Software Engineering

Group 8

Search And Rescue Assistant (S.A.R.A.)

https://abhiek187.github.io/emergency-response-drone/

Mar. 3rd, 2019

Team Members:
Sahana Asokan
Won Seok Chang
Avnish Patel
Abhishek Chaudhuri
Shantanu Ghosh
Srikrishnaraja Mahadas
Sri Sai Krishna Tottempudi
Vishal Venkateswaran

Individual Contributions Breakdown

Project Deliverable	Sahana	Won Seok	Avnish	Abhishek	Shantanu	Krishna M.	Krishna T.	Vishal
Interaction Diagrams (10 points)	10%	25%	10%	20%	15%	20%	15%	10%
Design Principles (10 points)	25%	20%	15%			20%	20%	
Merging Contributions (11 points)				100%				
Project Coordination (5 points)						100%		
Plan of Work (2 points)				50%	50%			
References (-5 points)	12.5%	12.5%	12.5%	12.5%	12.5%	12.5%	12.5%	12.5%

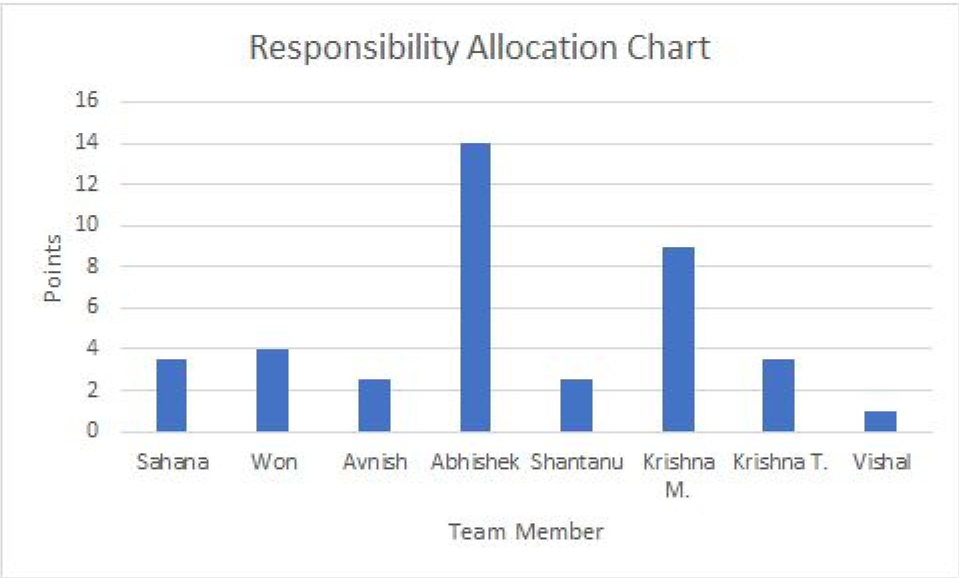
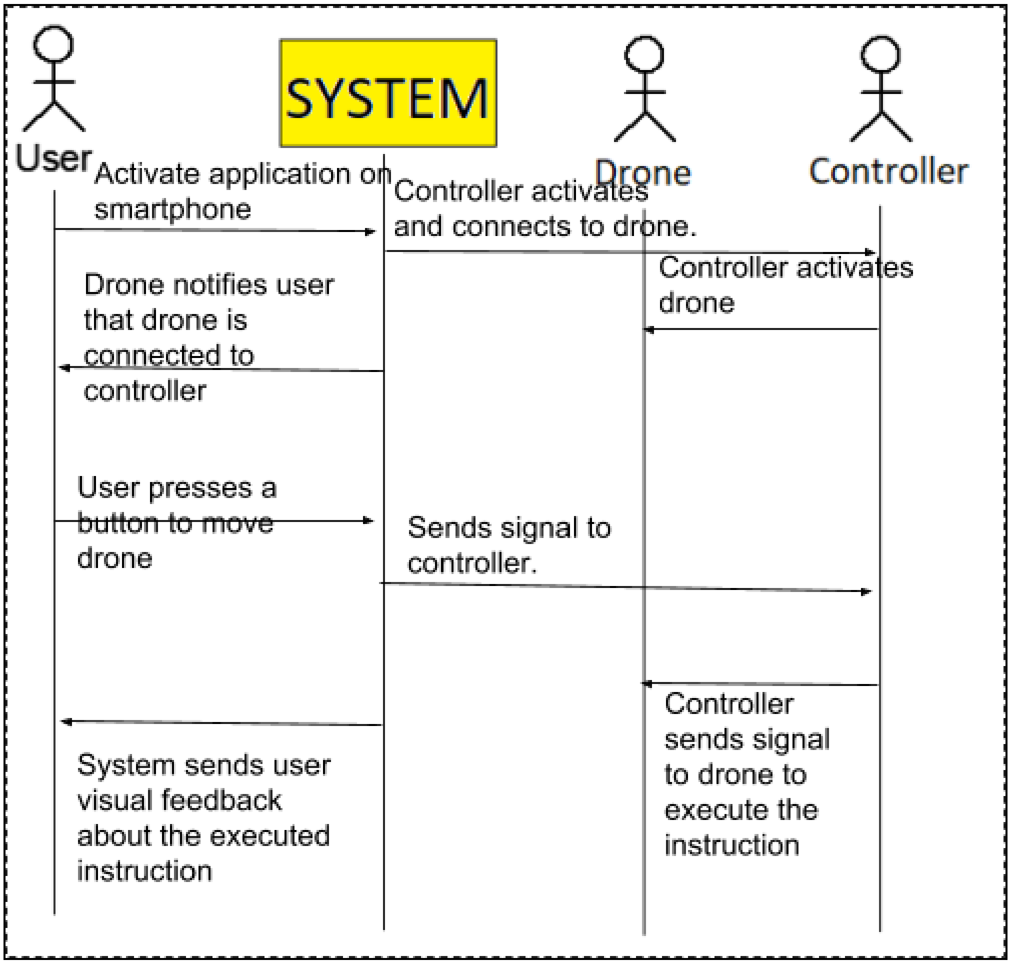


