FALL TEST PLAN

Sputnik

Driving Photo-Realistic Avatars using Drone

Team G

Annanya
Andrew Zhang
Ming Feng Li
Naman Menezes
Saksham Kukreja

Industrial Sponsor
Meta

18th September 2024

Carnegie Mellon University
THE ROBOTICS INSTITUTE
Master of Science
Robotic Systems Development

Table of Contents

1. Introduction

This document outlines the series of tests by Team G - Sputnik during the Fall 2024 semester. The primary objective of these tests is to validate that the system requirements established during the Spring 2024 and Fall 2024 semesters are successfully fulfilled at various stages of the sprint cycles. Team G - Sputnik will communicate the outcomes of these assessments through progress reviews and the Fall Validation Demonstration. By the time of the Fall Validation Demonstration, we aim to have a fully integrated autonomous human-following drone for avatar driving, seamlessly integrated with Meta Quest 3, meeting all the specified requirements detailed in Appendix A.

2. Logistics

The equipment and test sites required for the project are enumerated below.

- **DJI Mavic 3 Enterprise:** The hardware platform which will be used for human tracking and image capture.".
- **Drone Cage at AI MakerSpace:** This drone cage serves as the main testing site for our subsystems and the Spring Validation Demonstration. We can utilize this facility for scenarios involving hovering drones.
- **Drone Cage at Scaife Hall:** The drone cage is proposed to be a good test site for our Fall Validation Demonstration and to test scenarios involving motion of the human
- **MRSD Lab:** Located on the B-level of Newell Simon Hall, this facility is designated for unit and subsystem bench tests.
- **Onboard SDK:** This software development kit facilitates the streaming of data from the drone to our computational system.
- **Lambda Machine:** This system is utilized for intensive computations required for generating avatar models.
- **Raspberry Pi Zero:** Serving as the onboard central compute platform for the drone.
- **Meta Quest 3:** Used for visualizing the test results.
- **Meta Quest Pro:** This device provides improved visualization of the subject's face in cases where camera images may not yield satisfactory results.
- **Wifi Module:** Enhances communication speed between the drone and the system.

3. Schedule

Identifier	**Capabilities**	**Test(s)**	**Requirement(s)**
Progress Review 8 (25 September)	- Test Renderer Performance - Test Drone Path Planner - Test VR Headset Communicaton with the Windows Desktop	- T6 - T1 - T7	- M.P. 6 - M.P. 8 - M.P. 7
Progress Review 9 (9 October)	- Testing Decoder Inference Speed - Test Encoder Performance	- T5 - T2 - T3	- M.P. 1 - M.P.4
Progress Review 10 (30 October)	- Testing Decoder Joint Error Performance - Testing VR Headset Real Time Rendering	- T4 - T8	- M.P. 2 - M.P. 5
Progress Review 11 (13 November)	- Full software pipeline integration test	- T9	- M.P. 1 - M.P. 2 - M.P. 4 - M.P. 5
Progress Review 12 FVD (18 November)	- Reconstruct Avatars with Unseen Motion PSNR and Novel View PSNR $\geq$ 28 dB - Average RMSE of avatar over 100 frames < 100 cm^2 and FPS should be at least 36 - Percentage of frames capturing the Human should be $\geq$ 95%	- T10	All M.P.s

4. Tests

Table 1: Human Following Path Planner

Test No. 1: Human Following Path Planner	
Objective	Validate Human Detection Code Module
Elements	Perception Subsystem
Location	AI Maker Space
Equipment	1. Laptop 2. Drone 3. Lamda Machine
Personnel	Annanya
Procedure	1. Run the drone autonomously in front of the human 2. Track human with a radius of 2m to 2.5m
Verification Criteria	1. The human should be in 90% of the frames 2. There should be no collision with the walls and the human 3. The drone must always be in the range of 2m to 2.5m away from the human

Table 2: Encoder Pose Range Test

Test No. 2: Encoder Pose range test	
Objective	Validate the range of motions possible with the new dataset
Elements	Perception Subsystem
Location	AI Maker Space
Equipment	1. Lambda Machine 2. Drone SDK 3. Personal Laptop
Personnel	Saksham
Procedure	1. Start the ROS nodes on the Lambda Machine. 2. Establish communication between the drone and the Lambda Machine via ROS topics. 3. Observe the Motions possible with the new dataset and the model.
Verification Criteria	1. Reliable camera data transmission and commands between the drone and Lambda Machine. 2. The avatar performs the range of motions exactly as the human performs.

Table 3: Qualitative performance of the Encoder through tSNE plots

Test No. 3: Qualitative performance of the Encoder through tSNE plots	
Objective	Clear distinction observed in the latent space of the encoder for varied poses
Elements	Perception Subsystem
Location	AI Maker Space
Equipment	1. Drone 2. Lambda Machine 3. Laptop
Personnel	Saksham
Procedure	1. Integrate 3D pose estimation with the encoder model. 2. Run the code with a pre-recorded drone camera's video through the ROS node and record the latent vector of the avatar model. 3. Compress the recorded latent vector to a 2D space using tSNE plots.
Verification Criteria	1. The number of points should be equal to or greater than 90% of the video frames verifying the human detection model. 2. A distinct boundary should be visible after plotting the latent space results.

