**By Craig McCorrisken**

**Advanced Computer Graphics:**

**Individual Coursework Report**

# Brief

The brief for the individual project was to study and develop my knowledge of a 3D graphics effect of my choice and implement the chosen effect in a working project demo. The 3D graphics effect could be anything, however a list of possible topics was given to help inspire students. These included: Supporting (moving or interactive) multiple light sources and types, Texture Blending, Toon shading (with image processing/procedural geometry silhouetting), Implement a custom shader or GLSL image processor, etc.

# Proposed Solution

* Explain how I plan on solving the given problem
* Insert \*expected\* result images

Out of the list of possible topics available I chose to study and implement texture blending. I chose to study texture blending as it was briefly mentioned during an AGP lab and grabbed my interest enough for me to study more about it. For the texture blending demo, I will draw a box in the centre of the screen. The box will be drawn using a phong shader initially, or by pressing “3” on the keyboard. The box will be drawn using a separate shader which has texture blending implemented by the user pressing “4” on the keyboard. The user will be able to switch between the 2 shaders to see the difference between the effects, this will give the project an interactive element.

# Concept

* Explain the concept of Texture Blending to a non-knowledgeable person (high level, no code/ shader details)
* Insert diagrams for Texture Blending
* Images

# Before & after

* Show box using phong & using texture blending (images, brief description)

# Walk through

* Walk through the steps/ method of creating this project

# Video

* Insert video link here

//remove unnecessary code – FREE MARKS

# Implementation:

Firstly, I created 2 new shaders called “textureBlender.frag” & “textureBlender.vert” inside these files I created code based on the multi-texture section of the OpenGL Programming Guide book by Dave Shreiner which takes two separate textures and takes the position & colour of each texel of the image & blends them together to create a new image.

To allow the two images to blend together I had to implement the SDL\_image library which is an extension to SDL itself allows the loading of non-bitmap texture files. This was implemented to allow the loading of texture files which support transparency, namely .PNG files.

I had to modify the loadBitMap function given to us in the labs to allow the loading of all texture files instead of just being used to load Bitmap files.

I initialised 2x .PNG files and blended them together using my newly created texture blending shader and applied this to a box.

# How to Use:

To switch between the texture blending shader & the phong shader for the box, press 3 on your keyboard for texture blending & 4 for the phong shader.

WASD allows the user to move around the level freely.

# Video:

* Insert link