Table of Contents

Interaction Diagrams and Design Principles

Use Case 1 - Move Drone:

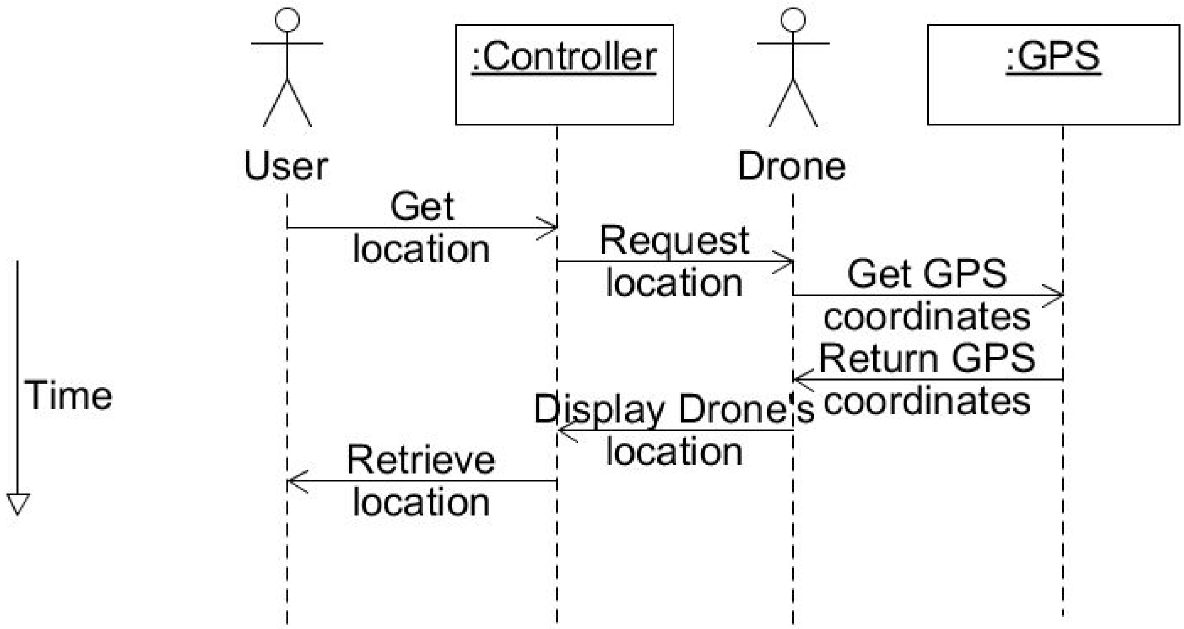


The diagram for the first use case is displayed above. In this case, the User will first activate the system in order to gain access to the controller. From there the User is able to use the controller to move the drone. The User will be able to see visual feedback from the drone.

