Integrating Autonomous Drones into the National Aerospace System

Connor A. Theilmann

Thesis Advisor: Dr. Vijay Kumar
Engineering Advisor: Dr. Sampath Kannan

Senior Capstone Thesis (EAS 499)
University of Pennsylvania
School of Engineering and Applied Science
Department of Computer and Information Science
April 29, 2015

Table of Contents

1. Introduction

With the recent proliferation of unmanned aerial vehicles (UAV) or “drone” technology, a great deal of uncertainty regarding their automation, commercial application, and ethics has been raised. High levels of connectivity through matured cellular, Bluetooth, Wi-Fi, and radio protocols in conjunction with smartphone ubiquity and increasingly comingled hardware and software has resulted in rapid advancement in drone capabilities. Easy access to these low cost UAVs has democratized access to aerial footage, mapping services, data collection and aerial logistics. These technical capabilities in addition to recent progress in computer vision and motion planning has made independent, automated drone tasks a reality.

Due to their wide range of capabilities and imminently possible autonomous function, many have raised questions regarding the safety, dependability, and ethics of drone operations. The tethering of the hardware to robust software has allowed for the possibility of these systems to operate autonomously for myriad consumer and commercial uses. However the way in which we regulate, monitor, and control these systems is still very much up for debate. Specifically, issues with privacy, national security, airspace monitoring, and obstacle avoidance must be considered when drafting legislation and developing the software and hardware for the next generation of UAVs.

In this paper, we will explore the many of the topics related to the integration of UAS operation into the National Aerospace System (NAS). Specifically, we will focus on the function and operation of the software that powers autonomous flight. We will review the technology landscape in conjunction with the ethical, regulatory, and practical concerns it presents to draw conclusions about how drones could eventually operate autonomously for commercial or consumer purposes in the United States.

Section 2 will provide a brief history and background of drone technology, Section 3 will provide an in-depth look at some of the popular algorithms and software used to control autonomous drones, Section 4 will explore some of the interesting use cases and introduce the current public perception of drones, Section 5 will provide a detailed overview of current drone regulation, and finally, Section 6 will touch on the technology and industries that must emerge for widespread adoption to be a reality.

2. Drones - Unmanned Aerial Vehicles (UAV)

2.1 A Brief History of Drones

A) Military Origins

Drones, like many other major technological innovations, have military roots. The first recorded use of what we consider to be drone operation occurred in 1849 when Austrians launched 200 pilotless balloons equipped with bombs against the Italian city of Venice.[1] Though these balloons don't meet the requirements of truly being a UAV, it was the first attempt at creating a pilotless system capable of accomplishing a task.

The first pioneer in what we consider to be a UAV is Nikola Tesla. In 1898, Tesla wowed a crowd at Madison Square Garden by controlling the direction of a small, unmanned boat with changes in radio frequencies. Later that year, he filed a patent that detailed his novel apparatus for controlling motion of a vehicle with a radio frequency, a concept Tesla described as *teleautomation*.[2] This device is widely conceived to be the first electronic remote control.

Tesla's work set the stage for the pioneering work of military focused pilotless aircraft by both British and American groups. In 1917, British Captain Archibald Low created a set of wooden aircraft equipped with explosive warheads that were known as "aerial targets." At the same time (and with a bit more success), Elmer Sperry and Peter Hewitt developed the "Hewitt-Sperry Automatic Airplane." The airplane, also known as the "flying bomb," was capable of flying 50 miles carrying a 300-pound bomb without a pilot. Most importantly, the aircraft utilized Sperry's invention of the "gyroscopic compass" to stabilize the flight trajectory of the unmanned aircraft.[3,4] This aircraft is widely regarded as the predecessor to the cruise missile.

FIGURE 1: HEWITT-SPERRY AUTOMATIC AIRPLANE

The next major iteration of UAS technology occurred during World War II when Reginald Denny determined that there was a demand for low-cost radio controlled airplanes for use as target practice by military anti-aircraft gunners. Denny sold 15,000 of his Radioplane OQ-2 aircrafts to the army during the war, thousands more of future iterations in the years after. Interestingly enough, it was at a Radioplane manufacturing facility where actress Marilyn Monroe, known then as Norma Jeane Dougherty, was first discovered through a photograph for the army magazine *Yank*.

FIGURE 2: MARILYN MONROE WITH OQ-3

FIGURE 3: OQ-3 USED AS TARGET PRACTICE

However, it is Edward Sorensen who is widely credited as the father of the modern UAV. His patent on a radio remote control system in 1946[5] stands as the foundation for modern radio controlled systems.

Further drone development was catalyzed in 1960 after U.S. Pilot Francis Gary Powers was shot down over the USSR while flying a U-2 aircraft for reconnaissance. The incident led to the US investing heavily in aircraft capable of gathering intelligence autonomously. The incident led to the United States Air Force (USAF) granting funds to the Ryan Aeronautical Company to produce a reconnaissance focused version of their "Firebee" target drones. The reconnaissance drone was code named "Red Wagon."[6] These drones, known colloquially as a "Lightning Bug" were launched from the wings of Lockheed Martin DC-130 Hercules airplanes and had pre-programmed flight patterns in addition to being controlled by pilots on board. These drones were not capable of landing, and instead needed to deploy parachutes to be recovered. Ryan drones were used extensively in Vietnam and South China by the USAF exclusively for reconnaissance purposes. The Firebee's created major political controversy when one was shot down in South China and led to the Chinese declaring a "major victory in the war."[7]

Finally, the last major innovation regarding militarized drones came with the introduction of the General Atomics MQ-1 Predator drone in 1995. Under the guidance of Iraqi born Abraham Karem, General Atomics produced the first widely used militarized drone in the Predator.[8] The Predator forever changed the nature of warfare by allowing for precise attacks to take place without risking human life. The Predator major innovation was in its ability to be controlled remotely by satellite relay as opposed to a nearby ground station line of sight link. The drone could operate at distances up to 460 miles from its launching station, and hover for up to 14 hours at a time. This enabled pilots in Nevada to control aircraft engaging in lethal missions across the world. With a per unit cost of around $4mm, these aircraft provided a relatively inexpensive and safe alternative to ground based missions or piloted aircraft attacks. According to their manufacturer, General Atomics, these drones have become so popular that at any given moment at least 54 Predator-series drones are airborne worldwide.[9]

B) Smartphone Enabled Consumer Drones

In 2010, Parrot's AR 1.0 drone changed the future of civil UAVs. While there was some availability of consumer focused UAVs prior to its launch, when French based Parrot unveiled the AR 1.0 at the 2010 Las Vegas Consumer Electronics Show, the $400 quadcopter's capabilities were beyond what anyone had seen before. The quadcopter was capable of being controlled by an iPhone app, and allowed users to take low quality photos and videos.[10] The AR 1.0 introduced the concept of aerial photography to the masses, though the flight time, maximum flying altitude, and photo quality were all minimal.

However, the current market leader is clearly Shenzhen, China based manufacturer DJI. Founded by Frank Wang in his dorm room at HKUST in 2006, DJI originally focused exclusively on producing the flight control software before launching its own hardware product. It wasn't until January of 2013 that DJI introduced what many consider to be the precursor to modern consumer drone technology. With a high-quality GoPro camera attached to the gimbal, the drone was capable of being controlled by a radio controller and relaying a live-stream camera feed to the user through a smartphone application. The drone had significant improvements in stabilization, flight altitude, flight time, and image quality over the Parrot predecessor. Its easy and dependable use, in addition to its high quality images allowed for democratized access to aerial photography for the masses, and sparked the consumer and commercial drone craze.

FIGURE 4: DJI PHANTOM 1

FIGURE 5: PARROT AR 1.0

2.2 Terminology and Function

A) Definition of UAV

For our purposes, an unmanned aerial vehicle (UAV) is a powered system capable of achieving flight. The system must be recoverable, capable of being operated remotely or with on-board software, and should be capable of carrying a payload that is not essential to its flight. The FAA further classifies a small UAV (sUAV) as weighing less than 55 pounds.[11]

B) Rotorcraft vs. Fixed Wing

The major delineation between UAVs comes in the orientation and number of propellers it uses.

Fixed wing aircraft use long wings to achieve lift and have forward facing propellers that provide throttle propel the aircraft. These aircraft are incapable of hovering over a specific area as they must always be moving in the horizontal or vertical direction, and use flaps on the wings and tail to adjust its position. Fixed wing aircraft have advantages in energy consumption as the shape and orientation of the wings provide lift and thus do not require energy consumption, but are far more difficult to precisely control compared to their copter counterparts.

Rotorcraft (also known as multirotor craft), differ from fixed wing vehicles in that they use rotors with rotating blades pointed vertically to provide lift. They contain one or more sets of rotors, each of which generally contains one to two blades. Multirotors are divided into classes of vehicles based on the number of rotors the aircraft uses. Generally, the Latin prefix for the number of rotors is followed by the suffix “copter.” For instance, aircraft with four rotors are called quadcopters, and aircraft with eight rotors are called octocopters. Multirotors generally use brushless DC motors to rotate the rotors and generate lift.

C) Brushless DC Motor

We will focus on multirotors, as they are of primary focus within the sUAV category. Most sUAVs use brushless DC motors to rotate the rotors either clockwise or counter-clockwise and create upward force, called thrust. The motors contain a circular configuration of center-facing coils, and a strong magnet (usually neodymium) rooted in the rotor in the center. The motors work by providing pulses of DC current to the coils to create a repelling and rotating electromagnetic on the magnet. Because the electromagnetic force is directed towards the center magnet, but rotating in location, the magnet constantly spins to re-align itself with the magnetic fields, thus also rotating the rotor and blades. The amount and sequence of power provided to the coils directly affects the direction, speed, and thrust generated by the rotor.

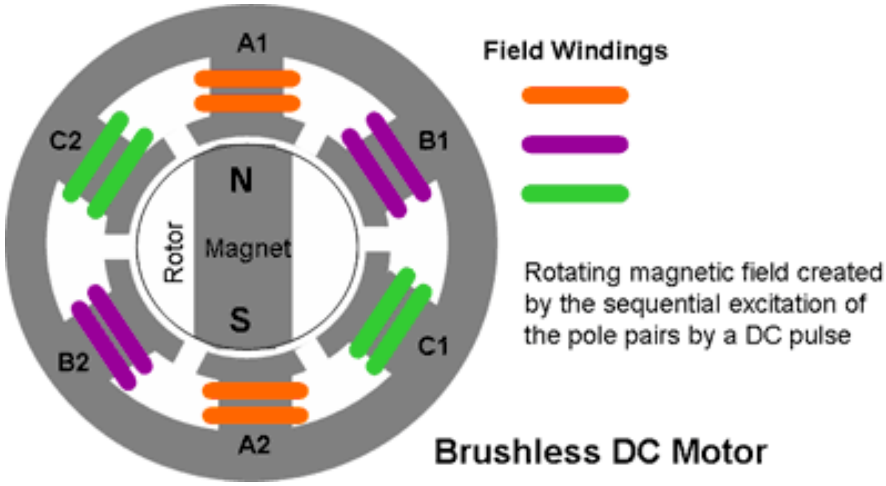


FIGURE 4: BASIC DIAGRAM OF BRUSHLESS DC MOTOR

D) Flight Dynamics of sUAV – Yaw, Pitch, and Roll

The flight dynamics of a multirotor, or any aircraft, is described using three angles of rotation – yaw, pitch, and roll, sometimes called the Euler Angles. These three angles describe how the aircraft rotates around each of the three dimensions around its center of mass. Changing the pitch will make the copter go forward or backwards, changing the roll will move the copter left and right, and changing the yaw will rotate the copter around its vertical axis.

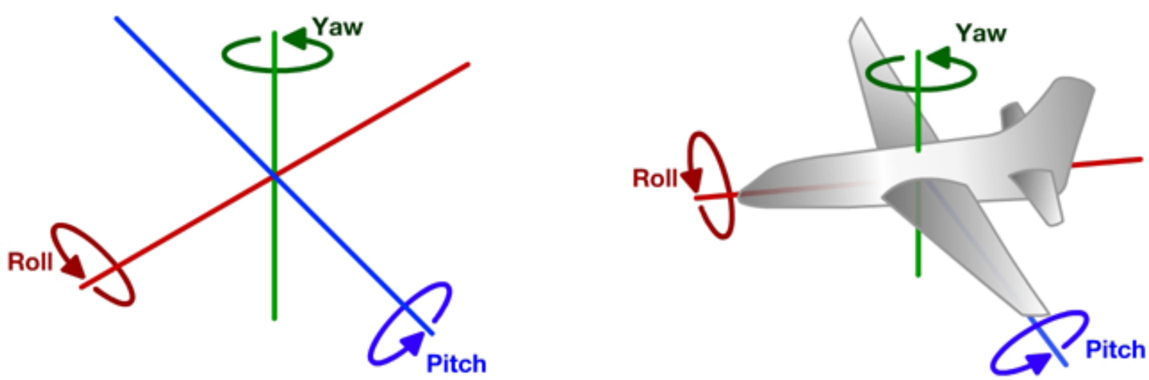


FIGURE 5: YAW, PITCH, AND ROLL

Creating thrust to move the UAV vertically is trivially achieved by increasing the speed of all the blades. However, moving in a more complex trajectory in 3D space requires more algorithmic thinking.

E) Electronic Speed Controller (ESC)

Combinations of roll, pitch, yaw, and throttle require more complex coordination of rotor speed, and are accomplished algorithmically by electronic speed controllers (ESC). Each rotor has its own ESC to provide levels of current to the motors such that they will spin at a desired speed and direction. An embedded microcontroller known as a MOSFET operates hundreds of times per second within the ESC to pulse currency in sequences that will rotate the blade. The central computation system achieves desired flight dynamics by coordinating blade rotation through inputs to the ESC's. For example, clockwise yaw requires the front right and back left rotors to rotate faster than other motors to achieve yaw.

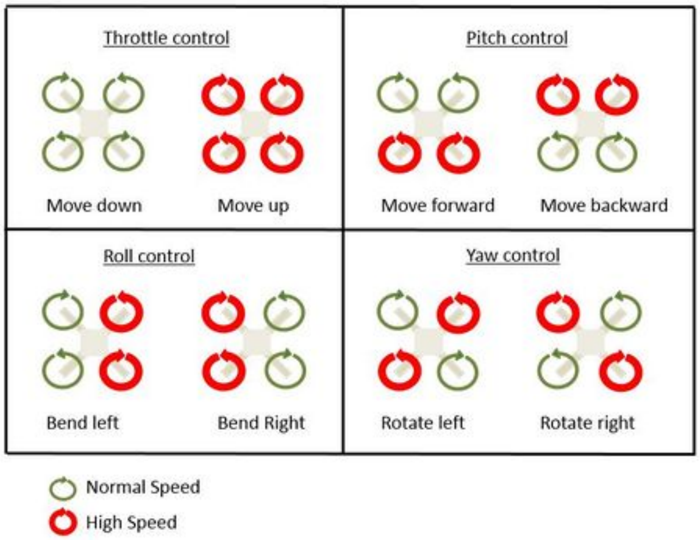


FIGURE 6: ADJUSTING SPEED OF ROTORS TO ACHIEVE DESIRED ORIENTATION

F) Gimbal

A gimbal is a payload that carries a camera or other sensor and stabilizes the payload's orientation and rotation in up to three axes. Using a gyroscope, the gimbal allows the camera or sensor to orient and stabilize itself independently of the multirotor. It is used when end users are capturing images or video over time and need smooth, stable footage. Without a gimbal, sensor footage is susceptible to noise from vibration or jerky, aggressive motions when the vehicle accelerates or changes attitude. They can also be extremely helpful when there is need for independent control of the camera angle– i.e. in cinema production by the director/cinematographer or in "follow-me" functionality (discussed in 4.2 Aerial Photography) object/person by software onboard the drone.

