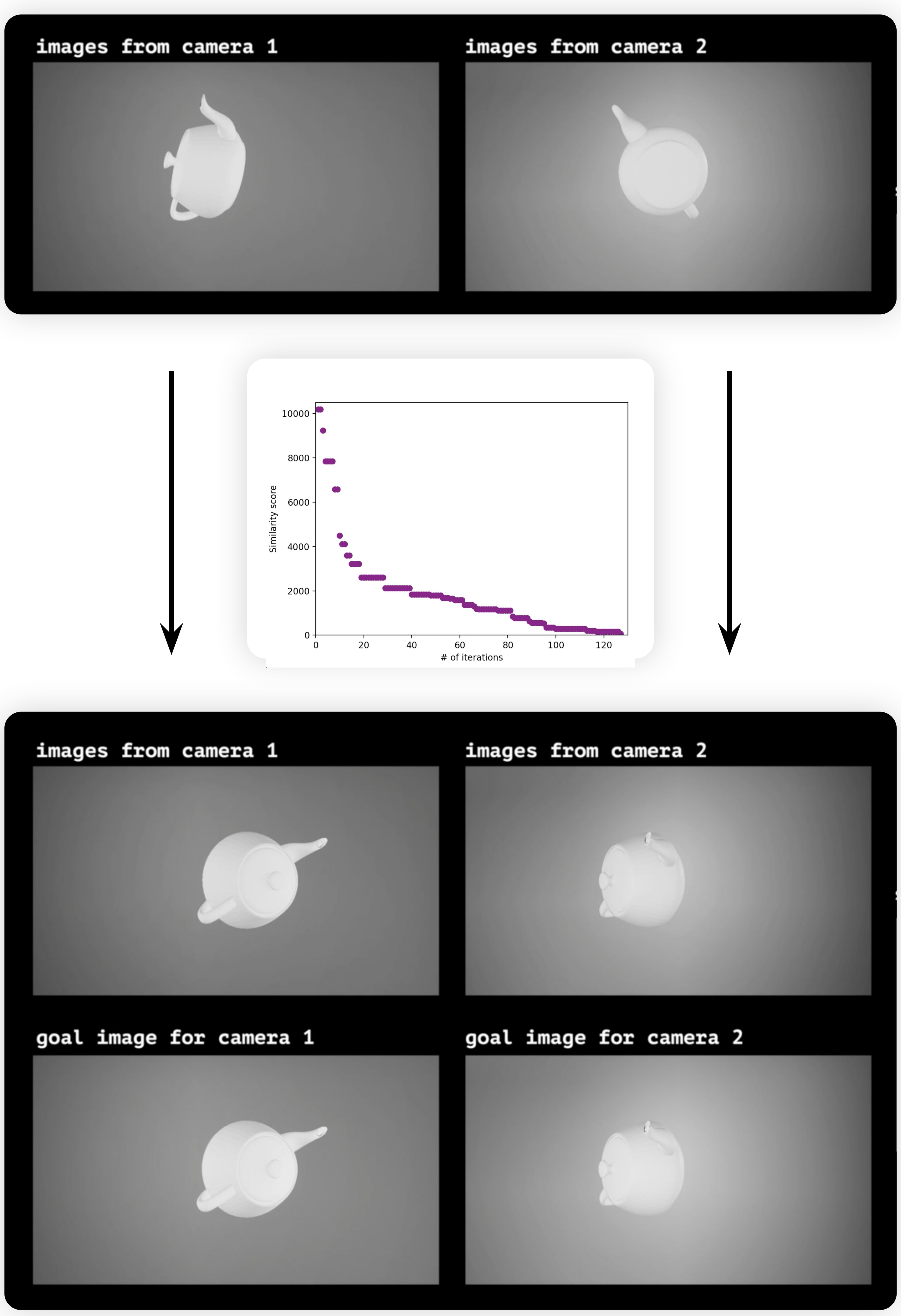
We can use a **3D video game engine** to accurately estimate the orientations of **3D objects.**

Nathan Hausspiegel, Physics Department, Wesleyan University, Middletown, CT

Introduction

- Our lab records videos of 3D-printed particles moving in a high-volume tank of water with four high-speed cameras
- We examine their motion by determining the position and rotation of these particles from the videos
- Our method for finding particle orientation from video footage involves rendering images of a 3D model of the particle from the perspectives of the four cameras and using a search algorithm to rotate and translate the model until it matches the video footage
- We currently use MATLAB for searching and image rendering; however, image rendering in MATLAB is slow and lacks detail, and does not take the lens distortion of the real-world cameras into account

Diagram below: The teapot starts at an incorrect orientation and, over many iterations, the algorithm changes its orientation until it matches the goal images. The graph shows the similarity score for each iteration of the algorithm.



