Kali Escrima Arnis Trainer

an Animated Web Application teaching Traditional Filipino Martial Arts

Isabella Dahilig ’18
Senior Advisor: Professor Yoon

Introduction

During the late fourteenth century, the Spanish Conquistadors invaded the Philippines and threatened filipino culture. The practice of filipino martial arts, known as Kali Eskrima Arnis, was forbidden by the Spaniards in order to thwart potential rebellions.[1] Despite the Spaniards' military forces, filipino people disguised their martial arts in various ways. They masked their deadly techniques as dance rituals filled with fluid movements and lively performances played along with filipino percussion music.

Kali Eskrima Arnis is an indigenous Filipino martial art, formed from self-defense and passed on from generation to generation; this martial arts specializes in weaponry combat, empty-hand fighting, and grappling.[2] Kali is resourceful, in which self-defense techniques utilize the entire body for efficiency and energy.

Access to learning traditional filipino martial arts is limited. Due to endless occupation in Philippines, from Spain, America, and Japan, some of the oldest traditions, including martial arts, were kept secretive based on tradition. Filipino citizens were forced to reform their traditions with the Spanish, and many forsook culture by the appeal of “more modern” culture during the American occupation. Many americanized versions of this martial art exist from cheap entrepreneurship and media entertainment, so finding an authentic master of the martial art is rare. From all of these difficulties hindering people to learn the martial art, Kali Eskrima Arnis is a dying culture.

Introduction

My goal is to create an authentic Kali Eskrima Arnis education through a web application of animated tutorials and training sessions for each technique.

- For the scope of my senior project, only one technique known as the “knife generator” will be implemented in this web application. The “knife generator” is practiced by two people, each with a knife, engaging in a continuous of attacks and blocks.
- For easy accessibility regardless of location, the educational experience will be implemented as a web application, with WebGL as the framework to animate the user and the instructor/training partner.

Technologies and Platforms

- Blender, ver. 2.97 is an open-source 3D computer graphics software, utilized for implementing a skeleton rigging within an outsourced Blender model.
- An additional open-source exporter is required to export Blender model as a JSON object, which captures
- WebGL is a JavaScript API, and operates to render 3D graphics within a compatible web browser (i.e, Safari, FireFox, Chrome). WebGL is programmed in JavaScript and displayed as a <canvas> element in HTML.
- Three.js is a lightweight library, responsible for intricate 3D animation and functions by loading a scene full of objects in HTML; once the canvas loads to completion, the canvas begins to redraw the scene, which simulates movement. The entire three.js library was downloaded, and additional components can be imported as needed.

Integral Programming Components

Blender

Within Blender, an open-sourced Blender model of a human male was imported; 51 bones were placed inside the model and parented to the vertices of the blender model. Thus, when the bones were transformed so were the associated vertices of the model. Determining the weight, or which vertices were transformed for each bone, was manually configured.

Three.js

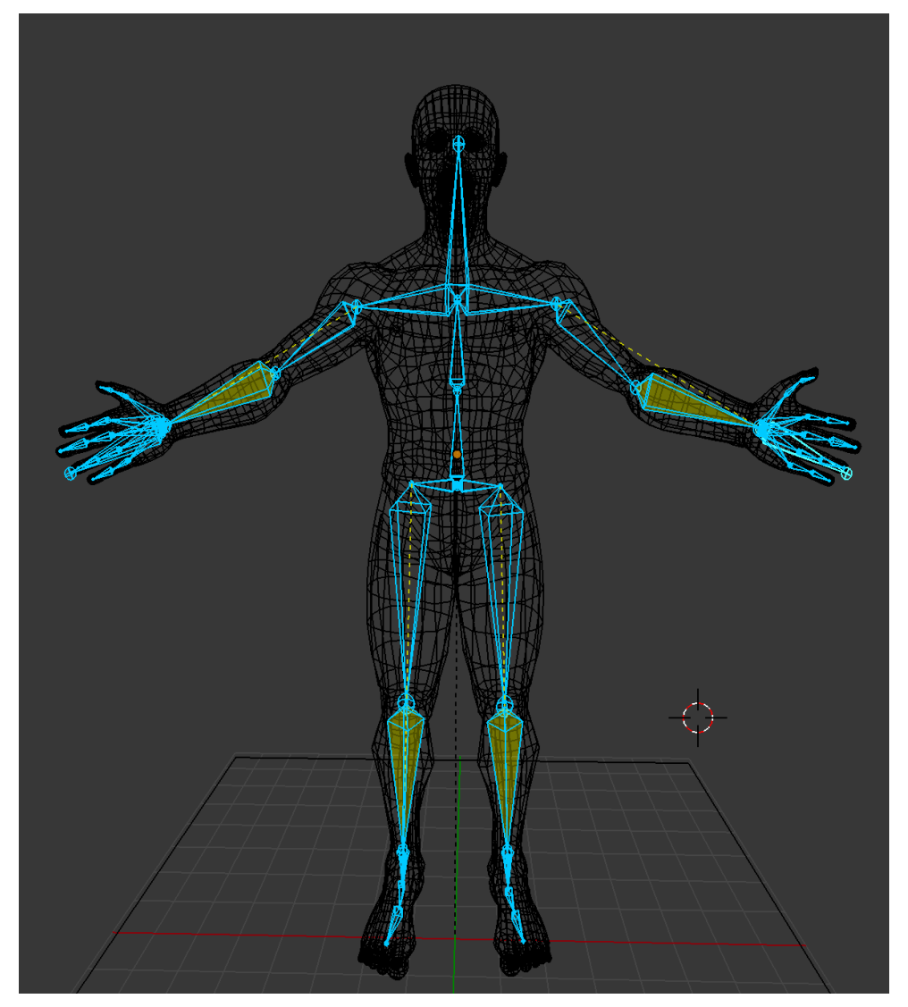
The rigged models and open-sourced knife JSON objects were imported with a loader, and within the animate loop will add the models to the scene when they finish instantiating as SkinnedMesh objects and Mesh objects respectively with imported geometries and manually created materials. Light objects were imported to the scene as required by the type of materials wrapping around the objects. For testing purposes, an Orbit Control functionality was imported, so the user can zoom, pan, or orbit the scene.

