Introduction

Emergent medical situations happen everyday and everywhere and require proper treatment. However, the current ambulance distribution system is far from effective on providing emergent medical service due to the restriction of time, location, traffic situation, etc. In this case, many people who encounter emergent medical situations do not receive proper treatment and may face threats to their lives. A more effective way to provide urgent medical service is required.

As drones is more widely used nowadays, it has become a very useful tool for entertainment, photographing, and monitoring. However, it has never been used in providing immediate medical service. Drones have great flexibility and can travel a distance of half a kilometer in a few minutes. When an urgent medical situation occurs, a drone can reach the scene and carry a moderate amount of medical supplies in a few minutes. Meanwhile, an ambulance may take 15-20 minutes to reach the scene and provide help. There is a huge potential for drones to help dealing with simple but urgent medical situation like heart attack, asthma, epilepsy. When facing a complex medical situation, drones are not capable of delivering better help than ambulances do, but in other cases, drones can provide more instant service than ambulances do, which may be crucial in saving peoples life. Instead of replacing the current ambulances distribution system, we plan to build a medical drone system that aims to be a supplement to the ambulances distribution system, which can remarkably reduce the workload of system and provide more effective service. The system is great for places like schools, park, shops malls and can provide medical supply (mainly medicines) for medical situations including heart attack, asthma, epilepsy, etc.

Purpose

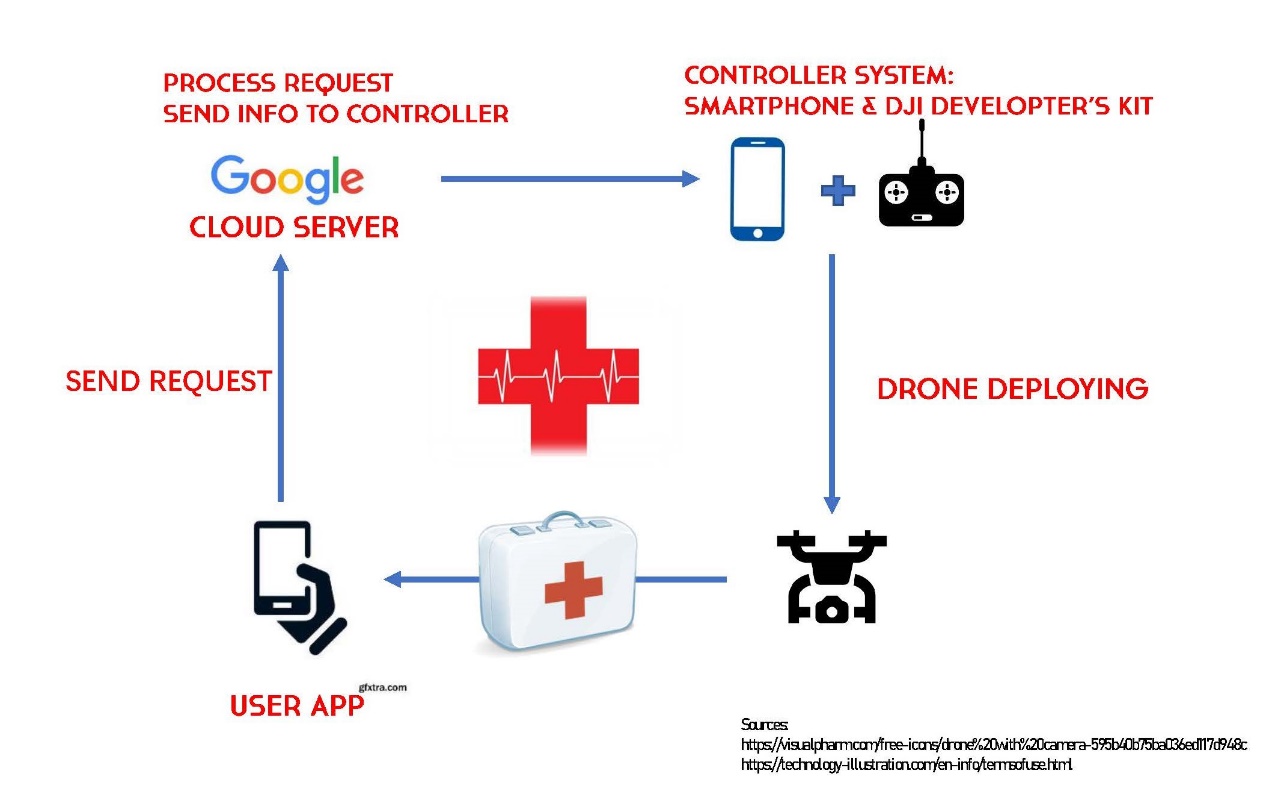
The purpose of the project is to design a medical drone system that allows its users to acquire medical support delivered by an AI-controlled drone under emergencies.