G) Sensors

Sensors allow UAVs to collect data about its state or its surroundings. We will cover a few of the most common here:

- **Inertial Measurement Unit (IMU)** – the IMU is the most crucial sensor for determining a UAVs state. The IMU uses a combination of accelerometers and gyroscopes to determine the craft's velocity, orientation, and gravitational pull. Interpreting its output allows us to represent the craft in 3D space with 6 degrees of freedom (DOF) – using both the 3D position of the center of mass and the 3 flight dynamic angles (yaw, pitch, roll). Generally, sUAV carry MEMS-based IMUs. MEMS stands for microelectromechanical systems, and allows for the IMU to be much lighter and smaller than others. However, MEMS based technology is more susceptible to noise than larger IMUs. Additionally of note, IMUs can represent the UAV with variety of DOF depending on the sensors it utilizes. For instance, the most basic IMU units contain an accelerometer and a gyroscope and represent 6 DOF. However, because both rely on gravitational forces, neither can represent the actual yaw angle. To be able to correctly calculate the current yaw angle, a magnetometer (compass) must be used, and this combination of sensors would allow for 9 DOF. Sometimes, the altimeter is also considered a part of the IMU, and would then result in the IMU representing the UAV with 10 DOF.

- **Altimeter –** Measures altitude of the aircraft. Sometimes grouped in with the IMU.
- **Camera –** Takes in light and converts it to a digital representation of the world.
    - **Monocular Camera:** Traditional camera. One lens, and one sensor capturing photons.
    - **Stereo Camera:** Has two or more lenses to allow it to capture 3D images by simulating human binocular vision. Range of the stereo system is limited by the baseline of the cameras. Generally, the range is directly correlated by a factor of 100 to the equidistant point between the lenses. For instance, if a stereo camera has 2 lenses that are 10 cm apart, it should be able to detect images that are 10 m away.
    - **RGB-D:** Capture traditional RGB color images, but augment each pixel with a "depth" of data. These are the sensors used in the Xbox Kinect. Kinect-style sensors are also more generally called *structured light sensors*. Typically these sensors don't work outdoors or in sunlight.
- **Lidar –** Using the known constant speed of light, Lidar sensors shoot rapid pulses of laser light at surfaces and measure the amount of time it takes for each pulse to return. Using the return time, the sensor can compute the distance between itself and the surface. Lidar is unaffected by sunlight (whereas cameras are very), and the fastest at measuring distance to surfaces of all the sensors listed. These sensors are particularly useful as the amount of light in the environment will not affect the measurements, so they can be used in both the day and night. However, they are very expensive and clunky at the moment.
- **Sonar –** Using a similar concept to Lidar, sonar sensors send pulses of sound with known speeds, called "pings," at a surface. By measuring the time it takes for the pulse to return to the sensor, it determines its distance from the surface. Because sounds waves are relatively slow, sonar is used mostly in indoor environments or in landing where the UAV is very close to the surfaces it is trying to detect. Additionally, these sensors are particularly susceptible to interference. For instance, if multiple UAVs utilizing sonar sensors are operating in close proximity they will 'hear' each other's pulses, resulting in inaccurate measurements. These sensors are relatively cheap, and lightweight, so there are specific use cases (especially in consumer UAVs) where they are desireable. .
- **Radar –** Again, using the concept of measuring pulse arrival times, radars send waves in the electromagnetic spectrum to map the drone's surroundings. Differences in the time of arrival of the reflected radar signal create variations in waves which, when combined, can create an image of objects.[12]
- **Infrared –** These scanners create heat maps by detecting infrared energy and converting it into electronic signals that we can represent as a picture of a given environment. These are especially useful in nighttime or cloudy conditions.

H) Flight Controller

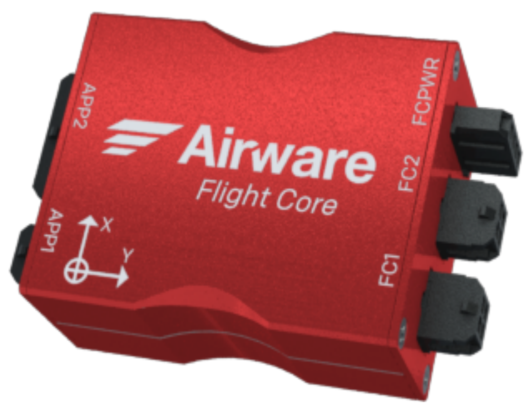
The flight controller is the programmable computational brain within the drone. Generally, it is a printed circuit board with an Intel or ARM processing chip as the CPU. It takes in information from the onboard sensors, and responds to requests from the pilot or installed software. These requests can include simple tasks like reporting the status of drone components, to more complex, algorithmic tasks like controlling the ESC's to stabilize the aircraft against wind using input from sensor data.

For a more in-depth perspective on the algorithms and software embedded on flight controllers, check out Section 3. Software Behind Autonomous Drone Operation below.

(A)

(B)

(C)



(D)

FIGURE 7: POPULAR FLIGHT CONTROLLERS (A) 3DR PIXHAWK (B) DJI A2
(C) PRECISIONHAWK LATAS (D) AIRWARE FLIGHT CORE

3. Software Behind Autonomous Drone Operation

The goal of UAV integration into the National Aerospace System is to allow for safe and regulated use of three types of flying scenarios. The first, and most simple, is visual line of sight (VLOS). VLOS simply means that the pilot is controlling the aircraft, and the aircraft remains within visual sight of the pilot at all times during the aircraft's operation. The second, beyond line of sight (BLOS), is the situation in which a pilot is still controlling the aircraft's motion, but is unable to see the aircraft. The third is autonomous UAV operation, where the pilot is entirely removed from having control over the UAVs operation. The UAV is controlled by dynamic software that regulates the aircraft's motion. In this section, we will explore some of the most recent techniques for allowing UAVs to effectively operate autonomously.

Due to their popularity, I will focus only on techniques that involve using visual based sensors or laser based range sensors to assist in autonomous operation.

3.1 Definitions

A) Navigation System

From a high level, navigation is the process of monitoring and controlling the movement of a craft or vehicle from one place to another. It involves the process of data acquisition, data analysis, and extraction to interpret the vehicle's states and its surrounding environment. It accomplishes these tasks with the goal of completing missions safely.[13] The most important functions for this level of autonomy are:

- **State estimation** is the process of using onboard sensor measurements to estimate variables related to an aircraft's state. Particularly of interest are position, orientation, and velocity – these variables are necessary for controlling the vehicle. The position and orientation of the UAV is known as the *pose*.
- **Localization** is a particular form of state estimation where the aircraft is simply locating its position within some known map or environment
- **Perception** is a navigational system's ability to use sensor inputs to build a class-based model of the surrounding environment. This involves delineating objects based on their characteristics. For instance, recognizing that a surface is a floor, and not a wall, would be part of perception.
- **Situational Awareness** is the process of using perception to conclude something about the surrounding environment and project what the environment will be like in the future.

B) Guidance System

The goal of a guidance system is to replace the cognitive decision-making done by a pilot during mission-based flight. It uses embedded software to exercise planning and decision-making regarding the aircraft's movement and trajectory. The guidance system works dynamically with the navigation system to obtain necessary information to make decisions. The components of the guidance system are:

- **Path Planning** is the high-level task of determining the location that the robot would like to travel. This can either be from onboard mission logic (i.e. deliver this package to this address), or could be a more algorithmic task like environmental coverage, where the UAV must autonomously choose new areas to explore with the goal of fully mapping its surroundings.
- **Trajectory Generation** is the process of computing a set of feasible trajectories to get the robot from its starting state, to a goal region. Generally, the trajectories are usually for short-range objectives between two waypoints within a larger path. Trajectories are different from paths in that they are constrained by the capabilities of the robot – i.e. input control or actuator limitations. Trajectory generation outputs specific control inputs necessary to smoothly, and safely move between two points.

C) UAV Dynamics Representation

To make inferences about the aircraft, we must be able to represent it in a logical way. Most simply, we do this by mathematically considering the UAV as a rigid body moving in 3D space. We understand that the UAV has the ability to produce force vectors using throttle, pitch, roll, and also torque vectors by using yaw to rotate about its center of mass. The flight dynamics are generally modeled by using the Newton-Euler equations of motion. We model the amount of thrust and torque acting on the rigid body by using force (F) and moment (M) vectors.

3.2 State Estimation Using Data Fusion

As noted above, state estimation is the process of using sensor data to estimate a UAVs current 3D pose (position and orientation) with respect to a local environment. Each sensor is only operable under certain environments, and each has noise or bias associated with its measurement. Therefore, the aircraft almost always uses multiple sources of information to make the most accurate estimation of the vehicle's 3D pose. The process of combining sensor measurements is called data fusion. Data fusion involves using algorithms to combine current and past sensor data to determine the most probable current location of the UAV.

Most generally, devices use GPS and IMU data to perform state estimation. However, as GPS data is not highly reliable in many scenarios of autonomous flight, secondary approaches combining additional sensor information are generally accepted as being the most reliable.[14,15]

Here, we will examine the most common methods for fusing IMU, Altimeter, and Lidar or Visual data to estimate a UAVs state.

A) Mathematical Representation of the UAV

As noted in [37] we define the world frame, $\mathcal{W}$, is defined by axes x_w, y_w, and z_w, where z_w is pointing upward.[16] The body frame, $\mathcal{B} = [x_B, y_B, z_B]$, is attached to the center of mass of the UAV, with x_B pointed towards the preferred forward direction and z_B perpendicular to the plane created with points at the center of mass of each rotor while the UAV is hovering perfectly above ground. It is important to note that parameters can be converted between the $\mathcal{W}$ and $\mathcal{B}$ frames using rigid-body tranformations.

We define the state of the UAV at some time, t, to be:

$$x_t = [p_t^w, \Phi_t^w, \dot{p}_t^b, {}^a b_t^b, {}^g b_t^b, {}^a b_t^w]^T$$

where $p_t^w = [x_t^w, y_t^w, z_t^w]$ is the UAVs 3D position in the world frame at time *t*. $\Phi_t^w = [\psi_t^w, \theta_t^w, \phi_t^w]$ represent the yaw, pitch, and roll Euler angles that represent the 3D orientation of the body in the world frame. These angles allow us to compute the rigid body orientation matrix, R_t^w, which can be used to convert a vector from the body frame at time *t* to the world frame. $\dot{p}_t^b$ is the 3D velocity vector in the body frame. The parameters with b represent bias of a certain instrument. Because gyroscope and accelerometer biases drift over time due to changes in the temperature of the sensors from varying environments, we include them as part of the state vector so they can be estimated as the vehicle flies. Here ,${}^g b_t^b$ represents the bias of the gyroscope in the body frame,${}^a b_t^b$ represents the bias of the accelerometer in the body frame, $, and\ {}^a b_t^w$represents the bias of the altimeter in respect to the world frame.

A) Kalman Filter

A Kalman filter is a linear estimation algorithm that uses inputs of estimates that are assumed to contain noise that can be modeled using a stochastically. It was first proposed by Dr. Rudolf Kalman in 1960.[17] The algorithm operates on streams of input data, which we assume to be noisy, to produce a best statistical guess of the true underlying system state.

The algorithm works by averaging a prediction of the system's position with a new measurement using a weighted average assigned from the covariance of the measurements. The result of the weighted average is a new state estimate that lies somewhere between the estimated and measured state. This process is repeated every time step, with the new prediction and its corresponding covariance feeding back into the prediction used in the following iteration. Therefore, the Kalman filter is a recursive algorithm that only requires the last estimate rather than the entire history of estimations to predict the new state.

The algorithm works recursively to estimate the state of a linear dynamic system (which UAVs are not) by using a series of presumably noisy measurements. The Kalman filter assumes the true state at time *t* to be based on the state at time *t-1* according to:

$$x_t = F_t x_{t-1} + B_t u_t + w_t$$

x_t is state vector containing state information like velocity and orientation as defined above. The vector u_t is the control vector containing inputs from the systems actions like throttle, or yaw. F_t is the state transition matrix which propagates the effect of the previous state vector in *t-1*, like velocity or orientation, to the new state vector. B_t is the control input matrix which applies the effects of the control vector (like throttle settings) onto the state vector at time *t*. The vector w_t contains the process noise terms for each parameter in the state vector x_t. The process noise terms are assumed to be zero mean multivariate distribution with covariance Q_t, as in $w_t \sim N(0, Q_t)$.[18,19] Measurements of the system are performed by :

$$z_t = H_t x_t + v_k$$

Where z_t is the vector of measurements, H_t is the transformation matrix which maps state vector parameters into the measurement domain, and v_t is the vector containing observation noise for each measurement in z_t which is assumed to also be zero mean Gaussian distributed such that $v_t \sim N(0, R_t)$.

It is important to note that these filters are modeled on a Markov Chain, and that the noise may not necessarily be Gaussian distributed. Nonetheless, we apply this Gaussian assumption because it considerably simplifies the math to derive the filter, and is usually a valid approximation.

The basic process of a Kalman filter is described in the figure below with time represented as *k* instead of *t*:

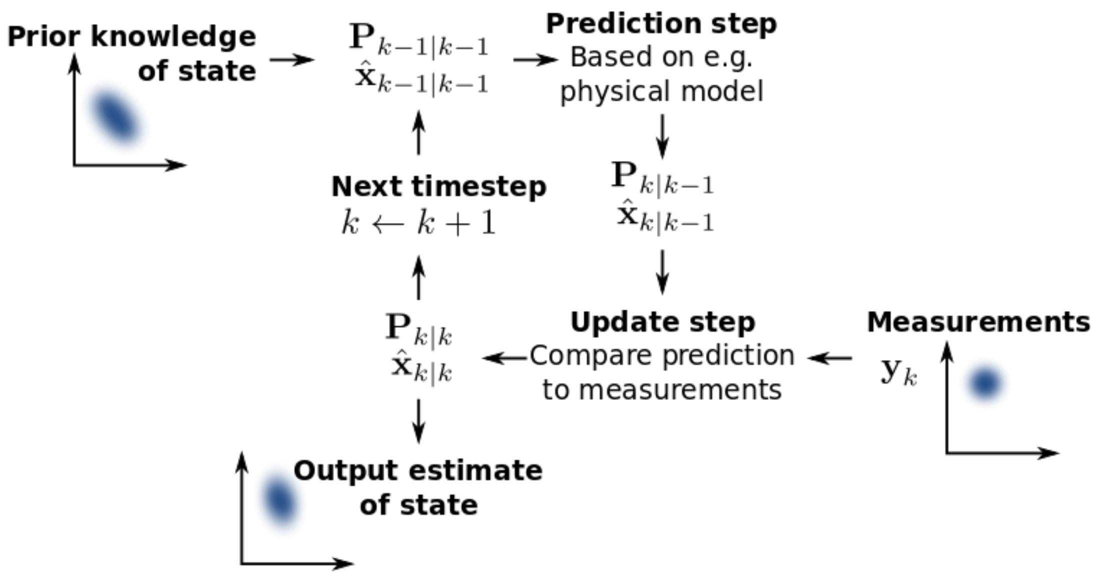


FIGURE 8: OVERVIEW OF KALMAN FILTER PROCESS

Because the Kalman filter only applies to linear systems, and estimates are only based on the most recent robot state, it is sub-optimal to use for state estimation in UAV. UAVs have non-linear dynamics, and information from past states Therefore, we use slightly modified versions of the algorithm to try and better represent the non-linear dynamics of a UAVs state.