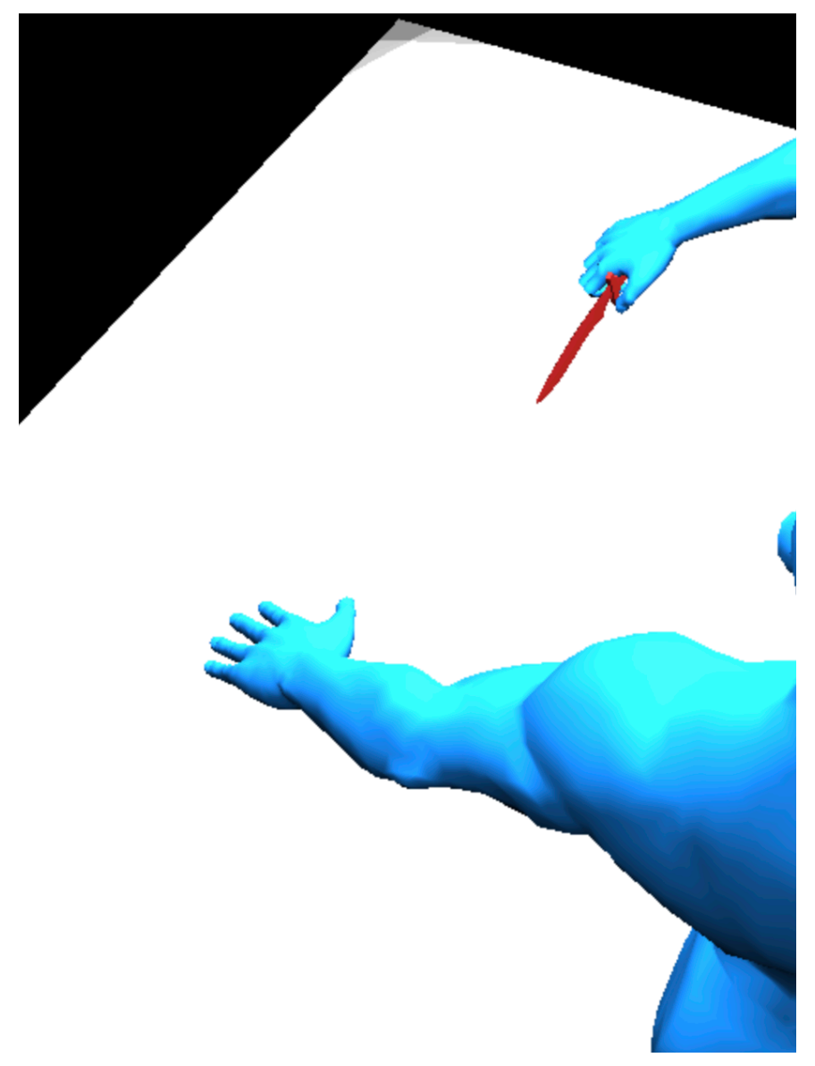
Once the basic environment was established, the models and knives were position to simulate the resting positions of trainers engaging in the “knife generator” technique. A knife was parented to the right forearm of each model, thus when the the forearm or any of its parents were manipulated, so was the knife. Animation was simulated by incrementing bone manipulations. In order to mimic physical interaction between models within the scene, multiple collision detections were implemented to trigger specified movements. In Three.js, collision detection works by instantiating a three-dimensional rectangle around the lesser-moving object and a ray at the most extended point of the moving object. Through this method, collision detection checks for the distance between the origin of the ray and the rectangle is nonnegative.

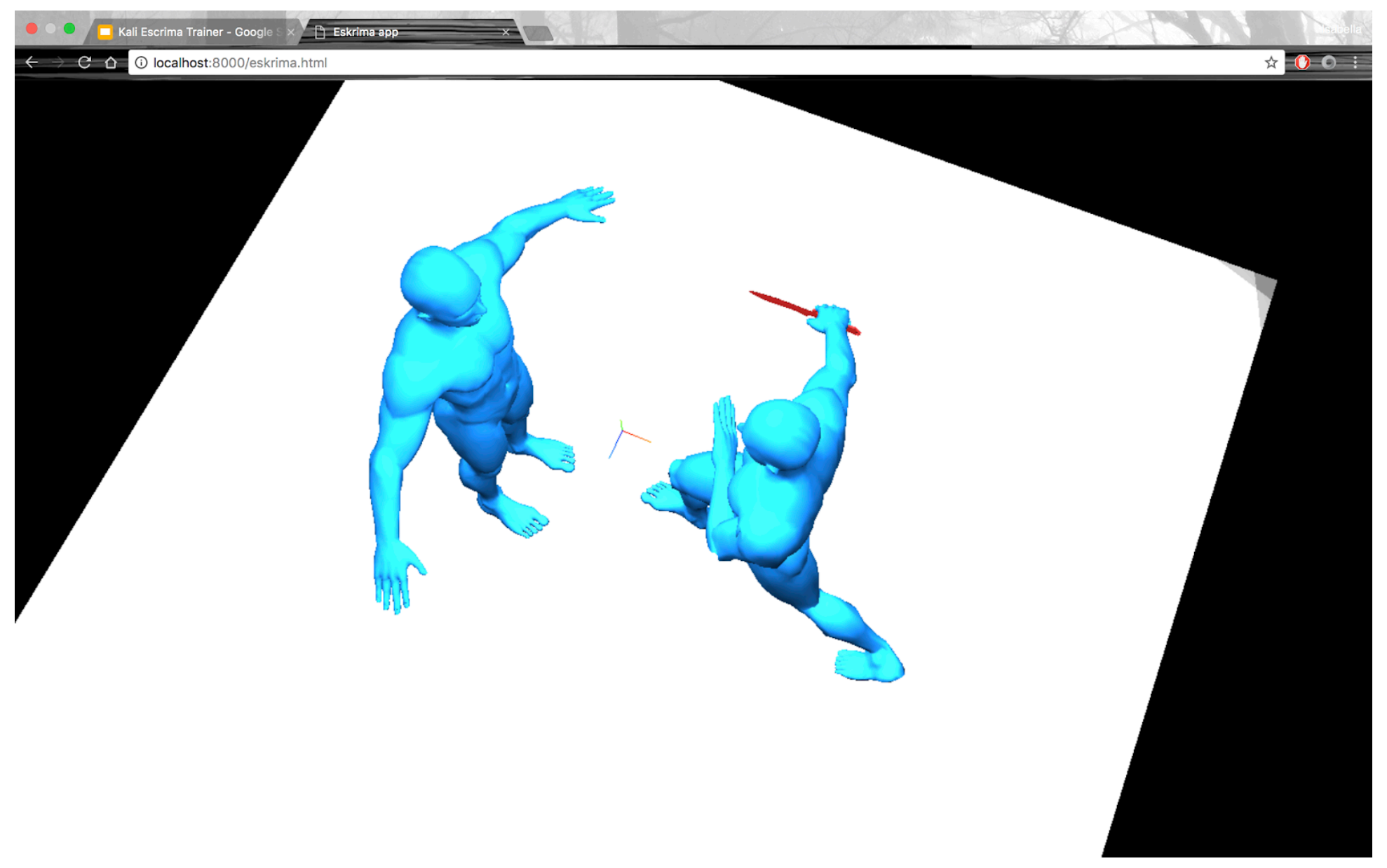
References

1. “A Brief History of Martial Arts, Eskrima and Balintawak Eskrima.” *Balintawak Eskrima Self Defense System*, WordPress, 19 Feb. 2011, apobalintawak.wordpress.com/history/ the-development-of-balintawak-and-eskrima/.
2. “Bounding Volume Collision Detection with THREE.js.” *Mozilla Developer Network*, Mozilla and Individual Contributors, developer.mozilla.org/en-US/docs/Games/Techniques/3D_collision_detection/ Bounding_volume_collision_detection_with_THREE.js.

Results







Discussion and Conclusions

At the end of the spring term, I plan to have a fully developed, interactive web application that will educate users on Kali Eskrima Arnis. When the application begins, the user will stand in front of the Xbox Kinect, and the user can watch it’s animated self on the screen. The user will then begin by choosing to learn the “knife generator” technique, and first choose either to be the attacker or the defender. Each role will have two tutorials, a step by step instruction of the technique itself and another set of steps for the knife disarm at each step. These tutorials will provide feedback on the user’s movement. After completing the tutorials, the user can choose the simulation, in which the user can practice the technique with a programmed model continuously performing moves, and the user can attempt disarms.

Acknowledgments

The presenters would like to thank Professor Yoon and the Computer Science Department for guidance.