Objective

The objectives of the project are to establish a system, which includes user application, network server, automatic drone controller, and a drone. Users are supposed to be able to require medical support on their smartphones with the application installed. The server should receive request from users and send the request to the drone controller. After processing the request, the drone controller should be able to fly the drone automatically to users and deliver the medical support they have required.

Approach:

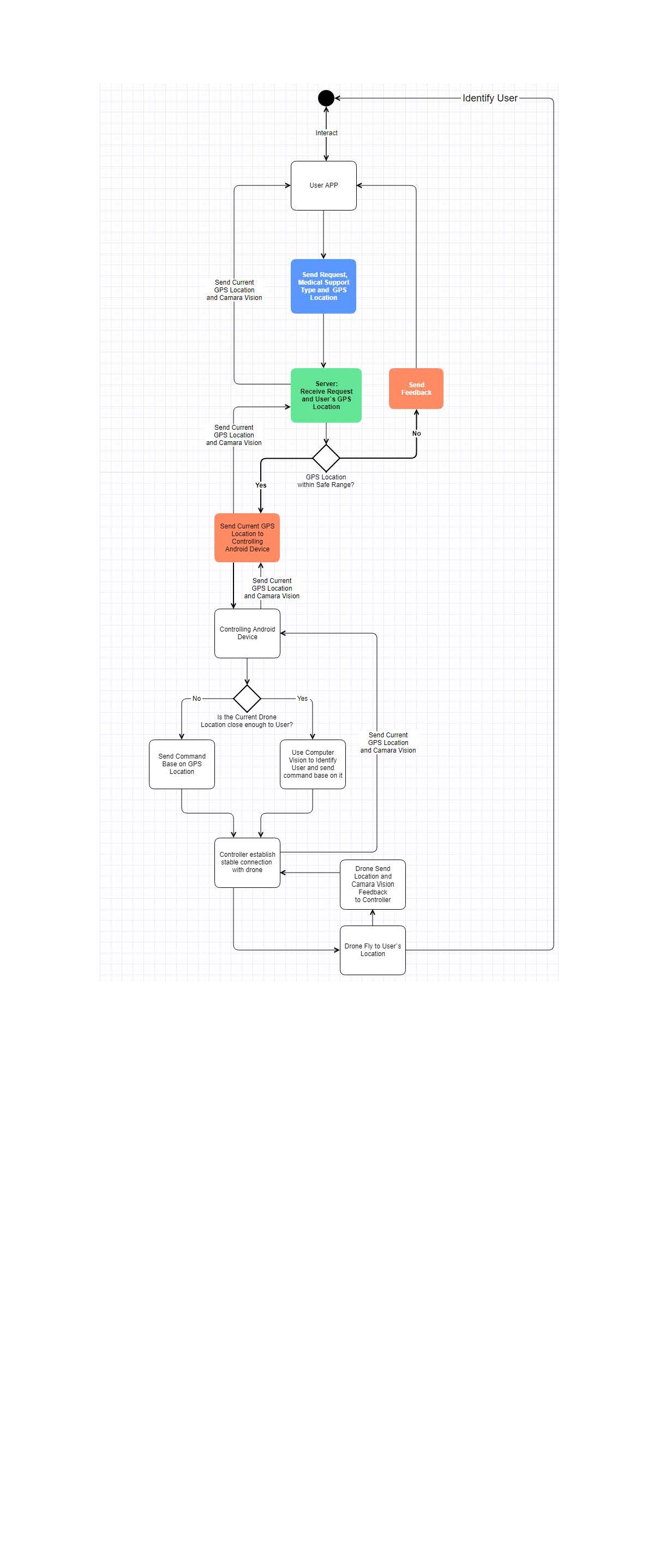
Basic Components:



**Image Processing**

As Shown in the Graph, the physical component of the project can be divided into 4 parts:

1. A user app that allows the user to send request according to the user`s need.
2. A server that can answer the request and communicate with the drone control device
3. An App for the controlling Android device to communicate with server and control the drone.
4. Computer Vision Application on the controlling device to help identifying and locating the user.

The Procedure of The Project can be drawn as follow :

Responsibilities:

We have defined roles for all members of our team, each in charge of a part of the project, which are: User App Developing, Server Building, Controlling and Connection App Developing, User Locating and Image Processing.

Meetings are held every two weeks with our advisor, to update him on our progress and seek advice when needed. Each member are familiar with a different smaller part of the project, like drone controlling and App developing, together we can build the project smoothly.

1. User App Development
   1. Description

We will build an Android APP that will implement the functionalities we need.

* 1. Functions
     1. Allow user to select medical support type and send request to our server
     2. Show the current drone location through google map
     3. Show the current camera vision of the drone on the user App

1. Server Building
2. Controlling Android Device App Development
   1. Description

We plan to use DJI drone as our primary device. DJI is the world's leader in commercial and civilian drone industry, accounting for over 70% of the drone market. DJI`s drone product is widely considered to be safe and reliable, with a mature development toolkit for developers to deploy their application.