B) Extended Kalman Filter (EKF)

The most common filter used to estimate an aircraft's 3D pose is the Extended Kalman Filter (EKF). Specifically in UAVs, we implement a discrete time, nonlinear discrete time function $x_{t+1} = f(x_t, u_t, w_t)$. Where x_t is the state vector, u_t is the control vector, and w_t is the process noise that is drawn from a zero mean multivariate Gaussian distribution with covariance Q_t. The predict function, f is used to compute the predicted state by using the previous estimate. The EKF tracks the state at time *t* as a Gaussian distribution with mean μ_t and covariance Σ_t.The predicted measurement follows the equation $z_t = h(x_t) + v_t$, where h is the update function that estimates the 6-DOF pose of the UAV predicted state, and $v_t \sim N(o, R_t)$.

The computation complexity of a EKF comes with how the mean and covariance are propagated forward starting at *t* = 0:

$$\bar{\mu}_{t+1} = f(\mu_t, u_t, 0)$$
$$\bar{\Sigma}_{t+1} = A_t \Sigma_t A_t^T + W_t Q W_t^T$$

where $\bar{\mu}$ and $\bar{\Sigma}$ represent the quantities before a measurement has occurred, and A_t and W_t are the partial derivatives of *f*.[20] This computation of partial derivatives involves the computation of Jacobian matrices, which can sometimes be mathematically complex. Therefore, when derivation may be difficult, or system dynamics are highly non-linear, many use a separate version of the Kalman filter, detailed below.

C) Unscented Kalman Filter (UKF)

Though computationally more expensive, when a system is assumed to be highly non-linear, an Unscented Kalman Filter has proven to be far more accurate than its EKF relative.[21] Furthermore, the UKF does not require the calculation of partial derivatives, so its implementation is mathematically simpler than the EKF. An unscented transformation is a way for calculating the statistics of a random variable that undergoes a nonlinear transformation by observing that it is easier to approximate a probability distribution than an arbitrary nonlinear function. So, from a high level, the UKF uses an unscented transformation function which, given a non-linear function, chooses a minimal set of sample points called *sigma points*, around the mean. It then propagates these sample points through the process model, and computes the predicted mean and covariance using exclusively these sample points. It then predicts the state vector and updates in a way similar to the EKF.

D) Out Of Order Measurements – Priority Queue Structure

When using multiple sensors, it is possible, and common, to have measurements arrive out-of-order to the data fusion filter. For instance, if a measurement from an earlier state arrives to the computational device where the filter is applied after a measurement that was taken later, problems can occur. This situation violates the Markov chain assumption of the Kalman filter. Additionally, sensors sometimes take time to process their measurements, and therefore can arrive lag in their arrival to the filter.

As measurement data usually contains a few bits containing a timestamp, we can solve this problem by using a priority queue structure as suggested by Shen in [41]. The priority queue is structured with the "oldest" measurement at the top of the stack, and the newest at the bottom. By defining a maximum allowable sensor delay t_d (suggested to be 100ms in literature), we discard all newly arrived measurements that have $t > t_d$ from the current state. After each state propagates, we look at the priority queue and process all measurements that have $t < t_d$.

3.3 Simultaneous Localization and Mapping (SLAM)

During autonomous flight, UAVs cannot be assumed to have accurate maps of their environment, as the world is constantly changing and the aircraft will often venture into previously unmapped spaces. Additionally, as described above, a UAV will never have perfect information about its pose (position, velocity, orientation) in the 3D world due to sensor noise and operating regions. So to ensure safe operation in the National Aerospace System (NAS), autonomous UAVs must be able to use onboard sensors to construct highly accurate maps of their observed environments, and simultaneously localize the aircrafts pose within these maps. This process is referred to as simultaneous localization and mapping (SLAM).

SLAM is often considered a "chicken-and-egg problem due to simultaneous requirement of both mapping and localizing within an environment. It is relatively easy to locate and identify a UAVs pose in a known environmental map, and it is simple to map an environment given a UAVs pose; however, simultaneously estimating the map and localizing itself relative to the map is a far more complex process. The process must exclusively use sensor measurements, $z_{1:t}$, and control inputs, $u_{1:t}$, to build the map and localize the robot.

Generally, SLAM is achieved by fusing together measurements from a variety of sensors including the IMU, GPS, laser scanner (lidar), stereo camera, monocular camera, or RGB-D camera. As each sensor has unique characteristics that lead to varying levels of effectiveness in different environments, the most accurate SLAM techniques generally involve fusion from both visual and laser based scanners. For instance, because laser scanners provide a set of distances of surfaces from the scanner, they have difficulty producing maps in homogenous building structures and can only generate 2D slices of the environment because they cannot make use of structure outside the sensing plane. In contrast, camera sensors measure the intensity of light falling into the 2D sensor plane and can make use of the full 3D environment surrounding the UAV. However, the images produced from visual sensors cannot independently identify the structures in the 3D image, and must rely on feature tracking algorithms to infer structure from image data. Additionally, visual sensor data is known to have a more limited angular field of view, and are more computationally expensive to process.[22]

SLAM problems possess both a continuous and discrete component to inferring the relationship between detected features of the environment, a process called *correspondence*. Through correspondence, the UAV can infer the relative motion of the vehicle over time (a process called *odometry*) in addition to more accurately mapping the environment around it. The continuous component involves locating the objects in the UAVs map and the UAVs 3D pose variables. This is done through feature detection algorithms for either range-based or visual-based sensors. The discrete component has to do with correspondence of the objects. When a feature-detecting algorithm detects an object, the UAV must determine the relation of this object with previously detected objects. This reasoning is discrete because the identified object has either been seen before or it hasn't.

From a probabilistic perspective, there are two forms of SLAM that can be achieved. The first involves estimating the posterior probability over the UAVs pose and map is known as *online* SLAM[23]:

$$p(x_t, m, c_t \mid z_{1:t}, u_{1:t})$$

where c_t is the vector of correspondence variables, x_t is the UAV state vector at time t, and *m* is the current map of the environment. This version of SLAM is called *online* because it only involves the estimation of the pose and map that occur at time *t*. Generally, online SLAM algorithms discard past control and sensor measurements once they have been processed by the algorithm.

The second version is called the *full SLAM problem*. In full SLAM, we seek to estimate the posterior over the map and the entire UAV path, $x_{1:t}$, instead of just the current pose x_t. Additionally, we use all correspondence variables over the path, $c_{1:t}$, instead of the current correspondence of objects tracked. Here, the goal is to estimate:

$$p(x_{1:t}, m, c_{1:t} \mid z_{1:t}, u_{1:t})$$

Each has slightly different algorithms, but the online SLAM is generally thought of as simply being the result of integrating out all past poses of the aircraft and summing over all past correspondence from a full SLAM estimation:

$$p(x_t, m, c_t \mid z_{1:t}, u_{1:t}) = \int \int \dots \int \sum_{c_1} \sum_{c_2} \dots \sum_{c_{t-1}} p(x_t, m \mid z_{1:t}, u_{1:t}) dx_1 \, dx_2 \dots dx_{t-1}$$

In both cases of SLAM, calculating this full posterior distribution is the end goal of the process. The full posterior captures all the known information about the UAVs environment, pose, and path. However, in practice, calculating this full posterior is impossible. This is because of the high dimensionality of the continuous parameter space, as state-of-the-art feature detection algorithms create maps with tens of thousands of detected features. Additionally, because correspondence between features is unknown, the number of possible assignments to the vector of all correspondence variables grows exponentially with time. Therefore, SLAM algorithms must use approximations to deal with correspondence problems.

A key process of SLAM is the ability for an algorithm to discern features or objects from the sensor data. This method of feature detection and tracking differs between visual and laser based sensors.

A) Visual Feature Detection and Tracking

These feature-detection algorithms search for unique points that are likely to be matched well in future images. A local feature is defined to be an image pattern that differs from its immediate neighborhood in terms of intensity, color, and texture. Algorithms generally define the points as either a corner, a point of intersection between two or more edges, or a blob, an image pattern that differs from the image neighborhood in intensity, color, or texture.[24] Literature has introduced a number of point-feature detectors. These algorithms are usually classified as either corner detectors, or blob detectors.[24] Popular corner detectors include the Harris[25], Shi-Tomasi[26], and FAST[27] algorithms. Popular blob detectors include SIFT[28], SURF[29], and CENSURE[30]. An overview of these algorithms, provided by [24], is shown below:

	Corner Detector	Blob Detector	Rotation Invariant	Scale Invariant	Affine Invariant	Repeatability	Localization Accuracy	Robustness	Efficiency
Haris	x		x			+++	+++	++	++
Shi-Tomasi	x		x			+++	+++	++	++
FAST	x		x	x		++	++	++	++++
SIFT		x	x	x	x	+++	++	+++	+
SURF		x	x	x	x	+++	++	++	++
CENSURE		x	x	x	x	+++	++	+++	+++

FIGURE 9: OVERVIEW OF FEATURE DETECTION ALGORITHMS

Once the features are tracked, an algorithm is used to create a correspondence between these features. A popular feature tracker, which is available in OpenCV, is the KLT optical flow tracker.[31] The use of the feature tracker allows the robot to calculate correspondence between tracked objects. KLT operates by gradient descent; therefore, it can only estimate correspondence between two very close or similar images. Alternatively, descriptor matching, which works on blob detectors like SIFT or SURF, can be used. Descriptor matching is far slower than feature tracking, but can (theoretically) handle images that are distant in space or time.

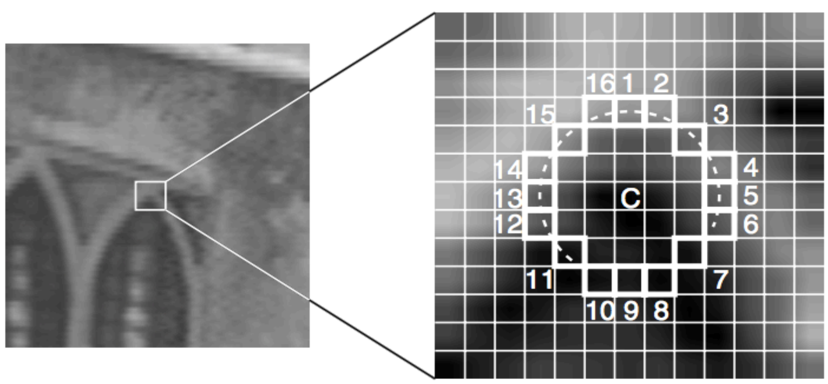


FIGURE 10: EXAMPLE OF CORNER DETECTION

B) Visual Odometry

Odometry is the process of estimating incremental motion of a robot through the use of sensors. This was originally implemented on wheel-based robots to incrementally estimate the motion of the robot by integrating the number of wheel turns over time. UAVs can implement visual odometry (VO), which estimates the UAVs motion and pose through algorithmic examination of sequential camera images to infer changes in motion and orientation. VO estimates the UAV pose by comparing image contents (features) to the map it has created of the environment. VO is considered incremental because it estimates motion in respect to a map, which is constantly updated from new images and past estimations of the UAV pose.[32]

A major challenge to VO involves the minimization of error. Because all sensor data is assumed to have noise associated with its measurements, the map of the environment can be assumed to be gradually acquiring more error as more images are processed. Highly functioning VO minimizes the rate of error accumulation such that the map remains locally consistent, meaning the map is consistent with its immediate neighborhood.

Additionally, components of the map created from VO that are far away in distance or time in which they were discovered can be inconsistent with each other – an error in sensor based mapping known as drift. VO algorithms are usually compared on their ability to minimize drift and sensor error.

Generally, VO algorithms are described as dense or sparse.[33] Dense VO uses as much of the image as possible to compute image intensity and estimate UAV motion. Sparse VO extracts features from the images, as described above, to drastically reduce the size of the input space, making the VO function far less computationally intensive.

Popular implementations of VO include the open source PTAM algorithm[34], and the algorithm presented in Markus Achtelik's Master's Thesis, which combines PTAM with inertial sensors[35].

Generally, VO requires the use of multiple cameras to operate. The images used to map the environment are called keyframes and are defined by the camera the image comes from, as in *left* and *right* frames. VO uses a detection algorithm to identify features within the frames, and then creates correspondence between the left and right frames for each set of images. By creating correspondences between the left and right frames of current and previous images, VO creates a map by using a graph of keyframes with edges denoting correspondence between keyframe features. After correspondence is computed and the graph is updated, we create an estimated 3D map of the space through triangulation. Once we have an estimated location of the features in 3D space, we can estimate the aircrafts pose and position in 3D space with 6 DOF through a least-squares estimation as proposed by Umeyama.[36] A diagram of this process is shown below:

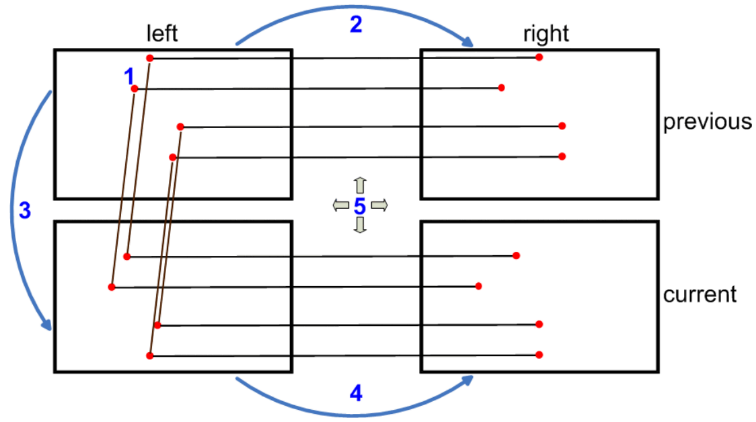


Figure 2-9: Steps performed on each frame in the stereo visual odometry algorithm: 1) Perform feature detection. 2) Find correspondences between *left* and *right* frame for depth reconstruction. 3) Find correspondences between the previous and current frames. 4) Repeat step 2 on current frames. 5) Frame to frame motion estimation.

FIGURE 11: OUTLINE OF RELATIVE MOTION ESTIMATION PROCESS

An overview of the VO process of determining relative position and pose estimation is described in the diagram below:

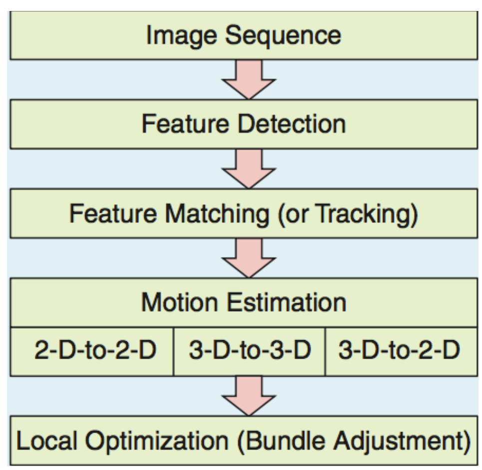


FIGURE 12: VO PROCESS OF MOTION ESTIMATION

C) Laser-Scan Matching for Relative Position Estimation

As described above, laser range finders work by emitting beams of laser light and measuring the time it takes for the beams to bounce off some surface and back to the sensor. These sensors allow us to infer the distance between the sensor and the nearest obstacle in the direction of the sensor. As described in [[22]], by sweeping the sensor in a circular motion, we can create a scan of the environment that contains the range of obstacles surrounding the UAV. Using Cartesian coordinates, the scan creates a set of points that represent distances to surrounding obstacles:

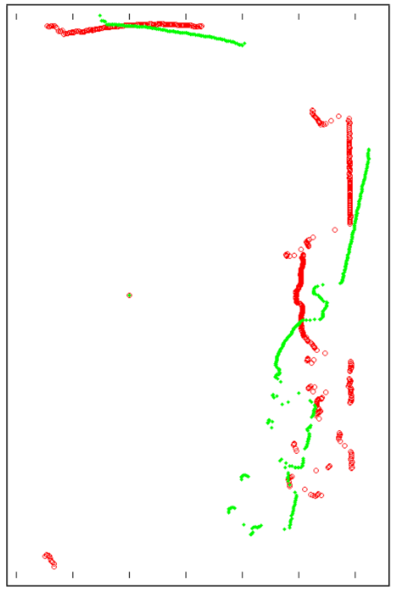


FIGURE 13: EXAMPLE OF SET OF POINTS FROM LASER SCAN

To determine the relative position and pose of the UAV, laser-scanning algorithms are given successive overlapping scans and tasked with determining the rigid-body transform that, when applied to applied to the older scan, results in a scan very similar to the newest scan. However, sequential scans will generally not measure the same points in the environment due to the moving objects in the environment. Therefore, scan-matching algorithms must be used to create correspondence between points found in scans.

