Matrix

By Ophir Havivi and Roni Ben Dom

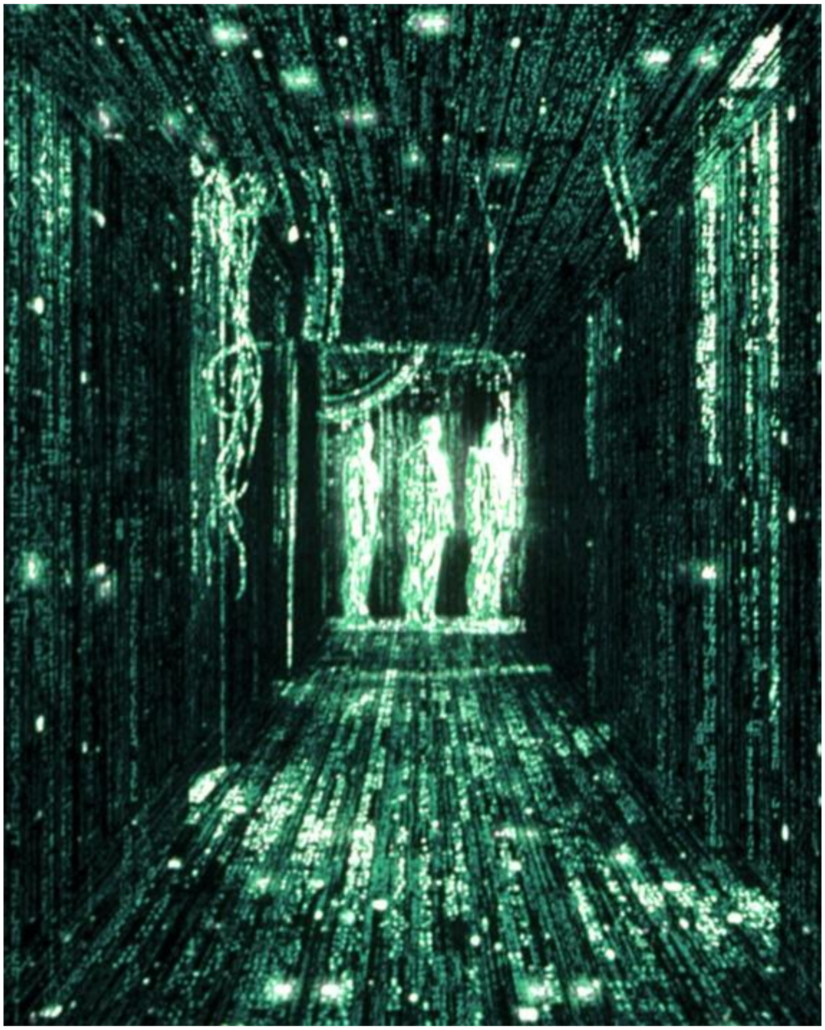
Supervisors: Boaz Sterenfeld and Yaron Honen

Contents

Abstract

The goal of this project was to create a matrix movie like effect over a pre constructed scene.

We used 3D Graphics engine to create a shader to mimic a scene from the famous Matrix movie of the text rain.



The shader was demonstrated using an abstract scene we've constructed for this purpose.

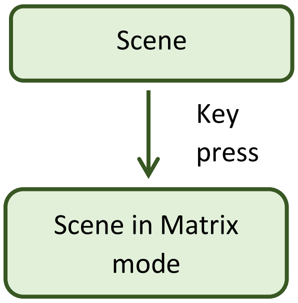
We've searched online for different ways to construct a rain-like shader and a way to apply font graphics onto it. The most substantial information for that came from Graphics oriented forums (i.e Unity's main forum) and shader building tutorials.

The shader we created was applied on the scene we constructed, it yet doesn't work perfectly for every scene – adjustments are needed, hence more development can be applied so it will fit every scene easily.

System description

The project doesn’t require any special gear or hardware, only Unity and Unreal environments.

The demonstrated system – an Unreal scene and the matrix shader applied on it.



Development process

We had two stages while trying to implement the shader, first we tried to work with unity and later on with unreal.

we started by learning the how to operate with unity – how to create objects, apply materials and move around the scene.
After learning the basic operations we started learning about materials and shaders while experimenting, first very basic shaders, only color and some texture.

At this stage, with a basic knowledge, we started on planning towards the project goal – Matrix shader.

Our first try to create a matrix shader was with a very basic shader based on a randomly created letter pattern – from C# code, each letter was made of 4X4 black pixels and a script randomly chooses what pixels to light in green. Afterwards on a second stage the code chose randomly how long and how lighter or darker shall the flow of letters shall be.
Those matrices were moved on the object to create the illusion of raining text.
we made a few input parameters for the shader including speed, and the length sentences.

After having this very basic shader we wanted to apply it on a scene with a few objects, in here we came across an obstacle, we didn't know how to change the shader off all the object all at once - together.
We tried various scripts until we succeeded to apply the shader to all objects at same time.

Moving on, we wanted to have different densities over different objects, which we found out was not possible with the shader we had. We tried to build a new one.
At this stage we had some help – Boaz told us about shader graph (A unity tool) and gave us a Matrix shader from a previous project.
We learned about the shader graph and started trying to improve the shader we received.
The shader is build on 2 kinds of pictures, one with letters and one with tracks the letter will flow in, changing these pictures will change the shader behavior.
Using a script, we created a few from each picture and by that we managed to have different density.
We added outer variables to the shader, enabling us to control the speed and the density, but we steel had one unresolved problem – the flowing direction was no always right.

At this part, we decided to give a try to unreal.

We took the basic idea and concepts we gained from working with Unity and tried to imply them withing Unreal development tools.

We used a premade scene at first and constructed out shader as a port of the shared made in Unity's ShaderGraph. The first draft was promising enough to keep this direction as a viable solution.

We incorporated the use of a model complexity by using the number of vertices in relation to its size in order to get a complexity factor as input to the shader for a more realistic effect.

Complexity derived factor of the shader included the text size and density, text color and speed of flow over the object.

Future work

The shader can be further improved in order to fit more easily on other scenes -