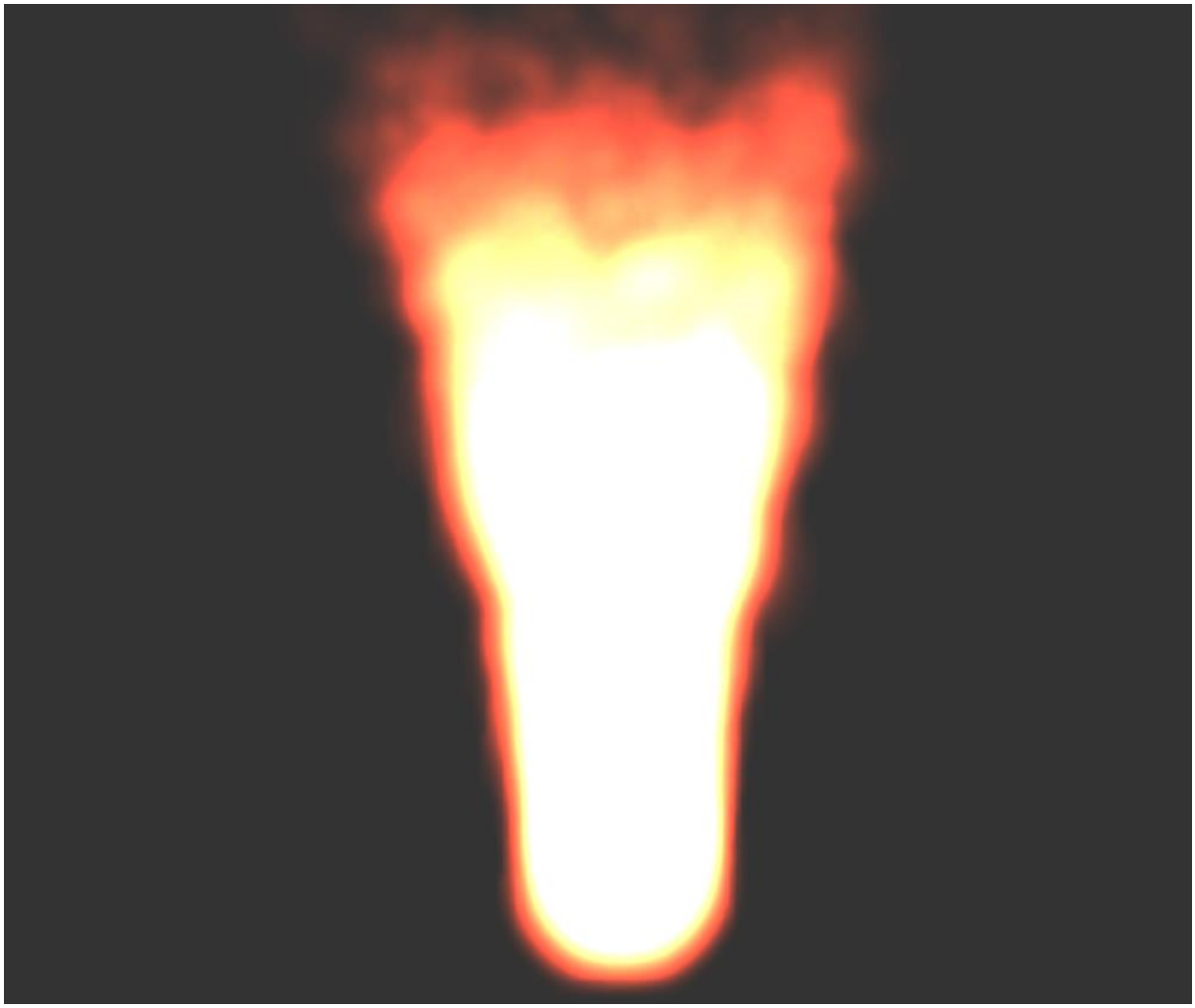

*Image 11 – Model built with 1000 particles*

## 3.2 Forest fire

Forest fire has been modeled using PyGame with forest fire algorithms.

### 3.2.1 First stage

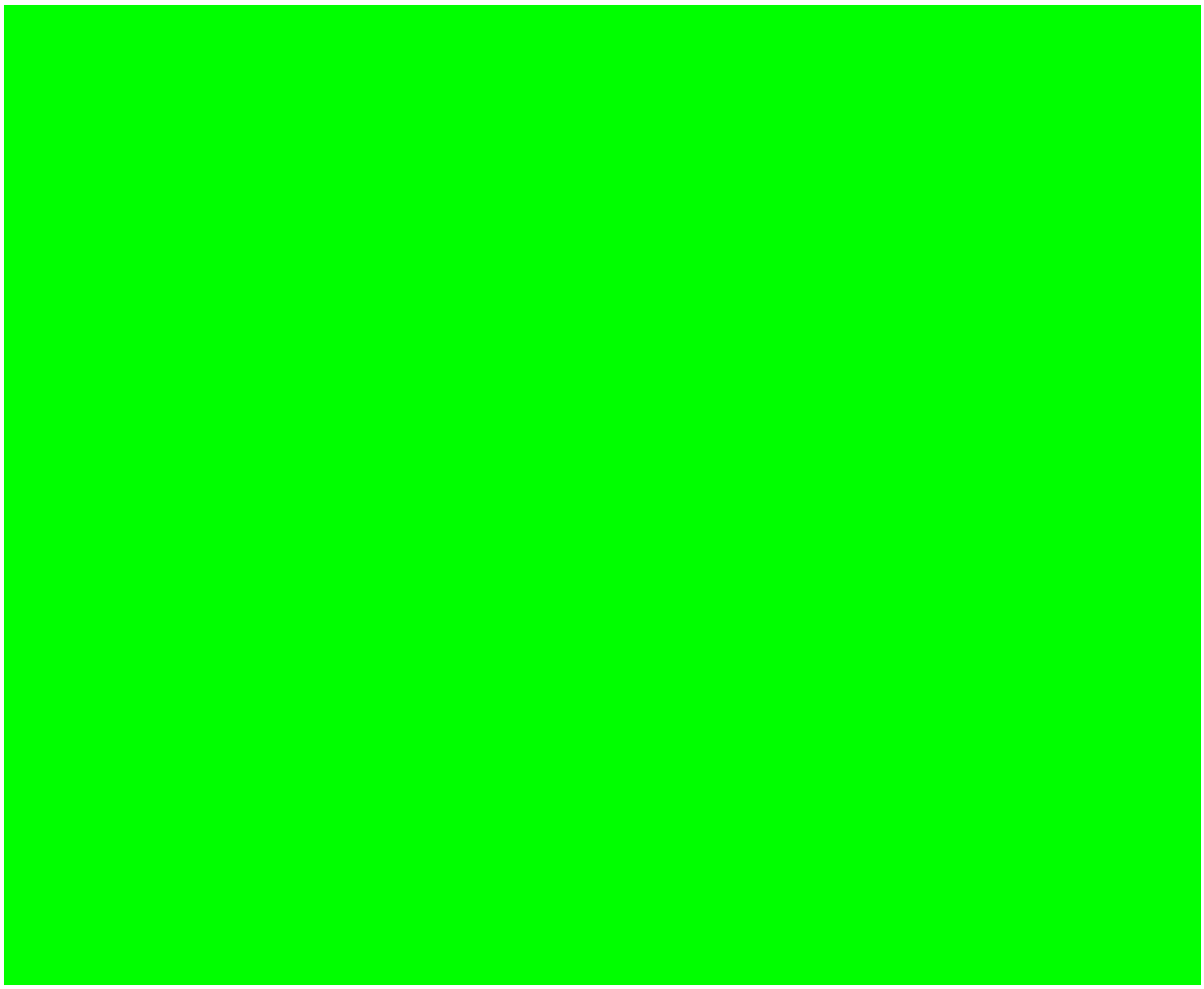

*Image 12 – First Stage of the model full of the trees*

In the first stage the model is full of green cells which means it is full of trees. There are no fire present and shows a normal full forest, before any fire occurs.

### 3.2.2 Second Stage

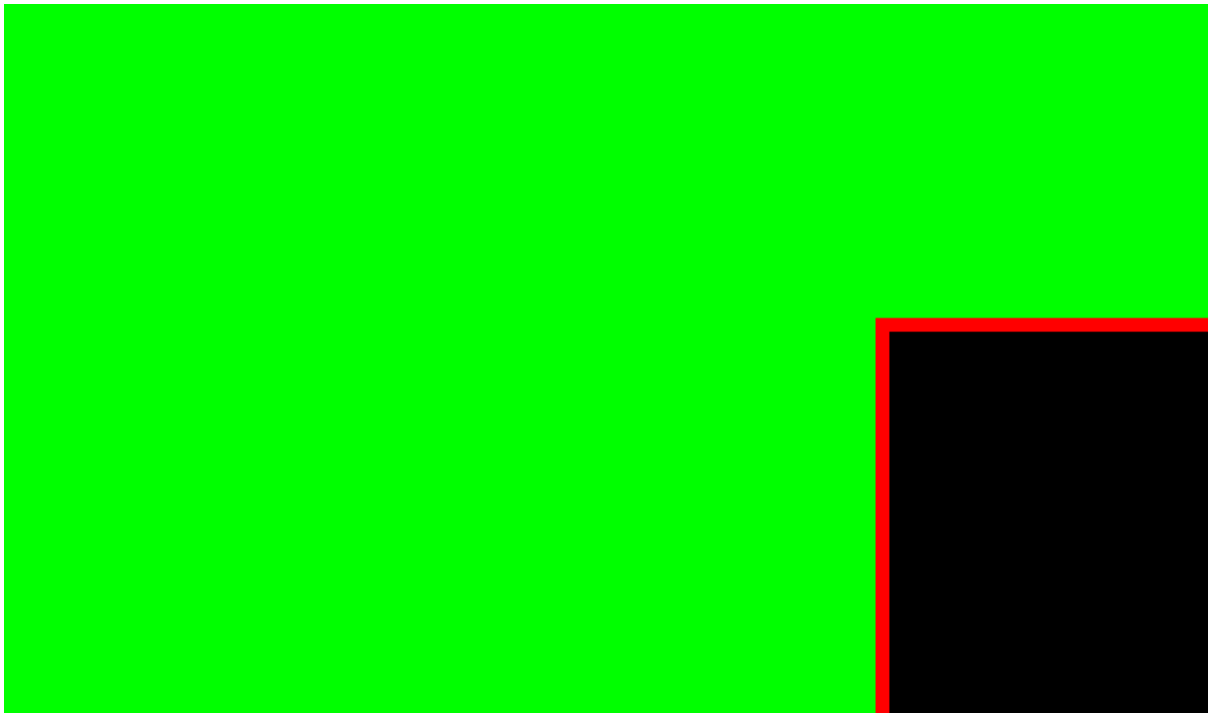

*Image 13 – Second Stage of the model with fire starting*

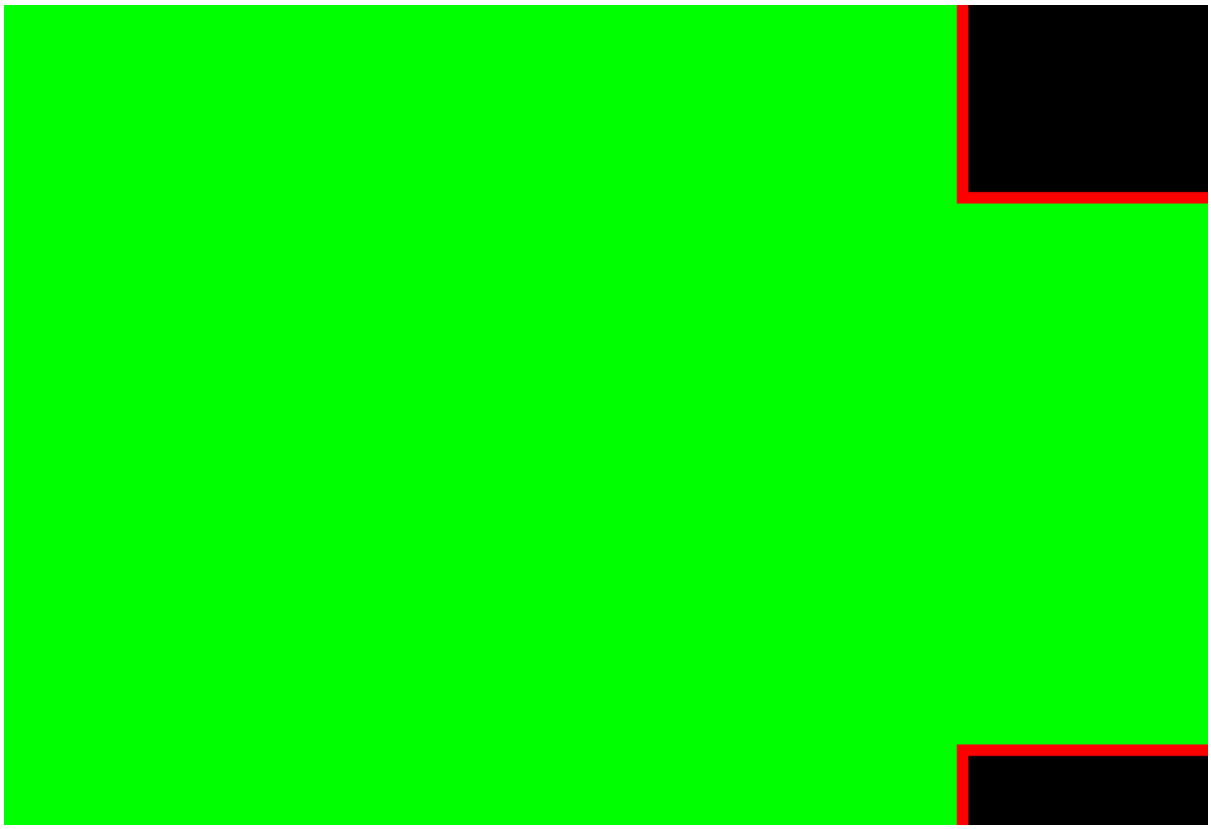

*Image 14 – Second Stage of the model with fire from multiple points*

In the second stage a fire is started with as red cell and spreads around burning down all the trees and turning them in to black cells. These fires are started from random points so there can be more than 1 ignition points.

### 3.2.3 Third Stage

After fire starts to move it burns up half of the screen and continues to burn the forest.