The most basic of these algorithms is the Iterative Closest Point algorithm (ICP) proposed by Zhang.[37] It alternates between finding correspondence between individual scan points, and finding the optimal rigid body transform that minimizes the Euclidian distance between points. The basic algorithm is outlined below:

Require: S_1 and S_2 (the scans to be matched)
Require: Δ (initial guess of the transformation)
while Δ not converged **do**
$\hat{S}_2 = \Delta \otimes S_2$ (project S_2 using current transform)
for $\mathbf{x}_i^1 \in S_1$ **do**
$$\mathbf{y}_i = \underset{\mathbf{x}_i^2 \in \hat{S}_2}{\operatorname{argmin}} ||\mathbf{x}_i^1 - \mathbf{x}_i^2||_2$$
end for
$$\Delta = [R, t] = \underset{[R,t]}{\operatorname{argmin}} \sum_{i=1}^{N} ||R\mathbf{y}_i + t - \mathbf{x}_i^1||_2$$
end while

FIGURE 14: ITERATIVE CLOSEST POINT (ICP) ALGORITHM

As computing the correspondence between points can be challenging and inaccurate, map based probabilistic approaches are more commonly used. These involve creating an occupancy grid map, M, of the previous scans, and matching incoming scans against the map. Each cell in the map contains the likelihood of the *ith* laser measurement return measured at point x_i as:

$$P(x_i \mid M)$$

The map allows one to compute the likelihood of an entire scan by calculating the likelihood that all point measurements are accurate. The likelihood of an entire scan can then be computed as:

$$P(S \mid M) = \prod_{i=1}^{N} P(x_i \mid M)$$

By searching through all possible rigid body transforms, we can deduce the transform that provides the most optimal alignment of the next laser scan. This approach is used in the Vasco scan-matching solution provided by the Carmen robotics toolkit.[38] A 2D cross section of a 3D pose likelihood map is shown below, where local peaks are locations where correspondence is maximized, and valleys are locations in the parameter space where correspondence is poor:

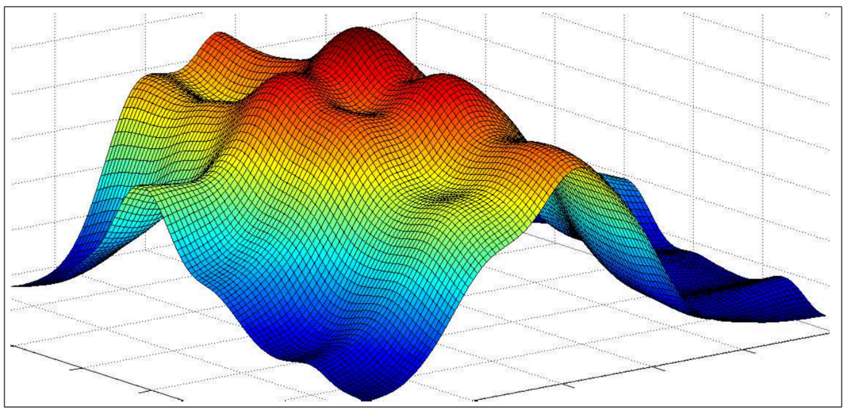


FIGURE 15: CROSS SECTION OF POSE LIKELIHOOD MAP

Though successive scans will not return readings from the same point, they usually will return readings from points on the same surface.[39] Using this logic, many model the surrounding environment of the UAV using a set of polyline contours. The algorithm for creating these maps iteratively connecting endpoints of possible contours from laser scans until no more endpoints exist that justify some “joining constraints” outlined in the algorithm. The contour map is then transformed into a likelihood map to describe how likely each cell is part of a contour.

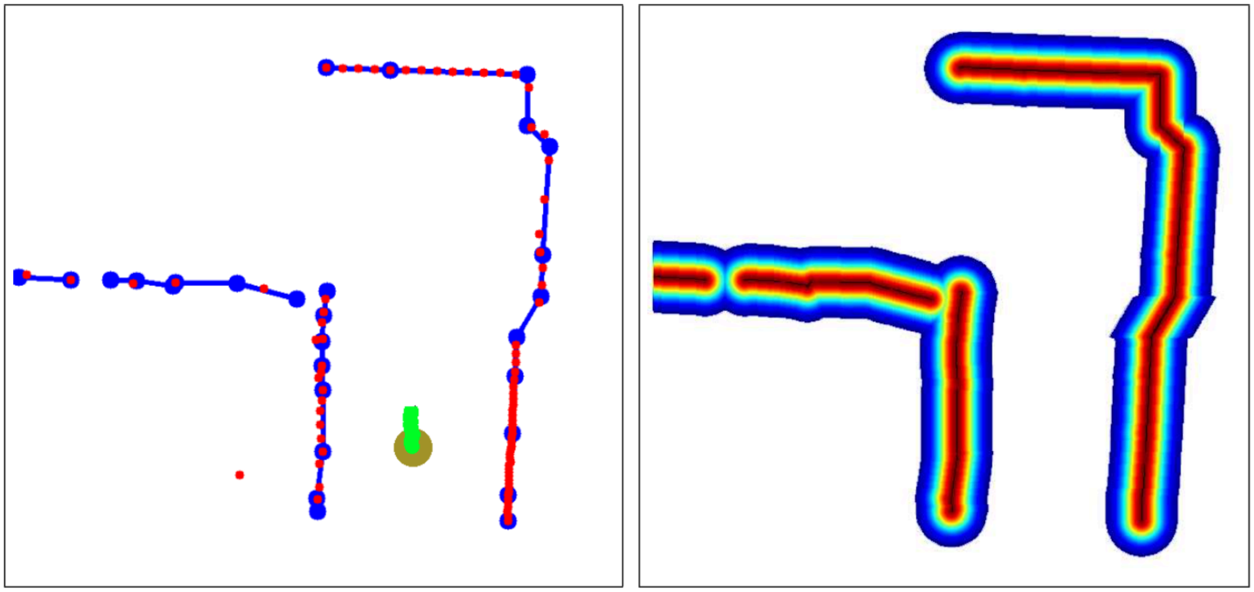


FIGURE 16: EXAMPLE OF CONTOUR MAP AND CONTOUR LIKELIHOOD MAP

Finally, with estimates of the aircrafts pose and position in 6-DOF, and successive sweeps of the laser sensor and features detected from the VO operation, SLAM algorithms usually output a 3D point cloud that represents the environment with features and surfaces represented as points in the map.[40] An example is shown below:

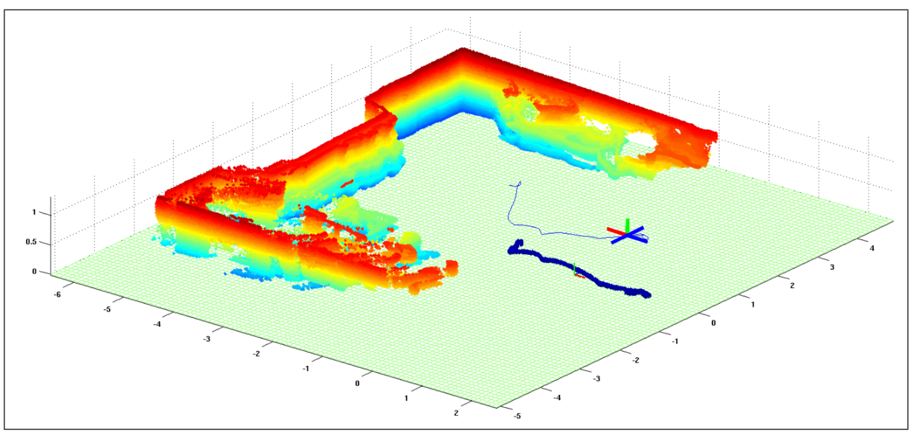


FIGURE 17: 3D POINT CLOUD CREATED FROM LASER & VISUAL SLAM

D) Environmental Mapping:

Key to SLAM, path planning, and obstacle avoidance, is the ability to represent the surrounding environment of the UAV. Though some map types are useful for the localization of the robot, such as point clouds or contour maps, path planning and obstacle avoidance require more detailed maps that include mapping of free space. The traditional approach to achieve full environmental coverage is frontier-based exploration (FBE).[41] Most frontier-based exploration utilizes the occupancy grid approach. Given a probabilistically correct UAV path and pose, the basic idea of an occupancy grid is to represent the map of the environment as a field of random variables arranged in an evenly spaced grid. Each variable is binary, and denotes the occupancy of the location of the grid point. Though SLAM techniques do not always utilize occupancy grid approaches to representing the environment (as discussed above) they are incredibly useful in post-processing of the pose, position, and environmental data found during the SLAM process.

Again, the goal of an occupancy map is to produce the posterior over the map's cells, m_i, given the input data:

$$p(m_i \mid z_{1:t}, x_{1:t})$$

where $z_{1:t}$ is the set of measurements up to time t, and $x_{1:t}$ is the path of the robot. Each cell is binary, with 1 denoting the space is occupied, and 0 denoting the space is free.

Most occupancy grid algorithms are only able to map the surrounding environment in 2D. However, UAV motion is clearly a 3D problem, so more advanced maps must be used to correctly represent the surrounding environment.

Though many map types have been proposed in literature to represent the UAV in 3D, including the Octree structure[42], multi-volume occupancy grids (MVOC) stand out as particularly interesting as they typically have memory cost on the same order as a 2D occupancy grid.

A MVOC consists of a 2D grid of square cells lying in the xy-plane, with each cell containing two lists of volumes, one containing obstacle readings and the other containing free space readings.[43] Each volume within the two lists is defined with the height of its bottom and top face, and the occupancy mass. The occupancy density can be computed for each volume using the three parameters, and corresponds to the amount of sensory information the volume has received. The positive volume list contains readings for a specific grid that indicate an obstacle, while the negative volume list contains readings that indicate free space. An example of the two volumes for a single cell is shown below:

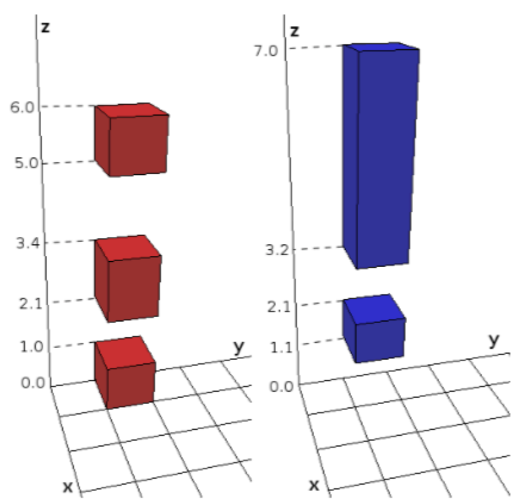


FIGURE 18: POSITIVE AND NEGATIVE VOLUMES FOR A SPECIFIC CELL

MVOG is created by taking in sensor information such as point clouds from laser scans, and generating the volume lists for each cell in the grid. Then, using the list of volumes for each cell, the probability of each cell being occupied is computed, and a resulting graph containing a

3D probabilistic representation of the environment is produced. An example of this transformation from a point cloud representation to a MVOG is shown below:

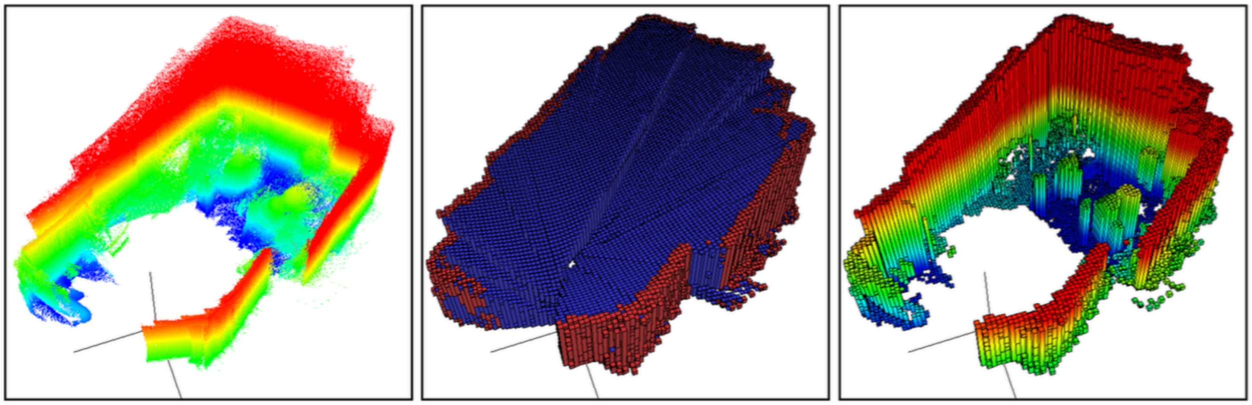


FIGURE 19: CREATION OF MVOG FROM POINT CLOUD ON LEFT, TO VOLUME LISTS IN MIDDLE, TO PROBABILISTIC 3D MVOG ON RIGHT

E) SLAM Algorithms

Many SLAM algorithms have been proposed in literature, and we will not cover the specific mechanics of how they operate here. In general, they work to fuse noisy sensor data to both locate the UAV and map the surrounding environment. SLAM is widely implemented in practice using PTAM algorithm proposed by Klein, which introduce the idea of dividing the tracking and mapping components of the process into separate thread to increase run-time. PTAM is popular due to its open-sourced availability on Github.[44] Other popular implementations include the GMapping algorithm[45] which is also available online through OpenSLAM.[46] Most SLAM techniques use EKF filters to fuse IMU, visual, laser, and other sensor data into the SLAM process. In general, the maps produced by SLAM are generally not detailed enough for precise path planning, so localization information and path data are often fused with sensor data to create more robust 3D maps of the environment for trajectory planning. A high level diagram of how visual, laser, and IMU sensors are fused to perform SLAM with EKF data fusion filtering to fuse pose and map estimates is shown below:

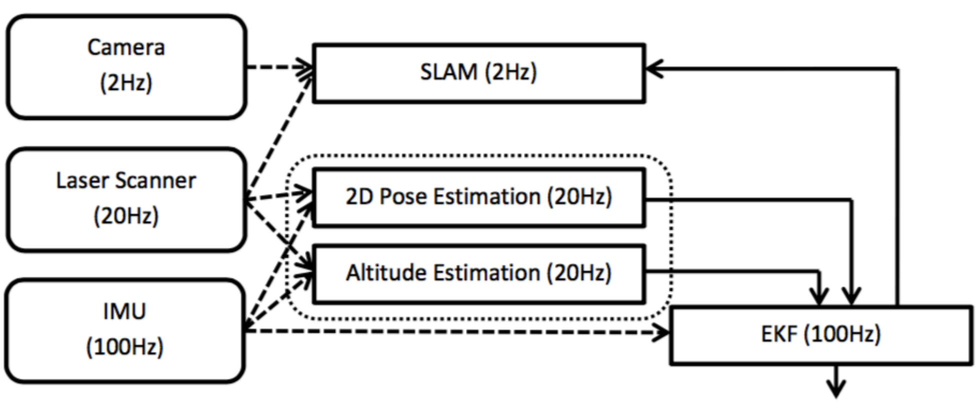


FIGURE 20: DIAGRAM OF SLAM FUNCTIONALITY

3.4 Trajectory Planning

Central to autonomous UAV operation is the ability for the aircraft to independently generate a collision-free trajectory from some starting point to a desired region. Using state estimation, relative position estimation, and mapping procedures outlined above, we will explore how UAVs determine optimal flight trajectories and avoid obstacles.