Design Principles:

The design principles utilized in this use case include the Low Coupling Principle. This design principle is utilized as the communication links that exist are very short. Most of the communication is done between the User and controller, and then the controller and drone.

UC-3: GetLocation

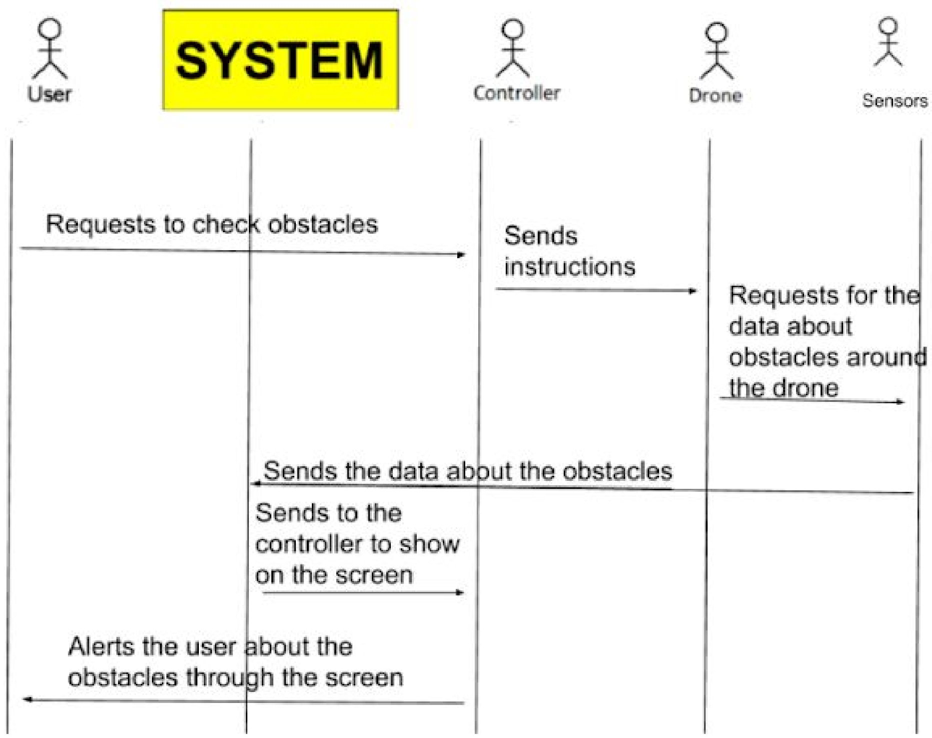


The diagram above demonstrates the interactions between classes in UC-3: get the location. Once the user has control of the drone and being able to maneuver around obstacles. First, the user sends a request to get the location of the drone to the controller which then gets requests it to the drone. The drone then requests the coordinates to the GPS server and gets the coordinates and sends it back to the controller to make it visible. The user sees the drone's location based on longitude and latitude.

Design Principle of UC-3:

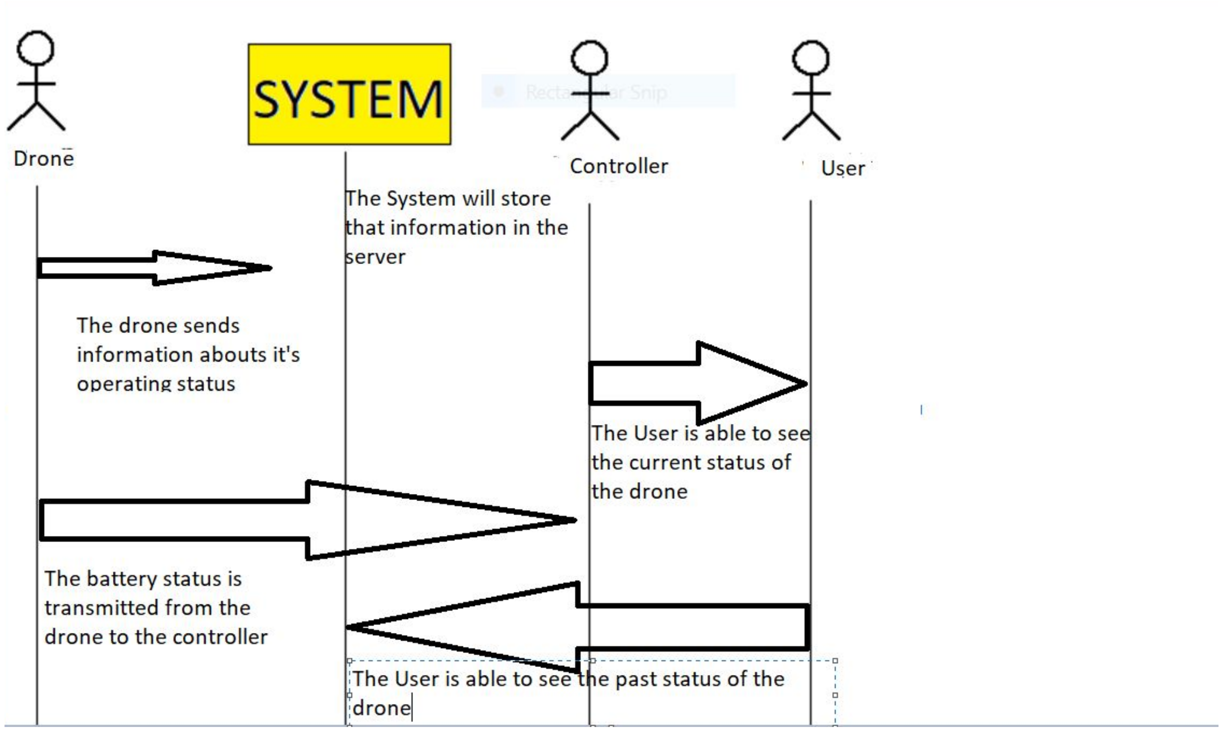
The design principles employed in the process of assigning responsibilities were the expert doer principle and high cohesion principle. The expert doer principle is used because each of the classes is an expert for specific functions. An example, the drone is responsible for getting the coordinates from the GPS server and relaying it back to the controller for the user to see.

Use Case 4 : Check Obstacles



Design Principle:
The design principle for this use case is the expert doer principle and high cohesion principle because the parameters for the obstacle is super specific. And that data should be focused on because it can affect the overall behavior of the drone. It is also important that the specific obstacles that are being checked for are being communicated to other sources.

Use Case 6 -Get Status



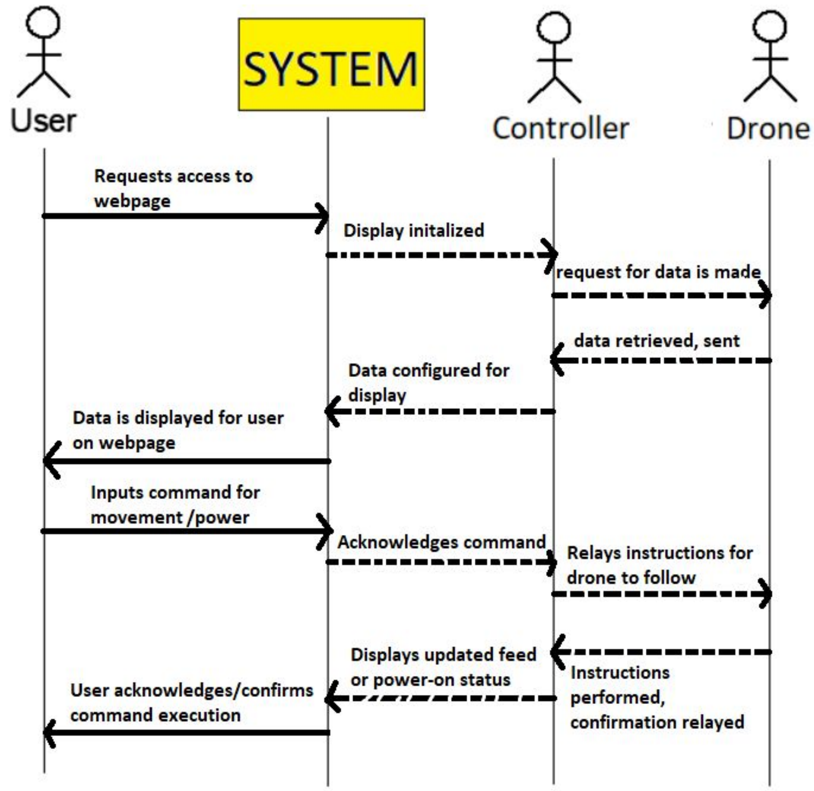
The interaction diagram for use case 6 is displayed above. The drone basically sends a signal to the system which the user can see the result of through the controller. The part of this use case is to let the user know of the operation status of the drone, in particular, the battery level. The user is also able to view the past values sent by the drone.

Design Principles:

The design principles utilized by this use case are Expert Doer Principle, High Cohesion Principle, and the Low Coupling Principle. Since this use case is the only use case that knows about the battery status it makes sense that the Expert Doer Principle is used. As for the High Cohesion

Principle, the only computation done by this part is the battery level. The Low Coupling Principle deals with the concept that this use case does minor communication between the drone and the controller.

<u>Use Case 7: GetData</u>

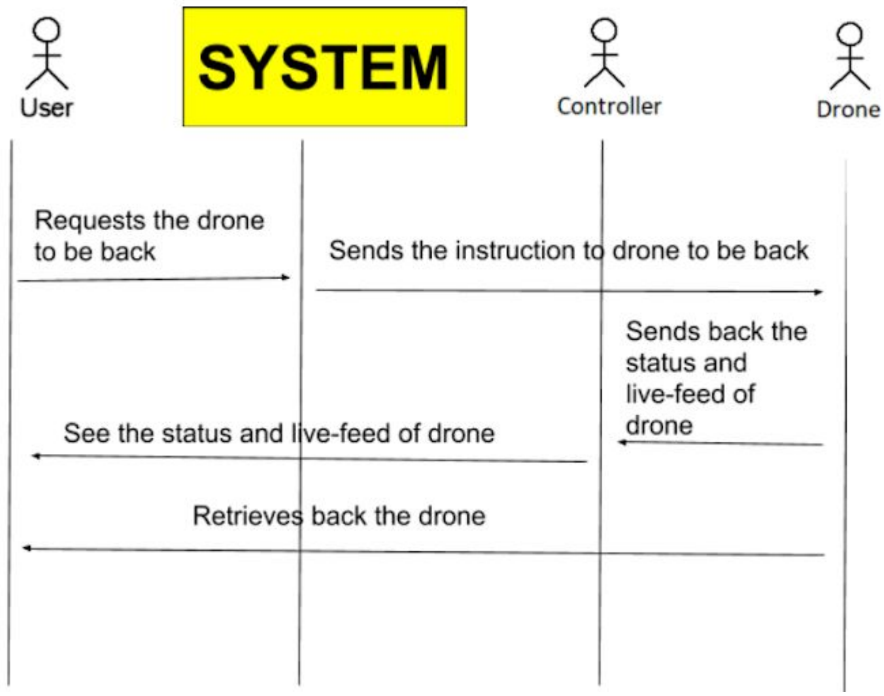


The diagram above demonstrates how the user, controller, and drone interact with each other to show necessary data of drone to the user, so the user can control the drone. When the system needs data, the drone sends the data that is saved on it to the controller upon the request by the controller. When the controller receives the data, it displays it on the webpage, so the user can see the data and make necessary judgments of controlling the drone. The user will verify its execution by the updated live-feed.

Design Principles:

The design principle of this use case is High Cohesion Principle. There is more focus on displaying the necessary data and sending instructions to drone to control it, rather than having a high responsibility of computing data.

Use Case 8 :Return to Home



Design Principles:

The design principle for this use case is both Expert Doer Principle and Low Coupling Principle. It is important that this object reports when the drone needs to be returned to the original location and to be able to recognize when its supposed to return to the original location. It is also implementing Low Coupling Principle because it should only be focusing on data and communication relating to returning home.

Project Management and Plan of Work

Merging the Contributions from Individual Team Members

Shantanu came up with the project idea and was able to explain how we could contribute to the project during weekly meetings. We decided to split the work into four subgroups: image processing, location data, physical data, and obstacles. Since not everyone can make it to the weekly meetings, each subgroup has set up their own meeting times to discuss specific functionalities to be implemented in this project. This also ensures that each person can discuss how they will contribute toward building S.A.R.A.

Krishna Mahadas created and shared the Google Drive for our project so we could easily collaborate on creating the reports.

Abhishek manages the GitHub repository to maintain the project code and divide the work among the team. Each branch corresponds to the different subgroups. Each person works on their subgroup work and when it's ready to be implemented, it is merged into the master branch.