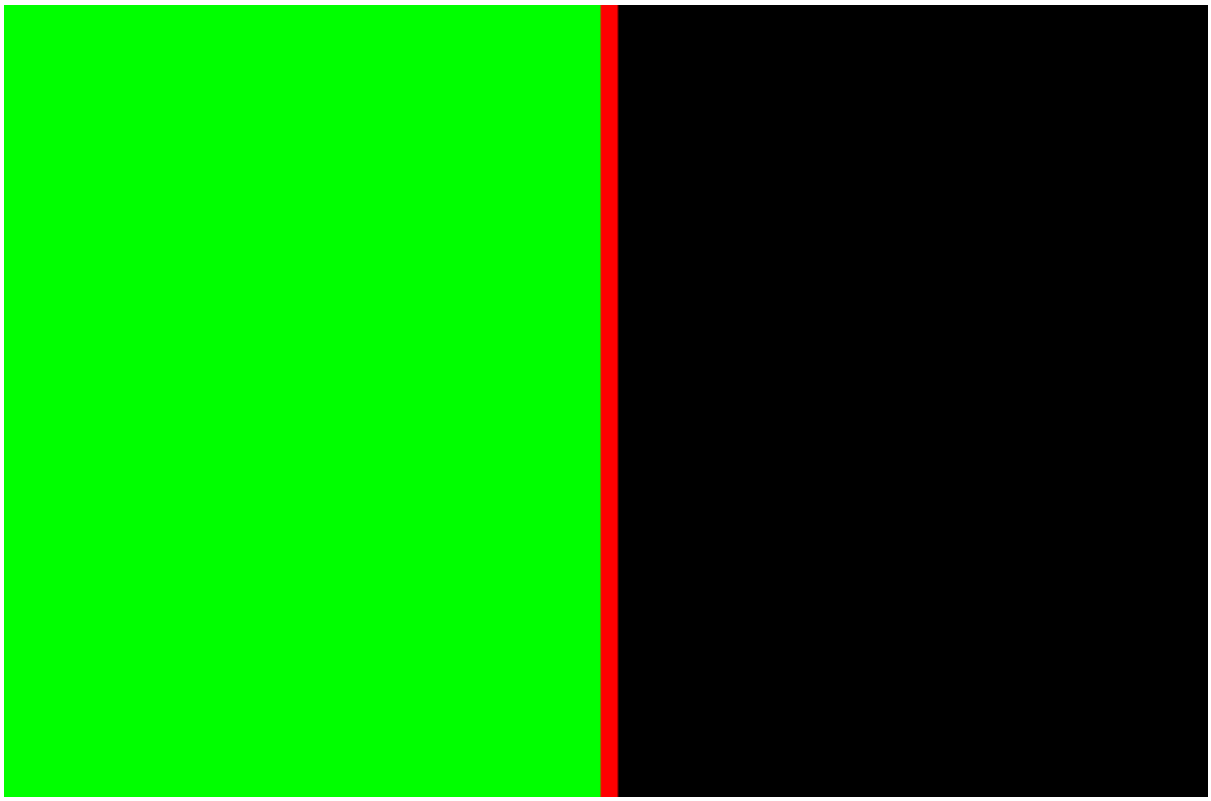

*Image 15 – Third Stage of the model with fire burning half of the forest*

### 3.2.4 Fourth Stage

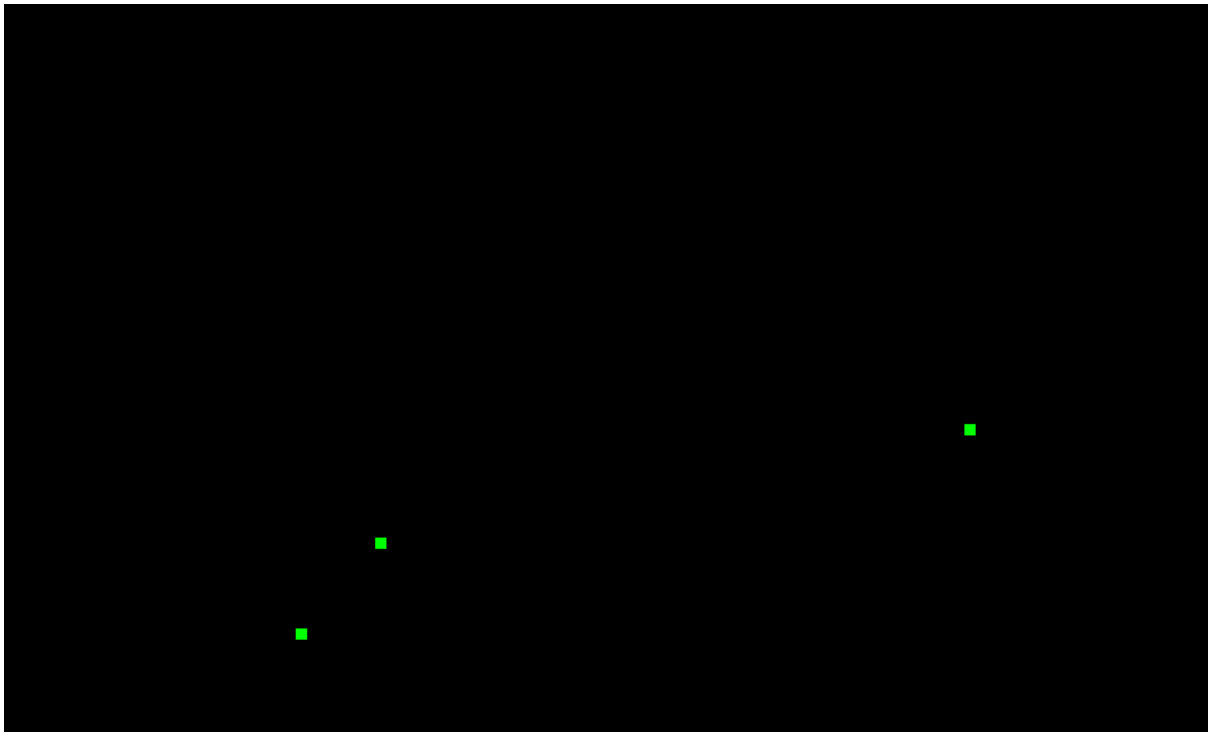

*Image 16 – Fourth stage of the forest fire, where everything is burnt and turned to black*

In the fourth stage fire burnt down the forest completely and left very few trees that survived the fire.

