Abstract
There is a need for a small, lightweight, portable aerial surveillance system that will fill the gap created by current technologies. Existing technologies are either cost prohibitive or require advanced user control in the case of large unmanned aerial vehicles (UAVs), or are cumbersome or require trained users, in the case of small UAVs. The Portable Aerial Layered-Sensing Camera (PALOC), as the project will herein be called, is an attempt to fill that specific void. The device will consist of two parts, a reconnaissance module (Surveillance Sphere) and a base station (Command Center). The reconnaissance module will be launched into the air, deploy a parachute, and stream live video to the base station until its descent is complete. The Command Center will serve to capture all streamed data to display to the end user, will allow the operator to manipulate the video feed, and will act as an interface to the Surveillance Sphere, allowing the operator to view live information about the status of the descent.

Hardware Theory of Operation
The Surveillance Sphere is designed to be a fully autonomous device that once connected to the Command Center via a Wi-Fi connection, given the initialize signal, and launched into the air will handle all necessary functions.

To begin the sequence, the FPGA Auxiliary Board will receive an initialize signal from the Command Center and will turn on the FPGA Board and, after an appropriate amount of time, will send it the initialize signal so that it can initialize all necessary components, including all 6 cameras, and begin monitoring the accelerometer. After initialization, but before reaching the apex, the cameras will be sending data to the FPGA in order to ensure that the cameras are functioning properly; however, the FPGA will ignore the data. The accelerometer will trigger once the Surveillance Sphere has reached the apex, deploying the parachute and causing the FPGA to begin stitching the camera images. The stitched image will be passed to the FPGA Auxiliary Board to be compressed prior to being sent over the Wi-Fi interface to the Command Center.

The process of capturing the images, stitching, compressing, and transmitting the video feed will be performed continuously until the Surveillance Sphere reaches the nadir. At that point, the accelerometer will trigger again, causing the FPGA Board to send a signal to the FPGA Auxiliary Board which will turn off all other modules in the Surveillance Sphere. While in this low power state, only the FPGA Auxiliary Board will be powered on.

A battery pack will be used to power the Surveillance Sphere and will be charged using an accessible port on the Surveillance Sphere. An FPGA Daughterboard is used to provide a compact means of mounting the FPGA Auxiliary Board and the FPGA board, while also providing power regulation for the different modules, and routing signals.

Design Requirement Specifications

Physical Requirements
- Weight should not exceed 10kg.
- Withstand a 10 meter fall.
- Operate within the temperature range of -30°C to 45°C
- Operate continuously for a minimum of 2 hours before needing recharge.

Surveillance Sphere
- Communicate wirelessly at a minimum unobstructed distance of 35m with a minimum throughput of 40 Mbps.
- Autonomously perform necessary actions for operation, such as camera initialization and parachute deployment.
- Correctly stitch the individual camera images into a single coherent frame at a minimum rate of 15 fps.
- Process the image data into a compressed video stream at a rate of at least 15 fps.
- Deploy a parachute at the apex of the trajectory in less than 0.5s to control the descent.

Command Center
- GUI must have a maximum response time of 100ms.
- Allow user to manipulate video stream playback.

Software Theory of Operation
The operator will interact with the Surveillance Sphere through the Command Center GUI. The GUI will provide the user the ability to initialize the system and display live information about the Sphere such as the state of the Wi-Fi connection, parachute status, and initialization status. The GUI shall also provide the operator the ability to manipulate the video stream in real-time.

Once the Surveillance Sphere is connected via Wi-Fi to the Command Center, launched, and reaches the apex, the Command Center will begin receiving the video stream. The video will be automatically saved to the hard drive on the Command Center, allowing the user to manipulate the data while it continues recording the live stream. The GUI will afford the operator the ability to pan, zoom, fast forward, rewind, go live, stop, pause, go to a specific time, and to take a screenshot of the video. This functionality will be implemented using a video player library. The GUI will pass the command to the library and the final manipulated data will be sent back to the GUI to be displayed to the operator. A Communication Manager will be implemented to maintain communication with the Surveillance Sphere and could support more than one Surveillance Sphere, bandwidth permitting. The Communication Manager will relay control and status signals between the Surveillance Sphere and the GUI while also saving the incoming video stream to the Command Center’s hard drive.

The Surveillance Sphere is comprised of several modules to process data and ensure that the device returns to the ground safely. The Network Client is used to maintain communication with the Communication Manager. The Network Client receives status signals from components in the Sphere as well as the video stream from the Compression Interface and relays those to the Communication Manager via the network connection. It is also used to receive control signals from the Communication Manager and relay those signals to the proper modules. The Accelerometer Data Processing module performs the task of monitoring the data captured by the accelerometer and notifies other modules when the critical points of the trajectory are reached, specifically the apex and the nadir. After receiving an initialize signal, the Accelerometer Control module will properly initialize the cameras. Collecting the six separate camera input feeds and stitching them into a single, coherent image is performed by the Video Stitching module. Once the image has been stitched, it is passed to the Compression Interface which sends the data to the Compression system on a chip (SoC) and then routes the resulting compressed data to the Network Client.