A website is going to be made and developed with relevant updates to the project. This will be managed by Abhishek. Other team members will help.

Project Coordination and Progress Report

Image Processing:

The image processing component of the project mainly implements the use case ViewCamera. So far we have already been able to display what the phone camera is seeing on other devices such as a pc. We tried using multiple third-party applications and features of the Android phone to see which works best. Some third-party applications we tried are Alfred and IPwebcam. Both of these applications are able to display decent quality video feed for a reasonable range using wifi. Another approach was using the screen mirroring function of the android. This approach also uses wifi and provides a really good quality image. However, it does not have much of the range due to the fact the phone needs to be close to where ever the display is being transmitted to. So we decided to try using the Alfred application for now. Currently, we are trying to manipulate the given video feed from the application, so that it is possible to transfer the video feed to our user interface. This is being done by using the HTML code of the webpage version of the application since the camera feed is currently being displayed there.

Location Data:

The use case that is the main function of location data is GetLocation. We already have code for this use case in HTML that provides the location of the given device in latitude and longitude form. The next step is to transmit this information from the android device to the user interface.

Physical Data:

The physical data part of the project deals with the GetStatus and GetData use cases. It will also include the MoveDrone use case. Due to the hardware component of the project, this part of the project can be in effect once the drone is in full operation. So currently, we are all working towards the construction and integration of the hardware and software components of the drone.

Obstacles:

The Obstacles section of the project deals with the remainder of the use cases. The use cases include CheckObstacles, AvoidObstacles, and ReturnToHome. Similar to the physical data component, this section will mainly be in effect once the drone is working. So as mentioned before we are all working to complete the hardware and software parts of the drone.

Plan of Work

- Milestones:
  - *Drone Camera Transmission:* Be able to provide a reliable stream from the onboard phone camera to a mobile device set aside to mock the operator's control device.
    - Date of Completion: March 8th, 2019
  - *Hardware-Associated Tasks:* After all necessary hardware components arrive between March 1st-3rd, the construction of the drone frame to fit the needs of the project. This includes mounting the onboard camera and microcontroller to the drone frame.
    - Date of Completion: March 20th, 2019
  - *Onboard Data Management/Transmission:* Determining the operational status of the drone from real-time data/status of equipment, signal, etc. This needs to be successfully relayed back to the operator's control device.
    - Date of Completion: April 1st, 2019
  - *Webpage Integration:* Collecting all relevant data and finalizing transmission/display of said data to the operator's control device.
    - Date of Completion: March 22nd, 2019
  - *"Crash-testing":* Success of Flight testing to determine drone's survivability
    - Date of completion: April 8th, 2019
  - *Obstacle avoidance:* In addition to recognizing objects in its' way, the drone will react to maneuver out of harm's way/avoid pathing into roadblocks
    - Date of completion: April 15th, 2019

Breakdown of Responsibilities

- Project divisions:(all tasks that are in progress/to be completed)
  - Visual Data Processing:
    - Shantanu: Management of the main wireless network/communication of data
    - Abhishek: Webpage development/Data handling on operator-side
    - Krishna Mahadas: Onboard camera handling, transmission (in progress)
  - Obstacle Management
    - Vishal: Managing sensor data, implementing avoidance/assoc. movement
  - Location Data
    - Avnish: Gathering onboard GPS data, transmission
  - Physical Drone Data
    - Krishna Tottempudi: Determining overall operational status from collected data
    - Sahana: Determining power levels/operational lifespan of drone real-time
    - Won Seok: Determining the strength of signal/connection to the operator

All other contributions to the project can be found in the individual contributions breakdown matrix on page 2.

References

https://www.dronesense.com/?gclid=EAIaIQobChMIqo-45_Gx4AIVwoCfCh2CbA0QEAAYASAAEgKMu_D_BwE

Image 1:
https://s.yimg.com/ny/api/res/1.2/2P8Y6UqlB8dKOiVIg9Rscg--~A/YXBwaWQ9aGlnaGxhbmRlcjtzbT0xO3c9ODAw/http://media.zenfs.com/en-US/homerun/digital_trends_973/8122e594705a009db372bf32720d9fe9

https://www.aopa.org/news-and-media/all-news/2018/october/01/drone-study-reveals-potential-and-limits

Coast Guard Table:
https://www.dco.uscg.mil/Portals/9/CG-5R/SARfactsInfo/SAR%20Sum%20Stats%2064-16.pdf