# 4 Discussion

## 4.1 Future Improvements

There are number of improvements that can be made in future simulation of fire in computer graphics. As this is just a simple modelling of fire with particles, if better and high-level computer graphics technology are used, the fires can look more realistic. Also, not every aspect of the fire or the outer environment has been considered when simulating the fire. Only wind and gravity were considered as most important factor to the fire.

## 4.2 Limitations of OpenGL

Although OpenGL is widely used system to simulate computer graphics, it is very old and slow compared to recent technologies that are used. OpenGL is run freely and open and gets updated after long discussions with associated parties. Thus, nowadays it can be considered outdated. OpenGL is easy and simple to use, but it is not the best technology to simulate computer graphics.

To perform better simulation modelling high-level computer graphics programs such as Unity could be used to make fire look more realistic.

## 4.3 Fire

Due to lack of information and knowledge on the natural science of fire, the fire was not simulated perfectly. Many of the algorithms that explains the fire was disregarded when simulating this model. If these there were more knowledge about the fire and these algorithms could have been applied to this simulated model of fire.

## 4.4 Forest Fire

As the original purpose was to simulate the forest fire, simple 2D animation with cells are not very accurate or enough to explain the forest fire clearly. By using this algorithm if a 3D computer graphics were implemented it would have given much clearer and better understanding of this.

Not 3D, but even if there were better graphics presenting the fire, tree and burnt down area, it would have given much clearer understanding of the forest fire. Due to lack of skills the forest fire model was not simulated with clear image. If the skills can be improved and if better technology was used to simulate the same model, much clearer simulation would have been created for normal people to understand with out need for deep understanding.

Also, if this was built in PyGame module, the fires have started randomly. If the fire can be controlled by the users, it would give users more interaction and better understanding of ignition points and fire's movement.

# 5 Conclusion

In conclusion, the computer graphics of fire model has been created well and simulated with particles. It was easy to view the live fire and was controllable with the size, combustible material in the center of the fire, wind affect and rotational view. This fire model can give somewhat understanding to people how the fire actually burns without getting burnt from seeing the real fire. As people interact with the fire and controls the fire, they will be able to get more understanding.

Also, from the forest fire modelling, it can give people knowledge on how fire spreads in forest and how fast they move. People interacting with this model to see how much a single fire destroys the forest can be more aware of forest fires.

# 6 References

[1] OpenGL ES Particle System Tutorial, R.R.Capeda, 2013, Available from: https://www.raywenderlich.com/2704-opengl-es-particle-system-tutorial-part-1-3. Accessed 20th of May 2020.

[2] OpenGL, Available from: https://www.opengl.org/ . Accessed 10th of May 2020.

[3] PyOpenGL – The Python OpenGL Binding. Available from: http://pyopengl.sourceforge.net/ Accessed on 25th of May 2020.

[4] Learn OpenGL, Available from: https://learnopengl.com/In-Practice/2D-Game/Particles . Accessed 27th of May 2020.

[5] Forest Fire Model, Available from: https://en.wikipedia.org/wiki/Forest-fire_model . Accessed 1st of June 2020.

# 7 Self-Assessment

This report is a mediate report that is presented with basic computer graphics. This report does have somewhat computer graphics, but these computer graphics are not presented with high techniques. Generally, the report seems to be well written and readable without much difficulties. Contexts have been well set and separated in multiple parts and not too much texts or difficult wordings have been used. Some computer graphics techniques are discussed in the report including which language and method have been used to demonstrated the simulation model. These includes the tools and libraries such as PyGame or OpenGL Particles are discussed.

Not too many types of images have been presented. There are large number of images in the report, but they are similar and does not have variety of graphics that have been done. They are generally similar types of graphics. These images are still enough to understand the basics of the computer graphics done on fire.

The technology that are used in this project has been compared with other technology explaining the simplicity and efficiency over other technologies. The billboard particle system is used over other simulation methods due to its simplicity and low cost.

The report outlines number of improvements that can be made in the future and if these were applied in current project, better result would have been achieved and more marks would have been awarded. However, this project and report is missing these and not too great.

Overall, this report would be considered to get PASS mark with around +5 to +7 additional marks.