A) Basic Path Planning Algorithms

Most basic path planning algorithms utilize occupancy grids created by SLAM or known environmental maps. In the most basic sense, path planners are given a goal cell that the UAV must move to, and the planner computes a shortest path using cells that are probabilistically "free" as nodes in a graph. They utilize well-known algorithms for computing the shortest path from a source node to another node in a weighted graph. These algorithms include those proposed by Dijkstra[47] and Bellman-Ford[48] to find the shortest path given the known graph of the free space between the starting point and the ending point.

However, these algorithms are computationally expensive with asymptotic bounds in the range of $O(n^2)$ for a given number of input nodes, n. Additionally, they require expensive recalculation when new information is added to occupancy grids, and the preprocessing of the graph to classify "free" cells based on the flight capabilities or safety of the UAVs trajectory make them less intriguing. Therefore, more recent algorithms to explore possible path planning have been introduced.

B) Single-Query Sampling-Based Motion Planners

From a high-level, path planning involves defining a goal state for an aircraft, and generating a safe path for the UAV, which is often assumed to be a cylinder in 3D space, to travel from a starting state to a goal state. The path continuously connects the UAV from the starting state of the aircraft to a desired goal state by outputting a series of desired 3D positions and yaw angles for the aircraft. It does this under a set of constraints that define how the aircraft can move, and how close it can get to obstacles.

In recent years, single-query sampling-based planners have become very popular due to their ability to quickly solve motion-planning problems.[49] Most popular are the class of tree planners that iteratively grow a tree, rooted at the start state, of motions in the state space of the UAV using different heuristics. Specific to this tree structure is the process of iteratively attempting to extend the tree with a new path segment, a valid motion between aircraft states, towards a new state. Trees are advantageous over their alternative, graphs, because the order of states along a path can be used to provide time delineation of the states in the path. This is extremely important when we convert the solution path into a set of control inputs to the UAV that would allow the aircraft to actually achieve the specified path. An example of a tree is shown below:

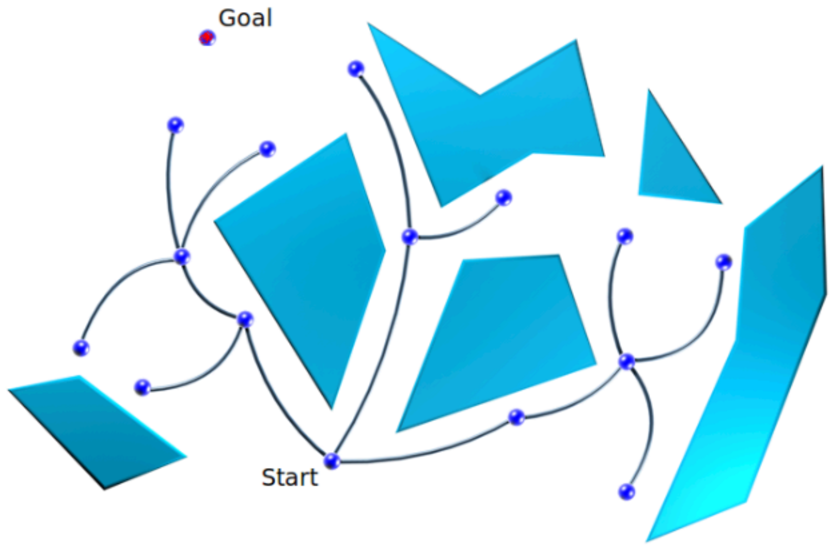


FIGURE 21: EXAMPLE OF TREE OF MOTIONS BETWEEN STATES. NODES REPRESENT AIRCRAFT STATES, AND EDGES REPRESENT MOTION BETWEEN STATES

The theoretical goal of all tree planners is to grow the tree of motions such that it eventually covers the entire state space. However, since we are only interested in optimal motions that allow us to move from a starting state to a state in a specified goal region, total coverage is not required for the tree planner to be effective.

The problem starts by defining the state space, the set of states the UAV could take, as χ. It then defines the starting state $x_{start} \in \chi$, a set of states in the "goal region" $\chi_{GOAL} \subset \chi$, and an allowed time to search the state space, *t*. Additionally, from the map of the environment given to the algorithm, there is an input function that determines if a state in the state space is colliding with an obstacle, meaning that it is part of the obstacle state space $\chi^{obs} \subset \chi$, or if it a specified UAV state would not result in a collision and is part of the free state space $\chi^{free} \subset \chi$. The free and obstacle state spaces are not given to the algorithm, instead the algorithm continuously uses the function to determine if states are in free space or obstacle space, as the map in continuously updated during aircraft motion, and recalculating these space would be time intensive (as with what needs to be done in basic path planning algorithms above). It is also important to note that parts of the state space that are presumably "free" can be defined as part of the obstacle space if there are *geo-fenced* areas present. Geo-fencing is common in modern civil UAV usage to restrict aircraft from flying into restricted areas such as airports or government buildings. Nodes in the tree can only be possible UAV states that are an element of χ^{free}. Additionally, the algorithm is given a set *U* of all possible controls that can affect the UAV state.

A path in χ is defined as a continuous function $\sigma: [0, s] \rightarrow \chi$ where s is the length of the path. The set of all paths in χ with non-zero length is defined as $\sum_{\chi}$. The basic algorithm for building a tree with nodes in a specified goal region $\chi_{GOAL} \subset \chi$ is defined as:

```
INIT(T, 𝒳, s)        // unless T already initialized
while ELAPSEDTIME() < t and NOGOALFOUND(𝒢) do
   x_tree ← STATETOEXPANDFROM(T)
   p_add ← PATHTOCONSIDER(x_tree)
   if CHOOSETOADD(p_add)  then
      INSERT(T, p_add)
   end if
end while
return T
```

FIGURE 22: BASIC TREE BUILDING ALGORITHM

Given this tree, algorithms can determine feasible solutions by determining the paths that result in the final UAV state being in the goal region. However, the goal of path planning is not simply to get from point A to point B safely, but rather to move as efficiently as possible as UAVs have tightly constrained flight time due to battery life. Therefore, the goal of these algorithms is to sample UAV states enough times such that the low-cost paths can be found.

The cost function c defines the cost of the path defined using some performance metric that includes time, path distance, closeness to obstacles, or fuel consumption. The goal of optimal planning within this tree space is to find a path $\sigma^* : [0, s]$ such that $s \in \chi_{GOAL}$ and the cost of the path is the smallest of all feasible paths to the goal region: $c(\sigma^*) = \min(\sigma \in \sum_{\chi_{GOAL}} c(\sigma)$. It is important to note that in practice, the UAV will not stop the tree planning when a feasible solution is found, instead it will continue to sample the space in hopes of finding an optimal solution given some cost function. As more samples are added to the tree, the chance of finding the most optimal solution increases.

The first accepted implementation of this tree structure is the rapidly-exploring random belief tree (RRT) proposed by LaValle.[50] The key idea in RRT is to iteratively sample new states but adding bias to the exploration by pulling the search tree towards newly sampled nodes. This pulling of the search tree is often referred to as the “steering function” of the algorithm. The RRT is useful as it has probabilistic completeness, and exponential decay in the probability of failure in finding a feasible path with the number of samples.

Recently, an extension of the RRT algorithm, called the Rapidly-Exploring Random Graph (RRG), was proposed by Karaman and Frazzoli.[51] RRG is similar to RRT in that it connects the “nearest” samples in addition to connecting new samples to ever node within some “ball.”[52] This results in a connected graph that not only rapidly explores the state space, χ, but also is locally refined with each added sample. This refinement results in the RRG containing all possible paths through the environment given the constraints, with a large number of samples. This means that the RRG has the property of converging to the optimal solution given enough samples. The RRT* algorithm is the tree version of the RRG and exploits its property of converging to the optimal path by only keeping edges in the graph that result in lower cost at vertices within the ball. RRT* is frequently used for motion planning, and is available in an open-source implementation through the Open Motion Planning Library.[53]

To see the differences between RRT and RRT* in the same environment with different sample sizes, refer to the following figures:

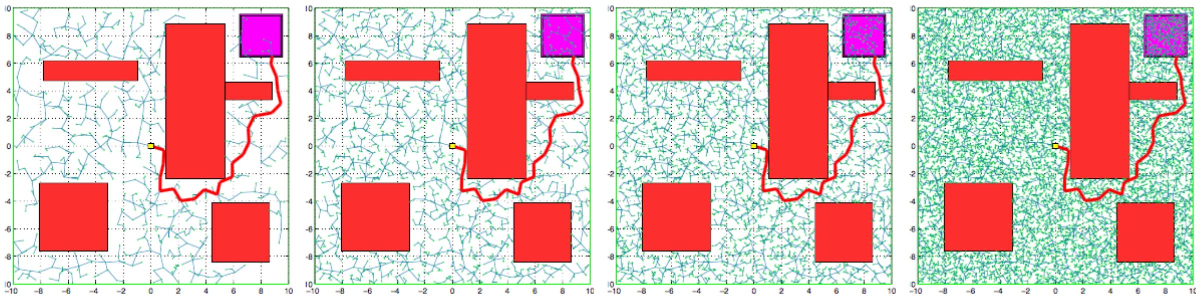


FIGURE 23: RRT AT 1000, 2500, 5000, 15000 ITERATIONS

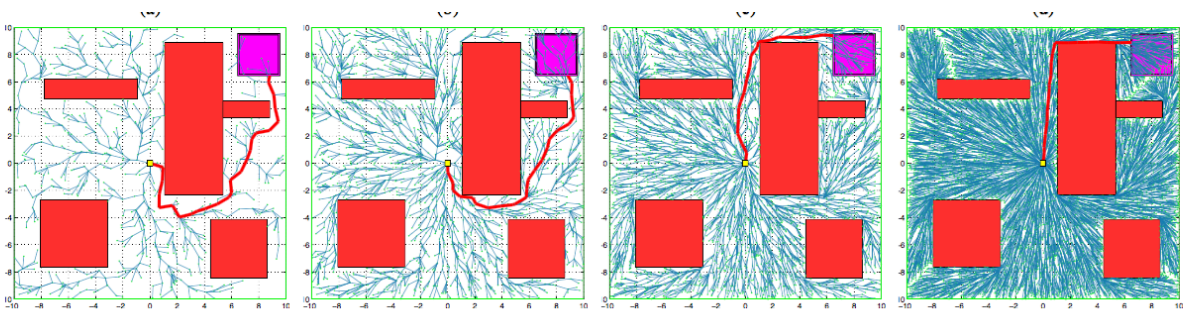


FIGURE 24: RRT* AT 1000, 2500, 5000, 15000 ITERATIONS

However, these algorithms assume state certainty, but as noted above, we can never be certain of the aircraft's state and pose. Instead, we use stochastic solutions, like Kalman filters, to estimate the aircraft's state. To solve for this problem, Adam Bry and Nicholas Roy introduced the Rapidly-Exploring Random Belief Tree (RRBT) to extend the RRT* algorithm to handle uncertainty in the aircrafts dynamics and measurements.[53] The algorithm is capable of accomplishing motion planning in the presence of state uncertainty, a process known as planning in belief space. RRBT is noted as being the most accurate means for accomplishing motion planning to date. An example of the RRBT algorithm after multiple iterations is shown below:

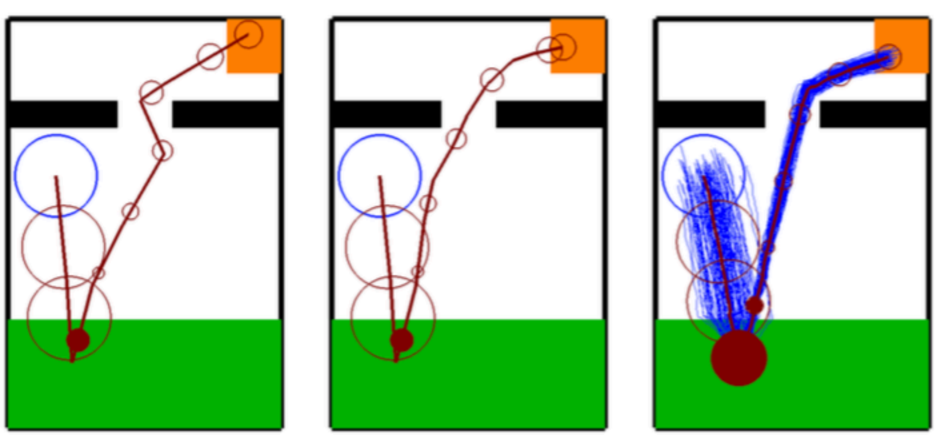


FIGURE 25: RRBT AFTER 100, 500, 10000 ITERATIONS

C) Complete Module Schematic

Pulling it all together, below is an example of a fully integrated flight system that would allow a UAV to operate autonomously while making estimates of its state and pose, avoiding obstacles, and mapping its environment.

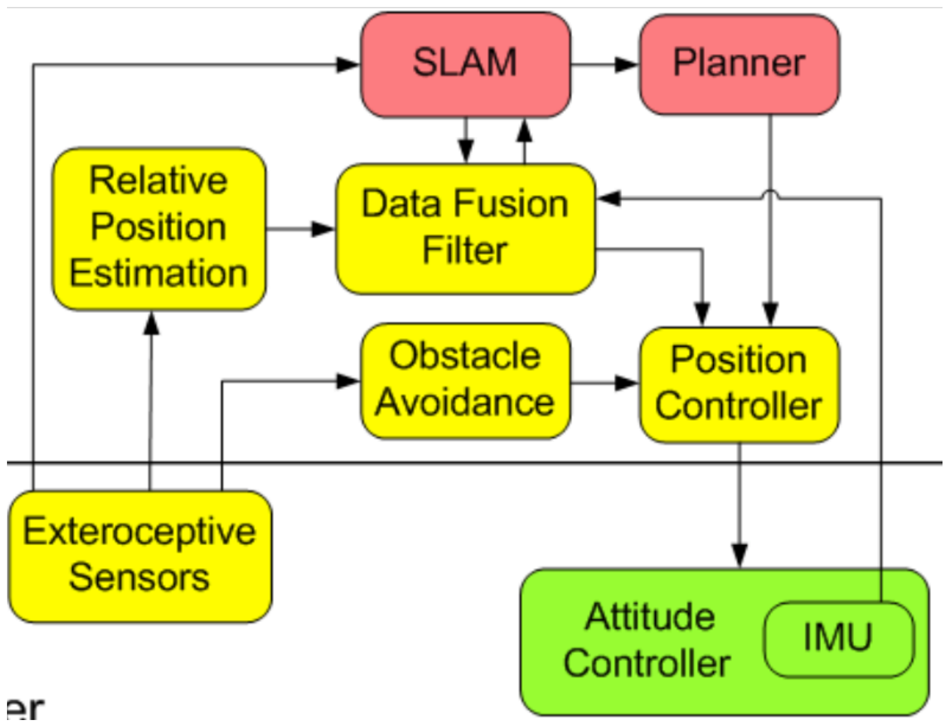


FIGURE 26: SCHEMATIC OF SENSING, CONTROLLING, AND PLANNING SYSTEM

4. Why They Are Interesting Today

Drones have quickly become one of the hottest topics of discussion in today's technology and academic worlds. This is, in large part, because the industry operates in an ever changing, and cloudy regulatory environment, and it has seen a drastic pace of technological innovation, adoption, and price reduction in recent years. It's astounding to compare the capabilities of the expensive early consumer drones of only a few years ago to the cheaper, more advanced models we see today that are driven by robust software and stabilized, ultra-high quality sensors.