We consider to connect the controlling device and server through the Internet, and we make a backup plan for our connection.

* 1. Connection between Server

Basically, we plan to connect the controlling device and our server through Internet.

* + 1. Backup Plan

Since we are not professional in Network Programming, it is possible that the stability and speed of the connection between controlling device and server is not enough satisfying, which may cause safety issues. In such case we will set our personal laptop as the server, and connect the controlling device and server with Bluetooth, which can provide satisfying stability and speed for connection but may limit the ability to answer multiple requests.

* 1. Controlling

DJI provides a mature development toolkit-MobileSDK, providing full access to all DJI drone`s capabilities. With MobileSDK we can simplify low level functionality developing such as flight stabilization, battery management or signal transmission and focus on our application.

We will develop our application base on the MobileSDK and integrate other necessary functionalities.

1. Locating and Image Processing
   1. Description:

We set 3 goals for user locating.

***Basic:***

We let the drone to reach the user through the GPS location, and let the drone land nearby. We will let the drone to land with a safe margin of 6 meters away from the user.

***Advance:***

We identify the user through his posture: we consider to ask the user to do a special posture (like raising the user`s hand or lying on the ground ) so that we can identify the user and land near the user.

***Ultimate:***

We still assume the user are in a special posture. In addition to this we will do a 3D reconstruction from the original image and let the drone do self-navigation.

We also make two backup plan for our locating mission.

* 1. Model Selection

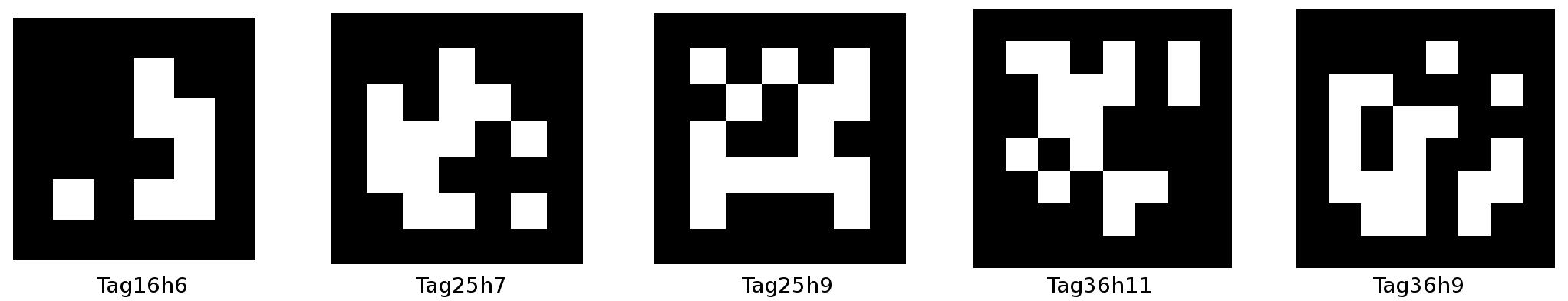
We plan to use YOLOv3 or TinyYOLOv3 to do Object Detection. YOLO is a deep Object Detection model with the highest processing speed and a satisfying accuracy. TinyYOLOv3 can provide a processing rate at 30 FPS on FPGA+DV500/DV700 AI accelerator.

* 1. Backup Plan1: Process On Server

In case the compute capability of the Android device does not meet our requirement, we will do the image processing on the server and send the control command to the controller directly.

* 1. Backup Plan2: AprilTag

AprilTags are 2D barcodes with a corresponding library that helps to localization.



The library will detects any AprilTags in the image and return a unique ID, coordinates for each AprilTag, if the size is known and camera is calibrated.

We plan to set some special locations and place some AprilTag on it, so that after receiving the request the drone will fly to the nearest AprilTag location to the user and land on it through locating the AprilTags. User can go to the nearest AprilTag spot to acquire medical support delivered by drone.

1. Others

***Reason for Drone Selection:***

Instead of build a drone from startup, we plan to use DJI drone which is more safe and stable. In addition, DJI provides a mature development toolkit for drone controlling, which will allow us to focus more on our application.

***Reason for Separating Controlling Device and Server:***

We plan to only use one drone for our project. However, in order to build a “system” that allows multiple request to be answer simultaneously, we want to separate the request answering part and the controlling part. In addition, we don`t want to build our system on a DIY connection between drone and server, instead the connection build by DJI is much more trustworthy, which also accounts for our choic.

Budget:

|  |  |
| --- | --- |
| DJI Spark | 399 |
| AWS Developer kit | 29 |
| Sum | 428 |
| DJI LightBridge(Optional) | 332 |
| Sum(With Optional) | 760 |

Timeline:

