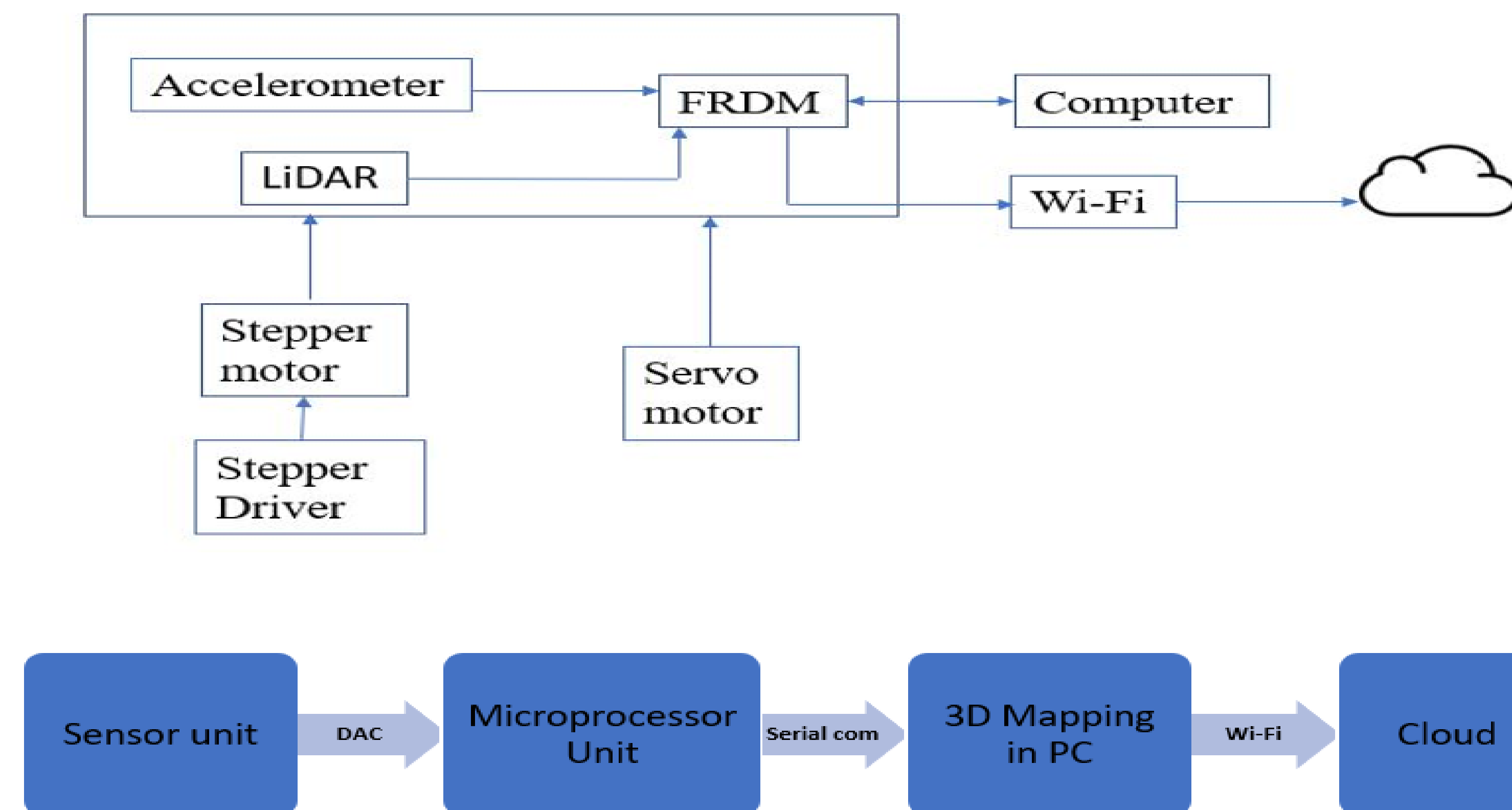




## Introduction

Visual monitoring systems, whether it is a CCTV or a standalone security camera at building tops or pan/tilt security camera at traffic intersections, has played important roles in laying the foundations of surveillance security networks. The recorded video footage of these systems assists in post analysis of events occurred in the past. Recent researches report that biometric identifications of subjects in surveillance video footages with different image processing and machine learning approaches have proved to be effective in crime scene detection and recognition. But the only challenge in all these existing systems lies in the fact that they produce two dimensional data. It has also been seen that machine learning algorithms fail to discriminate objects in low lighting conditions and hence high resolution infrared cameras took over as a visual sensors for night vision.