These new drones have advanced to their present level for a variety of reasons. These include drastic advances to the underlying hardware in a drone including the ESC (going from analog to digital), electric motor (going from brush to brushless DC motors), and battery (availability of affordable Li-Po). Though we will not cover these advances in depth here, the improvement of these components is imperative to today's drone technology.

Additionally, more tangential factors have played a role. For instance, computing costs have fallen by 33% annually over the past 15 years to a mere $.05 per million transistors, and smartphone enabled devices have rapidly proliferated in the US to a penetration of more than two-thirds of the potential user base.[54] Particularly, smartphone popularity has resulted in extensive innovation and price reductions in many of the same components (camera, IMU, CPU, GPU) that UAVs utilize. Additionally, mobile augmented reality (AR), like Google's project tango,[55] has similar software requirements (SLAM, Visual Odometry, Sensor Fusion) to UAVs, so research efforts have been compounded. These new powerful processors on board the UAV and in the pilot's smartphone have allowed for quick execution of flight-assisting software onboard the drone and beautifully designed applications on the pilot's smartphone that provide critical flight information, capabilities (such as photography), and control to the pilot.

These factors in conjunction with many others have allowed for the rise of the modern civil drone. Here, we will cover a few topics that relate to the economic, social, and ethical impact drones could have specifically in the United States.

4.1 Market Growth and Economic Impact

A) Market Growth

Civil drone sales have exploded in the United States in the past few years. Estimates suggest that at least 500,000 have already been sold in the US, and that the largest manufacturer, DJI, quadrupled their US sales in 2014.[56,57] The FAA, among others, has publically forecasted that within a decade consumer drones will be an $11.5 billion industry annually.[58] These numbers are staggering, especially considering the cloudy legal environment in which these drones can operate.

B) Venture Funding

In 2014, venture capital funding into domestic drone companies doubled from the previous year to $108M across 29 deals.[59] Additionally, a multiple hundred million dollar investment into Chinese manufacturer DJI has been rumored to be nearly closed this year, which would more than double 2014's venture funding into drones by itself.

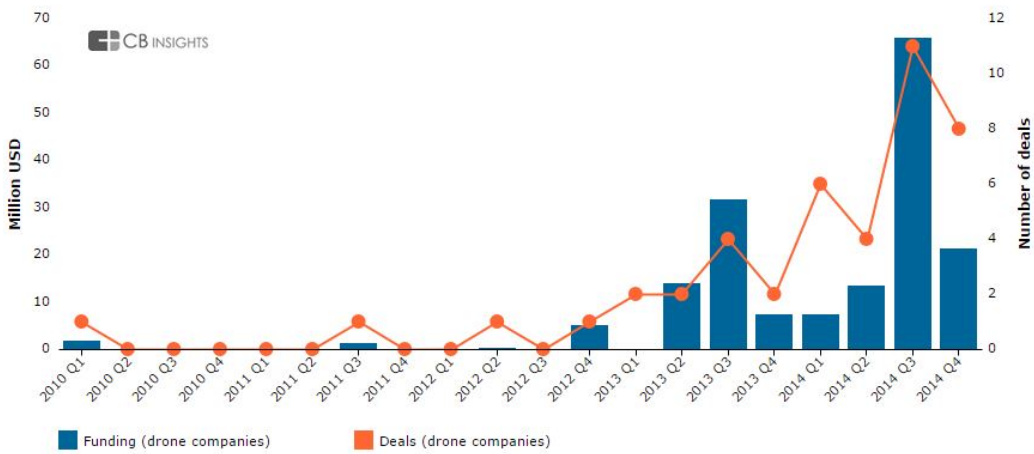


FIGURE 27: US VENTURE CAPITAL FUNDING OVER PAST 5 YEARS

C) Economic Impact

Integration and legalization of UAV usage in the NAS would have a dramatic economic impact. The United States leads UAV research and development efforts with an estimated 65% share of worldwide spending.[60] Researchers also believe that UAV integration would lead to an estimated 70,000 jobs being created, and a $13.6 billion dollar impact on the US economy in the first three years of integration.[61,62] Because of the US's leading role in research and development, and the large appetite for consumer and commercial applications of drones, it is clear that integration could be a welcome spark for both the economy for the US in the long term.

This economic impact could easily change if integration is not made quickly in the US. For the past few years, major companies such as Amazon, Google, and PrecisionHawk have focused their R&D spend outside of the US due to the more accommodating regulatory environments.[63] If integration is not achieved quickly, its clear that the US may be left behind in the drone market. Additionally, further delays would limit the domestic economic impact, as first mover advantages are important in such a technologically complex market.

4.2 Aerial Photography

Drones have democratized access to aerial photography. The best consumer models today have a 4k-resolution camera, effective 3-axis gimbal, and robust flight controller built-in. However, they are remarkably affordable, well within even a hobbyist's budget. All-in (with extra batteries + accessories) the drones cost between \$1,500-\$4,000 to purchase, and less than \$300 to rent for the day. Manned helicopters are suitable alternative for most shots, but at \$500-750 an hour, and \$3500 a day, there is a clear price advantage for the smaller, remote controlled copter.[64]

For hobbyists and professionals alike, capturing a previously unfathomable aerial shots is now an economic and technological reality. Easy to use smartphone apps stream the live camera feed and serve as the controller for the camera. Its simplicity enables even the least tech-savvy consumer to casually fly their stabilized drone over their home, party, sporting event or vacation to capture aerial shots. Additionally, highly functional gimbals allow for professionals to attach their favorite DSLR or camera to a functional multirotor with ease to capture shots from angles and locations that even helicopters cannot safely capture. Even better, robust software can now allow for drones to algorithmically accomplish shooting patterns - such as following a specific GPS signal or shooting a preprogrammed route - that would not be possible with a human operator.

Simply put, drones have unlocked the sky as a vantage point for even the most amateur photographer.

4.3 Logistics

Possibly the biggest market application of UAV technology comes in its ability to quickly transport payloads from point A to point B.[65] This concept is especially relevant given the recent rise of ecommerce (Etsy, Bonobos, Warby Parker) and instantaneous delivery services (Postmates, GrubHub, TaskRabbit, etc) as delivery time and costs are a major differentiator for customers. Jeff Bezos, CEO of Amazon, first introduced the public to the concept of drone logistics when he unveiled his concept for Amazon Prime Air on *60 Minutes* in December of 2013.[66] According to an FAA exemption letter written by Amazon, drone delivery would enable them to deliver packages weighing as much as 5 pounds at distances of up to 10 miles from a fulfillment center.[67] If true, this would enable the ecommerce giant to ship 86% of its packages to customers within 30 minutes of the order being placed.

But can this be economical? Digging into the math behind the cost of operating a drone, we indeed find that these deliveries are not only fast, but also incredibly economical. Drones use electricity to charge their battery for operation, and the industry-standard Li-Po cells used in UAVs today have a tightly bound lifecycle. So, the operating cost of a UAV is simply determined by approximating the variable electricity cost and amortized battery cost over a delivery distance. In 2014, Kiva Systems founder Raffaello D'Andrea estimated the cost of flying a 4kg drone with a 2kg package with a maximum total range of 10km (6 miles) with an industry-standard Li-Po battery and headwinds of up to 30 km/h for drone delivery company Matternet. Using some estimates of drone parameters, D'Andrea concluded the cost to be in the order of 10 cents per 10km delivery.[68] By estimating power consumption (in kW) of the aircraft to be:

$$\frac{(m_p + m_v)v_c}{370\eta r} + p$$

with m_p = payload mass in kg, m_v = vehicle mass in kg, r = lift-to-drag ratio, η = power transfer efficiency for motor and propeller, p = power consumption of electronics in kW, and v_c = cruising velocity of the aircraft in km/h.

Although some suggest that the battery represents no more than one-third of a UAV's weight[69], we will assume the UAV is capable of transporting a 2 kg Lithium Ion battery and a 2 kg package, thus making the total mass 6 kg. Using a conservative lift-to-drag ration of 3, a power transfer efficiency of 0.5, conservative total electronics power consumption to be 0.1 kW (power consumption of modern laptops), and a cruising velocity of 45 km/h, he estimates a total power consumption of 0.59 kW. Given his assumption of a high-end Li-ion with specific power of .35 kW/kg, a 2 kg battery could provide .7kW.

The worst-case energy requirement, *E*, in kWh is defined as:

$$\frac{d}{1 - v_r}\left(\frac{m_p + m_v}{370\eta r} + \frac{p}{v_c}\right)$$

with d = maximum range in km, and v_r = ratio of headwind to airspeed. Assuming the range to be 10 km, and ration of headwind to airspeed to be 2/3, as headwind is 30 km/h and airspeed is 45 km/h, we find an energy requirement of 0.39 kWh. A Li-ion battery with specific power of 0.35 kWh/kg can easily meet this energy requirement.

Economically, the average cost of electricity per km is:

$$\frac{c}{e}\left(\frac{m_p + m_v}{370\eta r} + \frac{p}{v}\right)$$

where c = the cost of electricity in \$/kWh, and e = the charging efficiency of the battery. Assuming the retail cost of electricity to be 0.1 \$/kWh and charging efficiency to be 0.8, we see that the electric cost of operating a drone with a 2 kg payload is around \$0.002 per km.

Estimating the amortized battery cost per km over its lifecycle is a bit less complex. The average battery cost per km is:

$$\frac{k}{l}\left(\frac{m_p + m_v}{370\eta r} + \frac{p}{v}\right)$$

where k = battery cost in \$/kWh, and l = the average number of charge cycles a battery can endure before the maximum charge is <80% of its original capacity. The number of cycles is used to estimate the non-cash expense of storing the energy delivered to the UAV in \$/kWh. It is equivalent to the accounting method of depreciating cars based on the number of miles driven.

Estimating a high-end Li-ion battery to cost $300/kWh, and have a lifespan of 500 cycles, we estimate the "depreciation cost" of the battery to be around $0.008 per km for a 2 kg payload.

Therefore, as stated before, we found the average operating cost of delivering a 2 kg package over a range of 10 km to be in the order of $0.10:

$$\left(\$0.002\left(\frac{electricity\ cost}{km}\right)+\$0.008\left(\frac{battery\ depreciation}{km}\right)\right)*\ 10\ km\approx\$0.10$$

As the battery D'Angelo proposed would not be completely discharged in a 10 km delivery, I estimated how the cost of delivery varies over the delivery radius (assumed to be half the range) and package mass with the above battery configuration and parameter estimates:

Package Mass (kg)	Delivery Radius (km): 2.5	3.75	5	6.25	7.5
0.5	$0.037	$0.056	$0.075	$0.094	$0.112
0.75	$0.039	$0.059	$0.078	$0.098	$0.117
1	$0.041	$0.061	$0.081	$0.102	$0.122
1.25	$0.042	$0.064	$0.085	$0.106	$0.127
1.5	$0.044	$0.066	$0.088	$0.110	$0.132
1.75	$0.046	$0.068	$0.091	$0.114	$0.137
2	$0.047	$0.071	**$0.094**	$0.118	$0.142
2.25	$0.049	$0.073	$0.098	$0.122	$0.147
2.5	$0.051	$0.076	$0.101	$0.126	$0.152
2.75	$0.052	$0.078	$0.104	$0.130	$0.156

FIGURE 30: SENSITIVITY OF DELIVERY COST WITH VIABLE DELIVERY RADIUS & PACKAGE MASS

Especially for light packages, it is clear that drone deliveries within a short radius of a launch site can be incredibly economical. To see the economics of a battery with different cost or lifecycle I created another table keeping the delivery radius at 5 km and the package at 2 kg:

Battery Cost ($/kWh)	Maximum Number of Battery Cycles: 400	450	500	550	600
$150	$0.065	$0.060	$0.055	$0.052	$0.049
$180	$0.075	$0.068	$0.063	$0.059	$0.055
$210	$0.085	$0.077	$0.071	$0.066	$0.062
$240	$0.094	$0.086	$0.079	$0.073	$0.068
$270	$0.104	$0.094	$0.087	$0.080	$0.075
$300	$0.114	$0.103	**$0.094**	$0.087	$0.081
$330	$0.124	$0.112	$0.102	$0.094	$0.088
$360	$0.134	$0.121	$0.110	$0.102	$0.094
$390	$0.143	$0.129	$0.118	$0.109	$0.101
$420	$0.153	$0.138	$0.126	$0.116	$0.108
$450	$0.163	$0.147	$0.134	$0.123	$0.114

FIGURE 31: SENSITIVITY OF DELIVERY COST WITH BATTERY CYCLES AND BATTERY COST

Battery cost per unit of energy and battery lifecycle are each of equal importance to the drone delivery cost. Improvements in either would greatly reduce the cost of delivery.

However, there are a few problems with these estimations. First, unless these drones are landing on charging stations or battery swapping mechanisms (as shown in 6.1) at the location packages are dropped off, the 10 km range essentially limits the radius of delivery to be 5 km (3.1 miles). This is far less than what Amazon proposed, and not nearly wide enough to provide a great logistics solution for a distribution facility, which is usually located in less expensive areas slightly outside of densely populated areas.

Second, Li-ion batteries are no longer the industry standard, and instead have been replaced by Lithium Polymer batteries (as discussed in 6.1). Li-Po batteries can provide higher levels of current to the UAV system ,have higher specific power, and possibly longer lifecycles than Li-ion. However, they are currently more expensive per unit of energy (kWh). It is important to note that Li-Po prices should be reduced in the near term as massive new factories like Tesla's Gigafactory in Nevada, are completed. The new factories will reduce cost by bringing a much-needed boost of supply to the market in addition to cheaper production through economies of scale.[70]

Finally, these cost estimates do not include any allocation per km of the fixed cost of the drone hardware, R&D costs, or support staff necessary to keep the drone in operation. Though the physical cost of the drone hardware is relatively low (somewhere in the $1,000-$10,000 range per drone), the highly trained technicians, engineers, and software developers needed to develop and sustain these systems are expensive. The impact of this overhead on the cost per km of drone delivery is difficult to estimate given the wide range of costs. However, bundled software and hardware packages available from companies like Matternet[71] would reduce the uncertainty regarding support staff and R&D costs. Assuming they charge a price that corresponds to total overhead being in the range of <$1 / delivery, it is clear that the total variable and overhead cost of drone delivery would be far cheaper than other options. Specifically, one team showed that at $1 per delivery of a 5 pound package within a 10 mile radius, drone delivery would be vastly less expensive than current alternatives for Amazon:[72]

FIGURE 32: COMPARISON OF DELIVERY COST OF 5 LB PACKAGE WITHIN 10 MILE RADIUS

It is also important to note that drone delivery could be the only viable logistics solution for areas of the world without access to all-season roads. An estimated one billion people in the world do not have access to roads for at least some part of the year.[73] Even if they are not as economical, if drone delivery can simply allow for logistics of important payloads between areas that are not connected by roads, drones could be incredibly impactful. For instance, medical supplies, vaccines, and test samples have been shown to be flow between areas in mountainous or road deprived areas in a fraction of the time.

Clearly, use of drones to deliver payloads could have a substantial impact on the logistics industry across a variety of use cases within the United States and globally.

4.4 Sensor Usage

Drones allow inexpensive aerial use of visual, multispectral, thermal, laser, or hyperspectral sensors. These sensors have been shown to be incredibly valuable across a variety of fields including precision agriculture, geoinformatics (GIS), insurance, construction, mining, and oil & gas among many others. Drones have allowed for these sensors to capture sensor data from angles and altitudes never thought possible. By pairing robust software to algorithmically maneuver the aircraft to capture data for a specific task, and with complex modeling tools, drones are enabling us to map and understand our environment in unique ways. These applications are still in their early stages, but many companies are working on top of the reliable consumer hardware created by DJI, 3DR, and PrecisionHawk to interpret and model the physical way in incredible ways.

Applications are especially interesting in agriculture where interpretations of sensor data can lead to estimates of crop yields, crop status, water levels, drainage locations, planting evaluation, and more.