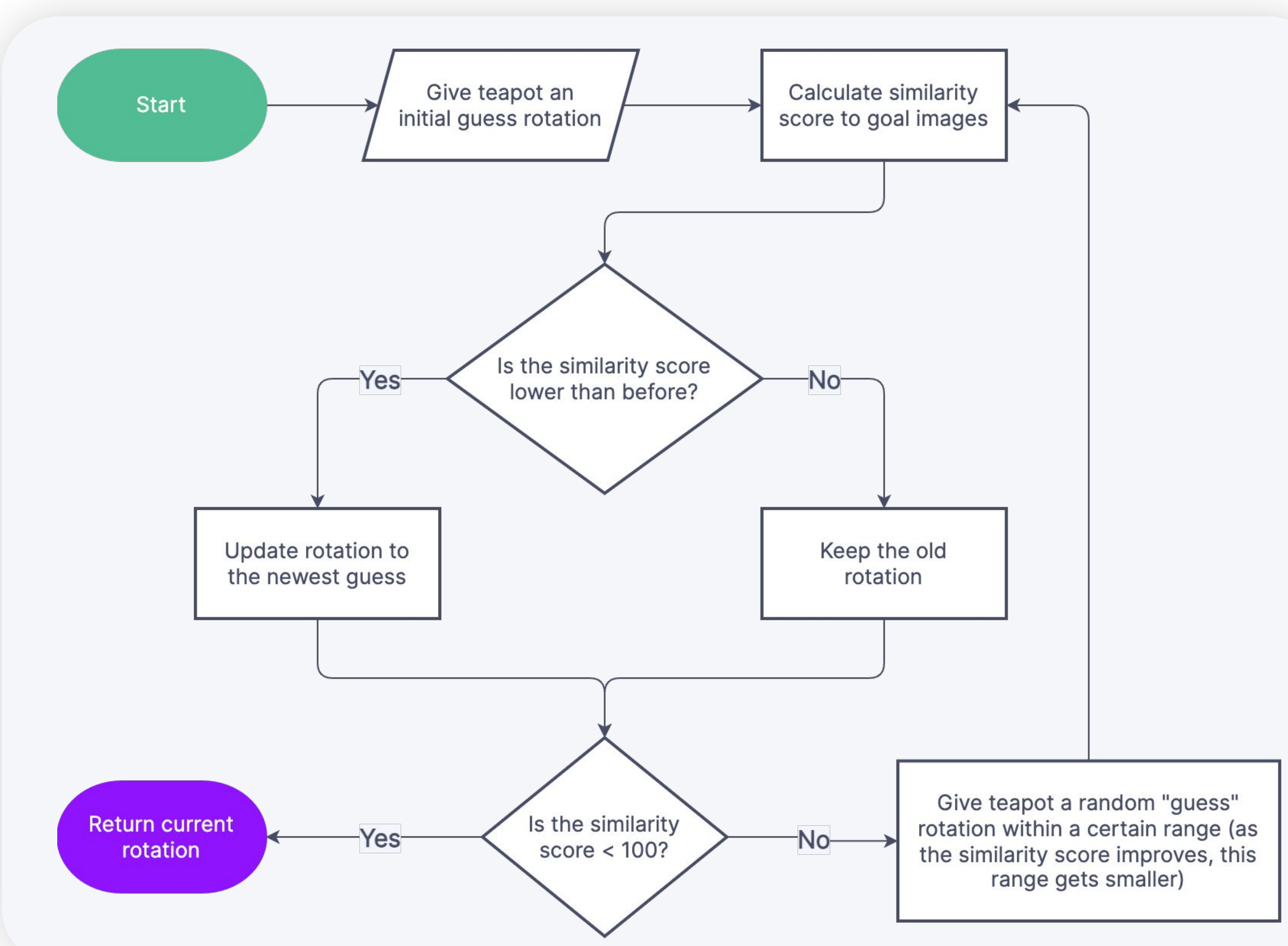
Objectives

- Is it possible to implement this method within a 3D video game engine?
- Would this engine be able to render images with high detail and accurate lens distortion?
- Would this engine be able to replicate real world lighting accurately?
- Is this new method faster than the MATLAB implementation?

Methods

- We selected Unreal Engine 5 for its capacity for highly detailed imagery and its tools for applying real-world lens distortion to in-game cameras
- Python was used to interact with and script the in-game environment
- We used a “teapot” 3D model, only two cameras, and pre-rendered goal images as a dummy setup for implementing the search algorithm
- The algorithm attempts to minimize the residual sum of squares between the goal images and the model images

Diagram below: A flowchart describing a simplified version of the orientation search algorithm. The threshold of < 100 used to end the process is arbitrary, and describes how similar the model images need to be to the goal images to end the algorithm.



Conclusions

This new method was effective in a test case with 3D-rendered goal images and a simplified environment. Further research is needed to improve the speed and real-world accuracy of this method, and to determine whether or not it is suitable for long-term use in analyzing lab data.

Acknowledgements: I would like to thank my professor Dr. Greg Voth for mentoring and guiding my research throughout this project, as well as the other members of the Voth lab for their help. I would also like to thank the QAC for accepting me into the Summer 2022 apprenticeship program, and for their support and funding for this project.