## Block Diagram



## Specifications

### Hardware

- ❖ LiDAR lite V3
- ❖ ADXL335 accelerometer
- ❖ FRDM K64F Microcontroller
- ❖ ESP8266 WI-FI module
- ❖ NEMA 17 stepper motor
- ❖ Tower Pro MG995 Servo motor
- ❖ Slip ring
- ❖ A4988 Stepper Driver

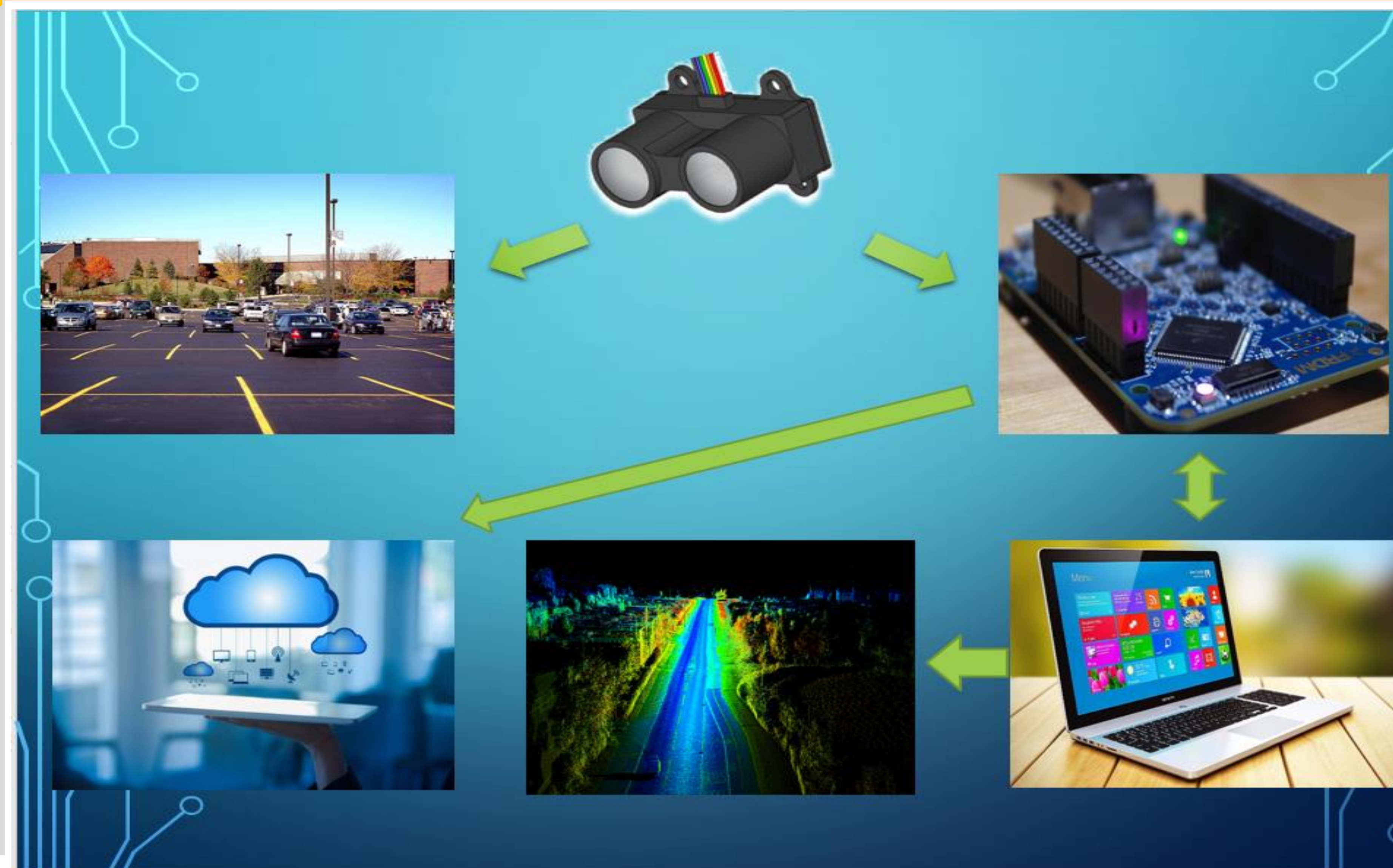
### Software

- ❖ Mbed Online
- ❖ Python
- ❖ PHP

## Aim

The main idea of this proposal is to generate a real-time three dimensional environmental map using sensor fusion of a **LiDAR** and **Inertial Measurement Unit**. The **LiDAR** will be mounted on a **360 degree rotating structure** along with tilt facilitated by a **Servo mechanism** to map points on the spherical coordinate system, whose elevation and azimuth can be used to generate the **point cloud**. This real-time 3D map can be used to localize points of interest in the point cloud and upload their **coordinates** to **Internet Cloud via Wi-Fi**, using IOT terminologies for remote monitoring. However this real-time map can be used as live 3D footage of open space parking lots and hence can be utilized by future autonomous cars to locate empty parking spots and generate a navigation path to the located spot.

## System Implementation



## Prototype