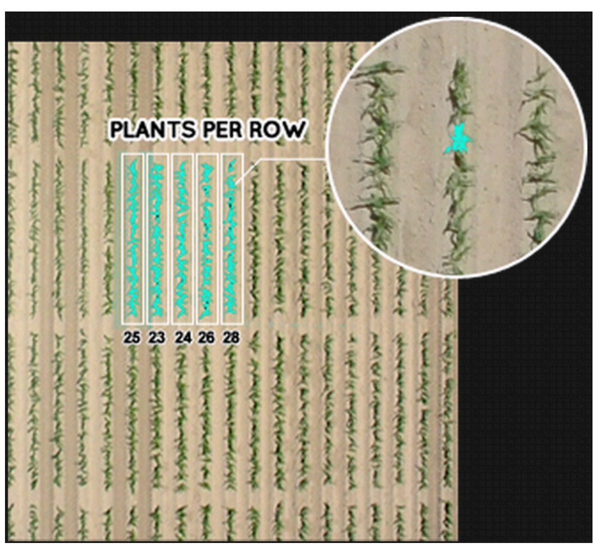


FIGURE 33: EXAMPLE OF CROP COUNTING FROM DRONE SENSOR DATA

Use of drone sensors to assist in monitoring or rescue missions for emergency response groups, border protection, mining, or oil and gas are also intriguing, as drones provide a far better alternative to helicopters in some cases. Software that would allow drones to assist in emergency response in indoor emergency scenarios, like building fires or bomb threats, has been extensively researched in literature.

Use of drone sensors to assist in 3D modeling of buildings or famous landmarks has been particularly successful recently. Pix4D's 3D modeling of the iconic "Christ the Redeemer" statue in Rio de Janeiro is one particularly exciting example.[74] Until Pix4D's model, accurate 3D modeling was not possible due to the location, size, accessibility, and weather conditions surrounding the statue. 3D models were exclusively based off of hand estimates, or ground based photographs. Below is an example of how drones can be programmed to intelligently maneuver and collect sensor data that can be easily transformed into accurate 3D renderings of an object:

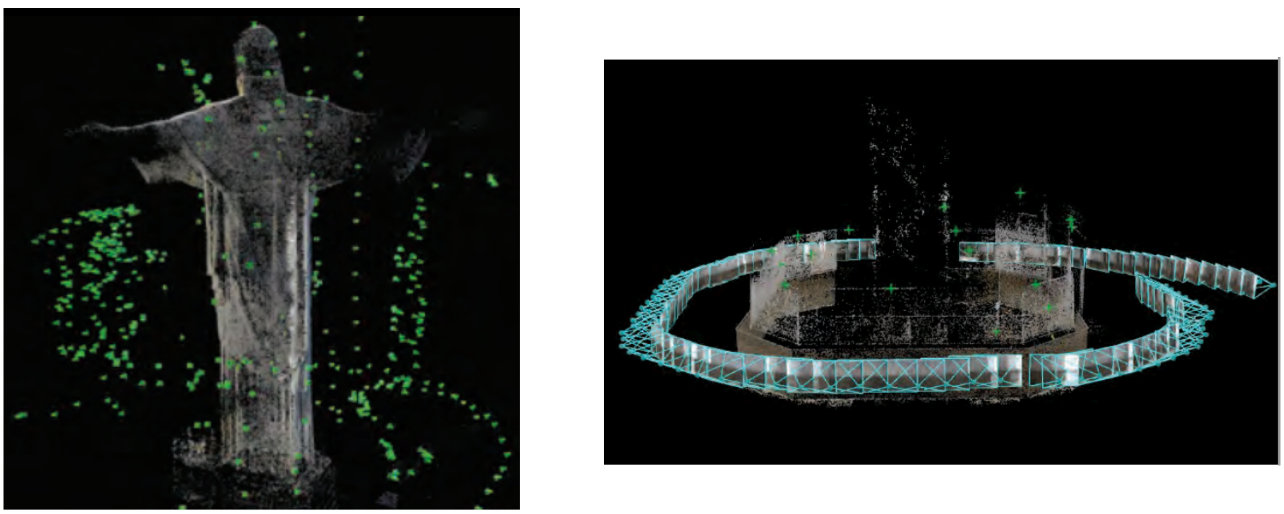


FIGURE 34: PIX4D POINT CLOUDS OF CRISTO REDENTOR FROM SENSOR DATA

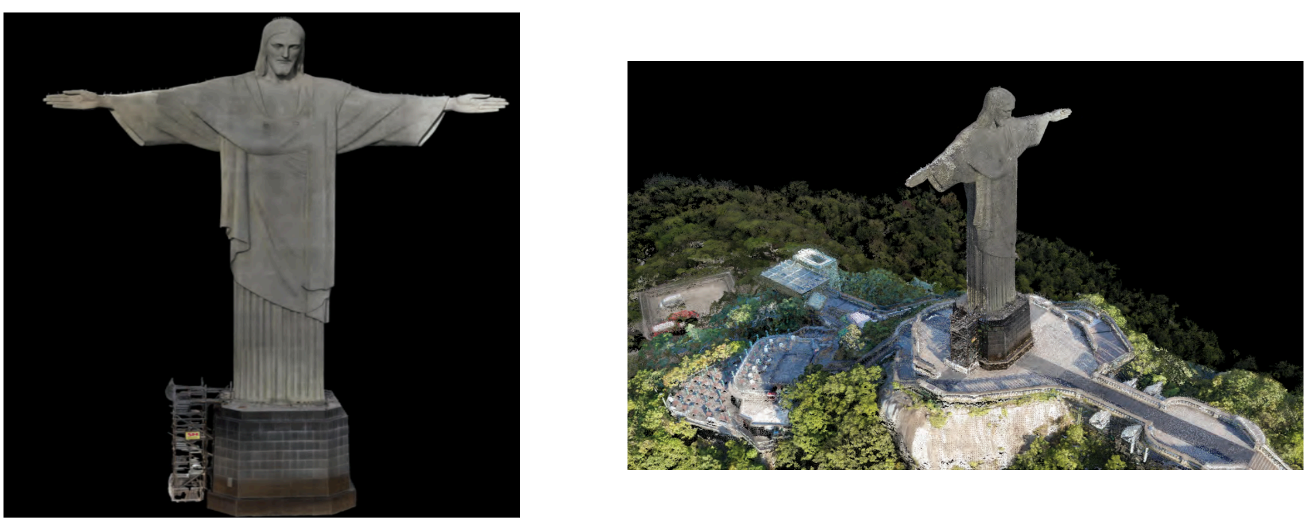


FIGURE 35: FINAL 3D OUTPUT

4.5 Safety & Privacy Concerns

A) National Safety

On January 26th, a DJI Phantom unintentionally crashed on the lawn of the White House.[75] DJI responded by quickly issuing a firmware update to prevent their drones from flying within 25 km of downtown Washington, D.C., however the incident engendered serious concerns of national and personal safety from malicious use or unintentional crashes of drones.[76] There is a clear public concern about physical and national safety with regard to drone crashes, though statistically, today's drones rarely fail due to human error.

B) Airports

Like birds, drones present significant risk to large, piloted aircraft. Even the smallest of drone can cause jet engine failure by from being sucked into the turbine when operating at close proximity. The worst fear is for a commercial aircraft to be affected in a way similar to how birds were sucked into the jet engines of US Airways flight 1549 causing complete failure of the propulsion system. Though many of the major drone manufacturers have restricted flights near airports, drones flying at high altitudes in any area can pose a threat to piloted aircraft. To date, there have been more than 15 incidents involving near mid air collisions (NMAC) between a drone and piloted aircraft both around airports and in other airspace, though no collisions have occurred yet.[77]

C) Hacking

In addition to concerns of malicious operators using their own drones, there is a legitimate concern that drones are also susceptible to malicious hackers. A research team at the University of Texas has shown that drones can be "spoofed" by providing false GPS information to the drone and leading it to fly in an unwanted trajectory.[78] Even worse, serial hacker Samy Kamkar has shown a way to hack nearby drones by overriding the WiFi connection between the pilot and the UAV.[79]

D) Privacy

Legitimate concerns over public privacy have also been raised. Capable of flying far away from the pilot, and gaining access to previously inaccessible areas, drones have widely been seen as a spying tool by many. It is a powerful tool for a malicious user as it allows him to uniquely invade privacy while (sometimes) staying both physically and legally out of danger. For instance, a user recently used a DJI Phantom record footage of Apple's top-secret, heavily guarded new headquarters by air.[80] He was not prosecuted as he did not break any FAA regulations at the time, but was clearly invading Apple's privacy.

But the issue arises even with those who don't intend to "spy." Even though it is not their purpose, most civil drones use some type of visual sensor, causing many to believe that their use may result in unintentional violations of public privacy.

4.6 Military Strikes

Since the introduction of General Aviation's Predator-series drone in 1995, public perception of drones has been focused on military drone strikes. Predators have been used extensively in "precision strikes" which aim to eliminate high-ranking enemy officials. However, many believe these strikes to be less than precise. Some reports claim that for every high-level official killed by a drone strike, on average 28 others are also killed in the attack.[81] Recently, these military drone strikes have been in the spotlight due to the accidental death (due to faulty intelligence) of two American hostages during an April attack on an al Qaeda compound.[82] These strikes have lead many to a negative perception of drones, as military and civil drones are often (mistakenly) grouped together. Many have gone as far as to protest UAV research and demand stringent civil and military UAV regulation to prevent further military use, as they fear capable autonomous drones could lead to targeted attacks on US citizens, or, even worse, an apocalyptic scenario where artificial intelligence could allow for drones to assert control over humans.

In short, drones have a bad wrap with the public today.

5. Regulation and Integration Requirements

Integrating UAVs into the National Aerospace System (NAS) is a complex technological and regulatory issue. The goal of integration is for drones of all sizes and types to have an understandable and straightforward set of rules and standards that regulate their usage. Regulation will include a number of separate laws and guidelines from numerous stakeholders. Basic regulation will include UAV certification protocol (similar to DOT testing of cars before they can be sold), minimum operational and performance standards (MOPS) which outline certain metrics flight controllers must meet to be acceptable, and usage guidelines that outline when, where, and how UAVs are flown.

At the helm of regulation, is the Federal Aviation Administration (FAA), which is part of the Department of Transportation (DOT). To speed up the act of integration, congress passed the *FAA Modernization and Reform Act of 2012*[83], which mandated the FAA to establish specific UAV provisions and deadlines for integration into the NAS. These mandates included a completion of integration plan by 2015, the selection of 6 test sites, publishing of a 5-year integration roadmap, creating an accelerated exemption system for commercial UAV use, and proposed rules for small UAVs operating within visual line of sight of the operator. However, the FAA has missed deadlines on all of the 17 provisions, and only implemented 9 to date. The effectiveness has been so poor, that the inspector general published a report in 2014 criticizing the FAA for its slow and ineffective rulemaking.[84]

Here we will outline the current UAV laws, and the considerations, and upcoming decisions involving integration of UAVs into the NAS.

5.1 Definitions

A) UAV Size

Currently, the FAA has only two classifications for UAVs. Small UAV (sUAV) are those which are less than 55 pounds in weight. Large UAVs are larger than 55 pounds in weight.

B) Line-of-Sight

UAV operation is delineated based on the ability of the pilot to see the aircraft while operating it. Visual-line-of-sight (VLOS) means that the pilot can visually see the aircraft during operation. Beyond-line-of-sight (BLOS) is a situation where the pilot is operating the aircraft without the ability to visually see it. Autonomous operation is almost always BLOS.

C) Airspace Classifications

Regulation of aircraft depends on the airspace in which it is operating. Airspace is delineated based on the height of the flight, the distance above mean sea level (MSL), distance above ground level, and the proximity to airports. Each classification has different requirements for entry, instrument use, pilot certification, and aircraft separation. All classes except for G are controlled under the FAA. An outline of the classifications is shown below:

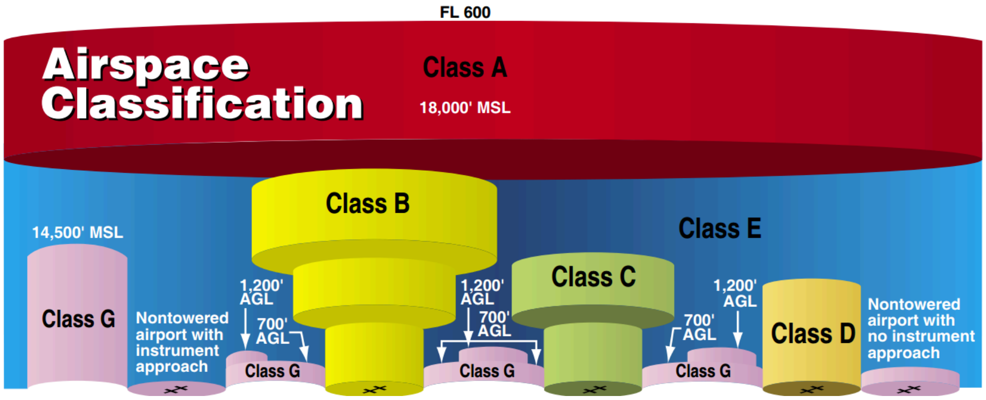


FIGURE 36: FAA AIRSPACE CLASSIFICATION[85]

5.2 Proposed Small UAV Regulation

After many months of delays and speculation, the FAA finally released its proposed rules for sUAV operation with their *Small UAS Notice of Proposed Rulemaking* in February of 2015.[86] Though not final, the notice finally outlined a few requirements for "non-recreational" (i.e. commercial) sUAV operation:

- Operation exclusively in VLOS, and in daylight conditions
- No pilot's license requirement, UAV pilots are classified as *operators* and must pass an aeronautical knowledge test every 24 months
- Pilot must be 17 years old
- Maximum altitude of 500 feet above ground
- Maximum airspeed of 100 mph
- Operation in class B, C, D, and E airspace are allowed with Air Traffic Control (ATC) permission
- No operation in class A (18,000 feet and above) airspace
- Reporting of any accident that results in property damage or physical injury within 10 days of the incident to the FAA

These regulations are currently *proposed* and are still in the process of receiving feedback from industry stakeholders. Finalized rules should be published in the next 120 days.

5.3 Current Commercial UAV Operation

Currently, commercial use of UAV is prohibited under federal law. However, under Section 333 of the FAA Modernization Act, businesses can apply for an exemption that would allow them to test their commercial UAVs. The exemption requires applicants to submit requests detailing the exact model they will be testing, and how it will be operated. Exempted companies also must submit constant reports to the FAA on their use. However, even with congress mandating that the FAA speed up this process, it has proven to be extremely slow. Amazon recently got approval to test their UAVs in the US, but the approval took so long that the model they requested exemption for was obsolete by the time it was approved.[87]

5.4 Beyond Line of Sight: Detect, Sense, and Avoid

Central to autonomous UAV integration into the NAS is their ability to detect, sense, and avoid obstacles and oncoming aircraft. The autonomous function of being able to proactively avoid mid-air collisions (MAC) is known as sense and avoid (SAA) or detect, sense, and avoid (DSA). As UAVs would be operating in the same airspace as manned aircraft in the NAS, the SAA systems used must provide a level of safety equaling or exceeding that of manned aircraft. Code of Federal Regulations 14, Part 91.113 outlines an aircraft's duty to "see and avoid" stating that: "…vigilance shall be maintained by each person operating an aircraft so as to see and avoid other aircraft. When a rule of this section gives another aircraft the right-of-way, the pilot shall give way to that aircraft and may not pass over, under, or ahead of it unless well clear."[88] Encounters in which see and avoid capabilities are used are relatively rare, with some estimating 3 near mid air encounters (NMAE) per 10,000 UAV flight hours[89]; however, these encounters must be programmatically prevented from becoming collisions for regulators to accept autonomous UAV operation in the NAS.

