IB Mathematics AA Higher Level

Exploring and implementing 3D computer graphics using linear algebra

L. Sawadsky

Internal Assessment: Mathematics AA HL

Session: May 2026

Word Count:

Contents

1 Introduction

Representing our three-dimensional world in the two dimensions our computer screens allow us has proven not to be a simple task, but thankfully one that centuries of research in linear algebra happen to assist greatly in. Simulating a three-dimensional environment in a computer system is done largely using research in physics: laws and constants that govern the behaviour of rigid bodies, fluid, and light are replicated in programs to create a world that functions, more or less, like ours. However, there is a problem designers run into that has no parallel in our world. We understand the laws by which physical things change, but *representing* these changes is a challenge unique to computerized versions of the world. My interests in video games, many of which heavily rely on 3D graphics algorithms, and computer science have led me to want to explore the mathematics and computation behind graphical engines. To support this investigation, I will also develop an implementation of the described principles in C++, using the 2D graphics library SDM to create a simple 3D graphics library. This investigation will explore two aspects of 3D graphics: the projection of a set of given vertices in 3D space onto a 2D screen, and some common manipulations of those vertices: translations, stretches, and rotations.

1.1 Context

Modern 3D graphics engines, such as OpenGL, process graphics in a pipeline: every refresh, operations are taken on space and the camera and the display is updated. Each transformation and camera update is processed as a 4x4 transformation matrix (this will be expanded on later in the investigation), and they are then multiplied together into one

2