POLITECHNIKA BIAŁOSTOCKA

WYDZIAŁ INFORMATYKI

PRACA DYPLOMOWA INŻYNIERSKA

TEMAT: SKELETAL ANIMATION USING INVERSE KINEMATICS IN THE UNITY ENGINE

WYKONAWCA: ŁUKASZ BIAŁCZAK

.....

podpis

PROMOTOR: DR INŻ. ADAM BOROWICZ

BIAŁYSTOK 2022 r.

Contents

1. Introduction

1.1 Motivation

Animation is the technique of displaying different positions of a character or object in rapid succession to create the illusion of movement. It is used in various forms of entertainment, such as movies and video games. In the latter, unlike in the former, the animation sequences are performed in real time and therefore impose additional constraints. Without the freedom to process a single frame for minutes or hours during the rendering of the scene, the animator must compromise on the quality and realism of the sequence in order to optimize for gameplay. One such optimization is the use of skeletal animation in which animation sequences are performed by manipulating a tree-like structure of interconnected bones, represented by transforms, to create the desired motion of the character. Furthermore, the interactive nature of video games makes it impossible for the artist to create predefined animation sequences for every possible situation that may occur in the game. As a result, predefined animation sequences are often generic and do not allow the character or object to interact naturally with their surroundings. Game developers have come up with many methods to improve the realism of animation in games such as playing cutscenes for critical interactions between a character and the world. However, this paper will focus on the use of procedural animation and, more specifically, the application of inverse kinematics to skeletal animations in video games.

Inverse kinematics is a technique used in fields such as robotics and computer graphics to determine the joint angles of a kinematic chain that will result in a particular part of the chain, usually an end effector, reaching a specified position in 3D space. In computer graphics specifically, the technique is used to animate the movement of characters and objects such that they interact with their surroundings in a more realistic manner.

There are multiple approaches and algorithms that exist within the inverse kinematics domain, such as analytical methods, gradient descent, and optimization techniques. The choice of approach varies depending on the complexity of the use case, the desired realism of the animation, and system limitations.

1.2 Problem Formulation

The aim of this dissertation is to gain a better understanding of the basic algorithms used in inverse kinematics, discover the built-in functionalities that the Unity engine offers for such implementation. The project implementation will apply these concepts to create pairs of animations which consist of baked and inverse kinematics variants. The use cases will expand the problem by introducing additional constraints which will be required to keep the consistency and realism of the animations. The variations will then be compared through the lens of realism and performance.

The author will begin by discussing the theory of the different approaches and algorithms used to solve the inverse kinematics problem, and the resulting choice of the algorithm to be used in the project implementation. The following sections will explain in depth the implementation of two use cases which demonstrate the purpose of inverse kinematics as a skeletal animation technique. Experiments will then be conducted to compare the inverse kinematics animations with their baked counterparts based on realism and performance. Finally, a summary and conclusion of important points will be presented to the reader.

2. Related Work

2.1 Overview

2.2 Use Cases

2.3 IK Algorithms

2.4 Advantages and Disadvantages

3. Inverse Kinematics in the Unity Engine

3.1 FABRIK implementation

3.2 Spider Movement

3.2.1 Project Setup

3.2.2 Scripts

3.3 Human Animation Sequence

3.3.1 Project Setup

3.3.2 Scripts

4. Experiments

5. Conclusion

Bibliography

[1] Jasvir Nagra, Clark D. Thomborson, and Christian S. Collberg. A functional taxonomy for software watermarking. In *ACSC*, pages 177–186, 2002.

[2] ISO/IEC-9126, International Standard ISO/IEC. In *Information technology: Software product evaluation: Quality characteristics and guidelines for their use*. International Standards Organisation, 1991.

[3] http://www.tex.ac.uk/cgi-bin/texfaq2html?label=multirow.

[4] http://www.texample.net/tikz/examples/computer-science-mindmap/.

[5] http://www.ctan.org/.

[6] H. Partl i inni T. Oetiker. Nie za krótkie wprowadzenie do systemu LaTeX.

[7] Wojciech Myszka. W³¹czanie grafik do tekstów w LaTeX.

[8] R. Kostecki. W miarê krótki i praktyczny kurs LaTeX w π^e minut.

[9] Forum gust. http://www.gust.org.pl/.

[10] Wykresy - TikZ & PGF Manual for Version 2.00.