Table 4: Decoder Output Joint Error Test

Test No. 4: Decoder Output Joint Error Test	
Objective	Ensure the Pose of Images Produced by the Decoder matches with the Input Poses
Elements	Perception Subsystem
Location	AI Maker Space
Equipment	1. Lambda Machine 2. Windows Machine 3. Human Target 4. DJI Mavic 3
Personnel	Naman
Procedure	1. Obtain 2 minutes of video using the drone while following the human 2. Run the images through the 3D Pose estimator and encoder 3. Run the encoded images through the decoder 4. Run the decoded images into the renderer and create a set of decoded images 5. For each frame compute the error with the joints between the input pose and the output pose 6. Compute the average of all the joint errors over 100 frames
Verification Criteria	1. The Average Root Mean Square Error (RMSE) over a set of 100 frames must be below 20 cm

Table 5: Decoder Inference Speed

Test No. 5: Decoder Inference Speed	
Objective	Verify that decoder inference time is within rendering bounds
Elements	Decoder Subsystem
Location	AI Maker Sapce
Equipment	1. Lambda Machine 2. Windows Machine 3. Dataset of Encoded Inputs
Personnel	Naman
Procedure	1. Prepare a dataset containing the encoded inputs 2. Using a ROS2 Publisher, publish the encoded inputs onto the ROS2 Network 3. Read the encoded inputs in the windows machine and run the decoder
Verification Criteria	1. The average time taken per frame by the decoder over 100 frames of video must be less than one second per frame

Table 6: Avatar Rendering Test

Test No. 6: Avatar Rendering Test	
Objective	Validate the rendering quality and speed
Elements	Avatar Rendering Subsystem
Location	AI Maker Space
Equipment	1. Lambda Machine 2. Windows Machine 3. Meta Quest 3
Personnel	Justin
Procedure	1. Take the deformed 3D avatar as a mesh with 3D vertices and texture. 2. Transform the mesh data format to PyTorch tensors for rendering. 3. Given query camera poses, the module renders images from query viewpoints. 4. Pass the rendering images to the VR headset.
Verification Criteria	1. The system could render accurate RGB images according to the 3D avatar generated by the decoder and the query camera poses. 2. The rendering speed in terms of FPS could achieve the system requirement.

Table 7: VR Headset Pose Capturing

Test No. 7: VR Headset Pose Capturing	
Objective	Validate the pose capturing from the VR headset.
Elements	VR Headset Subsystem
Location	AI Maker Space
Equipment	1. Windows Lambda Machine 2. Meta Quest 3
Personnel	Andrew
Procedure	1. Install Quest Link Software on Windows Desktop 2. Connect both Desktop and VR Headset to the router. 1. Establish a Quest Link connection and run a camera pose tracking script. 3. Print/Visualize the campos on desktop.
Verification Criteria	1. The VR Headset can establish a stable connection with the Lambda Machine 2. The VR Headset can send the camera pose for both eyes in with $\geq 30\text{FPS}$

Table 8: Avatar Visualization in Headset

Test No. 8: Avatar Visualization in Headset	
Objective	Validate Integration of sub-systems
Elements	Perception and Avatar Subsystem
Location	AI Maker Space
Equipment	1. The Windows Desktop 2. The Meta Quest 3
Personnel	Andrew
Procedure	1. Generate geometry and texture from the Decoder. 2. Render the left and right images based on the camera pose. 3. Visualize the scene on the Virtual Reality Headset by streaming the images using Quest Link.
Verification Criteria	1. The renderer can render correct images of the avatar based on the input camera poses. 2. The generated images can be displayed in the VR Headset in the correct position.

Table 9: Subsystems Integration Test

Test No. 9: Subsystems Integration Test	
Objective	Validate integrated system
Elements	Perception Subsystem, Renderer Subsystem, VR Headset Subsystem, Integration
Location	AI Maker Space
Equipment	1. Lambda System 2. Windows System 3. Wifi Router 4. Meta Quest 3
Personnel	All Team
Procedure	1. Run the fully integrated script by streaming a pre-recorded video to the pipeline 2. Visualize the driving avatar on the VR Headset
Verification Criteria	1. The code runs without any errors 2. Visualization produces desirable results 3. All the promised functional requirements are met including quality and speed

Table 10: FVD

Test No. 10: FVD	
Objective	Demonstrate fully functional system
Elements	Integrated System
Location	AI Maker Space
Equipment	1. Lambda System 2. DJI Mavic 3 3. Windows System 4. Wifi Router 5. Meta Quest 3
Personnel	All team
Procedure	1. Ready the Software and Hardware 2. Start the Integrated script to launch ROS nodes 3. The drone takes off facing the single human target 4. The human walks around freely in an obstacle free area 5. The data is processed by the system and the results are sent to the headset 6. The viewer watches the rendered avatar in the VR Headset
Verification Criteria	1. The drone follows the human at a safe distance according to our performance requirements 2. The avatar is rendered in the Virtual Reality headset according to our performance requirements

A. Appendix

Table 11: Mandatory Functional Requirements and Performance Metrics

ID	Functional Requirement	Performance Metric
M.P.1	Reconstruct Avatars	The reconstruction frequency should be ≥ 30 FPS.
M.P.2	Control Avatar	The Average Root Mean Square Error (RMSE) over a set of 100 frames must be below 20
M.P.3	Detect and Track the Human	Percentage of frames capturing the Human should be $\geq 90\%$.
M.P.4	Estimate Human 3D Poses	Mean Average Precision (mAP) $\geq 80\%$.
M.P.5	Visualization on VR Headset	Render two view visualization of the avatar given camera pose of both eyes with an FPS ≥ 30.
M.P.6	Plan Trajectories for Drone	The average distance between the drone with ego human and obstacle should be between $3\,\mathrm{m}$ to $5\,\mathrm{m}$ and $0.6\,\mathrm{m}$ to $0.8\,\mathrm{m}$ respectively.
M.P.7	Control Drone	MSE between drone's actual positions and the predefined waypoints $\leq 25\,\mathrm{cm}^2$.
M.P.8	Detect Obstacles	False Negative Rate (FNR) $\leq 8\%$.