The Radio Technical Commission for Aeronautics (RTCA) is a non-profit association that advises the FAA on its MOPS regulation. Particularly, Special Committee 228 (SC-228) was formed in May of 2013 and is in charge of presenting performance standards for SAA systems to the FAA; however, these guidelines will not be proposed until at the 3rd quarter of 2015, and a final proposal is not due for release until July 2016.[90] Additionally, the first set of MOPS guidelines will only include recommendations for UAVs in "instrument flight rule" airspace, and those using radio communication. The deadlines for MOPS for UAS using satellite communication in all airspaces have not been proposed yet.[91] The FAA has noted in its 5-year integration plan that the final technical standard order for SAA MOPS should not be expected until the 1st quarter of 2017, though the scope of this technical order is unclear.[92,93]

Therefore, most currently rely on ATSM's Document F2411-04e1 *Standard Specification for Design and Performance of an Airborne Sense-and-Avoid System* or RTCA's DO-229D[94] to benchmark SAA performance.[95] ATSM's document, proposed in 2005, synthesized the current regulation and proposed standards (including RTCA's DO-229D) to outline a proposal of operational requirements and capabilities for SAA systems to operate legally in the NAS. Most notably, it noted that a "UAS [must] be able to DSA other aircraft within a range of ±15°

elevation and ±110° azimuth and respond in sufficient time so that a collision is avoided by a minimum of 500 feet. The 500-foot margin of safety derives from what is commonly defined as a near midair collision."[96] Additionally the report suggested that SAA systems should be effective enough at preventing MAC as manned systems, which, up to 2002, was an average of 0.51 MAC per million operating hours for manned aircraft. This ratio is seen as the baseline to evaluate SAA system performance. Others, including Honeywell, have suggested that a rate of 5×10^{-9} MAC/Flight Hour be used in transponder (as discussed below) required airspace, and 1×10^{-7} MAC/Flight Hour in non-transponder airspace.[97] Additionally of interest is that the majority of MAC occur within 3 miles of an airport, with more than 50% occurring below 1000 feet of altitude.[98] Finally of note, Boeing has estimated that ~70% of commercial aircraft "hull loss incidents" are attributed to human, not machine error.[99]

An overview of the process for producing the minimum operational and performance standard (MOPS) for DSA is shown below:

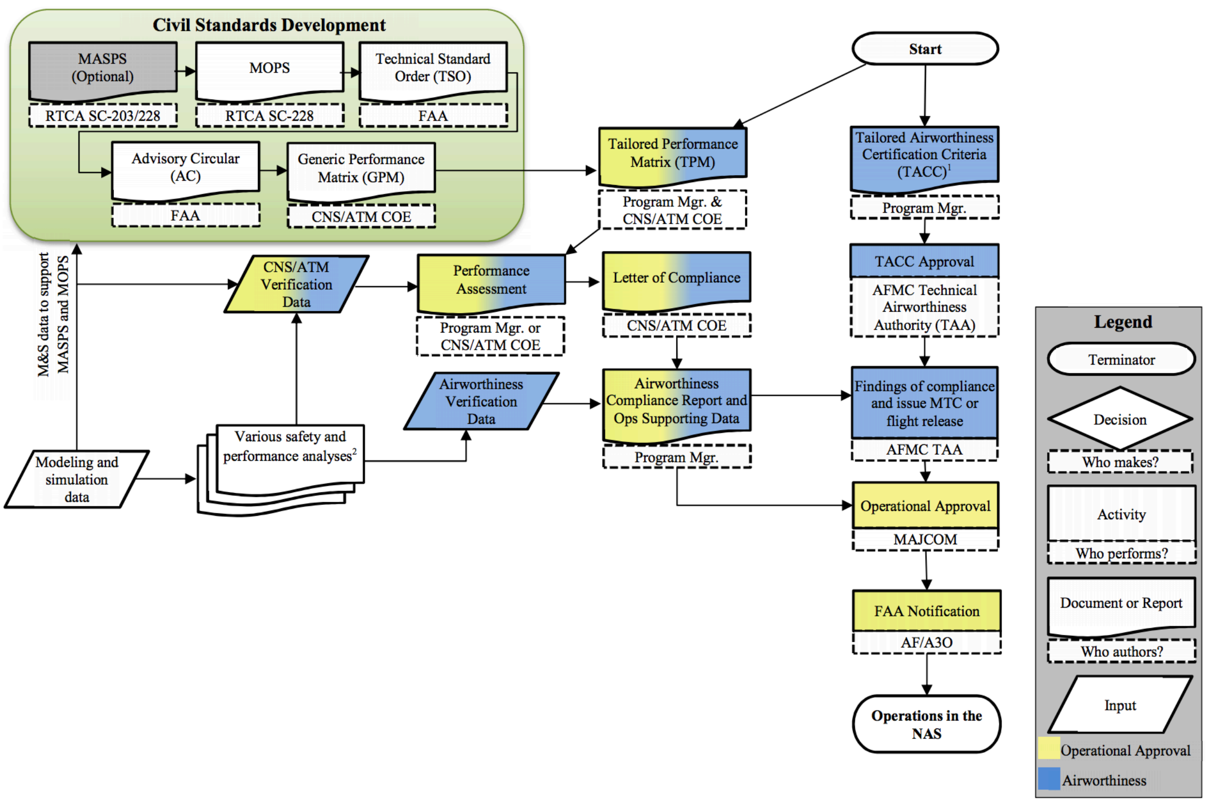


FIGURE 37: DECISION MAKING PROCESS FOR SAA MOPS APPROVAL[100]

Though the path planning methods presented in Section 3 is useful for low altitude or indoor flight, different technology is used to assist the UAV in avoidance of other aircraft at high altitude. Here we will explore some of the suggested technology from a high-level.

A) Cooperative Technology

Cooperative technology implies that the aircraft will interact with technology onboard other aircraft or systems on the ground. Though UAVs have not been required to have cooperative technology, there is a good chance those that operate at high altitude will be generally expected to incorporate them as they are highly effective at preventing near mid air encounters.

Traffic alert and collision avoidance systems (TCAS) is the primary cooperative technology used in the US. TCAS takes the form of a small box onboard the aircraft that transmits information back and forth to nearby aircraft via a transponder. However, much of this information is auditory, and therefore is not necessarily viable for autonomous UAV operation. Additionally, the size and weight of the TCAS transponders is cumbersome, especially to sUAV.

Automatic dependent surveillance-broadcast (ADS-B) is a new technology that allows both pilots and ground-based stations to detect other ADS-B equipped aircraft in the airspace.[101] Using GPS, ADS-B digitally transmits aircraft position, altitude, speed, aircraft type, flight number, and heading. The transmission is done through a broadcast with a discrete frequency through a universal access transceiver. Nearby aircraft and ground stations within a 150-200 mile radius of the aircraft can receive the ADS-B information. The aircraft information is updated several times per second, so broadcasts are always accurate. Because it is digitally based (and can therefore be programmatically interpreted by a UAV) and provides far more information than TCAS, ADS-B is the best cooperative technology available to assist UAV in SAA.

Though highly effective, a UAV system should not solely rely on cooperative technologies to achieve SAA capabilities. Because of its cost, and the lack of requirements for using it, it cannot be assumed that all other aircraft will possess cooperative technology. Additionally, these technologies provide no information about ground-based (trees, mountains, buildings) or non-aircraft obstacles (such as birds). Therefore, an effective SAA system must also make use of non-cooperative technology.

B) Non-cooperative Technology

Non-cooperative technology does not require other aircraft to possess the technology to be effective. These technologies are broken down into two categories: active and passive. Active systems transmit signals to search for obstacles. Passive systems don’t transmit signals, but instead rely on signals that come from the objects themselves (such as heat or motion).

Radar is by far the most active system available for UAV. As discussed above, radar sends waves in the electromagnetic spectrum and calculates the time it takes for the waves to reflect off of obstacles and return to the sensor. Specifically interesting for UAVs are synthetic aperture radar (SAR). SAR is interesting as they are relatively small, and recently have been developed to perform 3D scans, and detect motion detection. Radar is ideal for nighttime or cloudy environments, but are generally difficult to include on UAVs due to their size, weight, and cost.

Finally, visual cameras can be of use in light-intense environments. By using a feature detection algorithm as described in Section 4, images from camera sensors can provide valuable information about obstacles, though they have a limited range in sensing ability.

Other active systems, such as laser or sonar, can also be of use, and were discussed extensively in Section 2 and 4 above.

Passive systems, which simply interpret signals emitted by objects, are also of interest. One passive system would involve electro-optical sensors, which detect light emitted by objects. Infared based sensors are of particular interest for nighttime use as they only require heat emitted by objects to interpret them. Acoustic sensors, which use the sound emitted by other aircraft or objects to detect them, are also of interest.

5.5 Control and Non-Payload Communication

To operate autonomously, UAVs must be able to communicate with systems on the ground. This is generally referred to as control and non-payload communications (CNPC). NASA published a study in 2013 that identified the need of 34 MHz of spectrum for VLOS operations, and 56 MHz for BVLOS satellite based systems.[102] For CNPS, the World Radiocommunication Conference (WRC) has allocated 5030 MHz – 5091 MHz (C band), and 960MHz–977MHz (L band). There is also a possibility of more usage of the L band, which is currently exclusively used for Fixed Satellite Services (FSS) that could be used for CNPC. This will be discussed in the next WRC, and would need ultimate approval in the US from the Federal Communications Comission (FCC).

5.6 UAV Traffic Management (UTM)

Led by Dr. Parimal Kopardekar, NASA is developing a UAV Traffic Management System (UTM) to facilitate UAV operation at low altitude where mid-air collisions with manned aircraft are not probable. NASA compares the UTM to modern vehicle transportation infrastructure that consists of roads, lanes, stop signs, lights, and rules. According to NASA, the goal of UTM is to “enable safe and efficient low-altitude airspace operations by providing critical services such as airspace design and geo-fencing, separation management, weather and wind avoidance, routing, and contingency management.”[103] Central to its design will be the inclusion of strictly geo-fenced areas, where UAV operation is not allowed, and corridors, areas similar to streets where most UAV traffic will operate through.

NASA intends to build two types of UTM systems. The first, known as *portable* UTM, would be mobile and aimed to support specific UAV functions including precision agriculture and disaster relief. The second, *persistent* UTM, would be fixed to a certain geographical area and support continuous low-altitude UAV operation in the area.

NASA plans on developing the UTM systems through a series of four “builds” with increasing capability, and each delivered at 12-16 month intervals.[104] The first build will focus on geo-fencing, altitude control, and vehicle trajectory scheduling with up to 6 vehicles, and should be completed by Q1 2016. Following builds will compound upon each other until build 4, which will include contingency planning on a large-scale, failure planning, collision avoidance, a UTM web portal, and be tested on at least 20 heterogeneous vehicles over a variety or geographical conditions including dense urban environments. Build 4 is expected to finish testing by Q1 of 2020.

Though not much is known of the progress of build 1, industry stakeholders will be presented with progress at the annual stakeholder conference on July 28th, 2015 at the AMES research center in California.

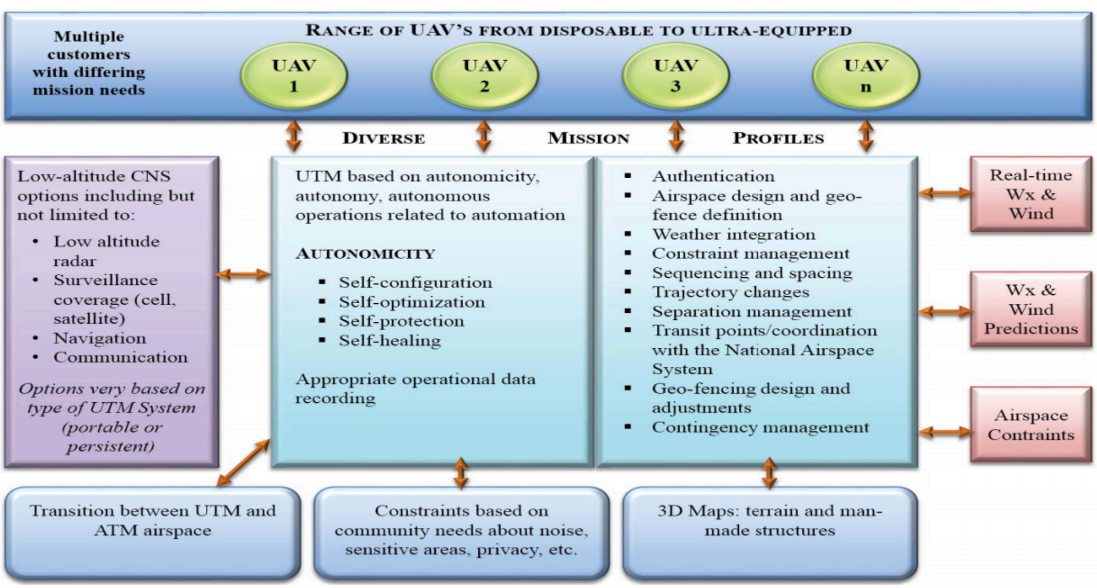


FIGURE 38: NASA'S DESCRIPTION OF UTM FUNCTIONALITY

6. What's Holding Us Back?

6.1 Battery Technology

By far the most prohibitive technology to UAS integration into the NAS is the current state of aircraft batteries. For UAVs batteries must have high capacity per unit of mass, as it is generally accepted that batteries comprise around a third of the UAVs takeoff mass.[105] We benchmark battery capabilities per unit of mass through its source specific energy, the total stored energy per unit mass of the power source (kWh/kg), and its source specific power, the instantaneous power delivered to the system per unit mass of the power source (kW/kg). Because of UAVs need for batteries capable of delivering high amounts of current for motor operation, and their sensitivity to weight, Li-Po batteries are generally used. Li-Po batteries are commonly found with source specific energy of >.100 kWh/kg and source specific power of >1 kW/kg. These batteries are commonly thought to a lifespan of around 500 cycles, and are relatively inexpensive today.

However, even with the most advanced Li-Po batteries, flight times of UAVs are limited to well below one hour.

A) Math Behind Flight Times

To estimate the flight time of a UAV hovering in terms of the battery capability and motor power requirements, I used the equations and logic from Yash Mulgaonkar's thesis.[106] The power required for a UAV to hover as P in watts, the thrust required to hover as T in newtons $\left(\frac{kg*m}{s^2}\right)$, battery voltage as V in volts, battery capacity as Q_b in mAh, the weight of a UAV without a payload as w_q in kg, the weight of the battery as w_b in kg, the total weight of the UAV system as W in kg, the energy of the battery as E_b in kWh, flight time at hover as t_h in hours, the proportionality constant between the power consumed and thrust as k_p, and finally the reciprocal of source specific energy of the battery as k_b.

Using these variables, we can find the flight time through the following equations. The power required for the UAV to hover, P, is defined as:

$$P = k_p * T^{\frac{3}{2}}$$

The energy of the battery, E_b, is defined as:

$$E_b = Q_b * V$$

The weight of the required battery and system defines as:

$$w_b = E_b * k_b$$

$$W = w_q + w_b$$

The equilibrium condition for finding the required thrust to maintain the weight of the system is:

$$T = W$$

The energy of the battery required to hover for a certain time is defined as:

$$E_b = P * t_h$$

Finally, combining the above equations, we can derive a final equation for flight time at hover in terms of inputs to the system:

$$E_b = k_p * T^{\frac{3}{2}} * \ t_h$$

$$t_h = \frac{E_b}{k_p * \left(w_q + (E_b * k_b)\right)^{\frac{3}{2}}}$$

It is easy to see that as k_b gets smaller, i.e. the source specific energy gets larger as k_b is the reciprocal, flight time at hover will get larger. Therefore, future battery research should be focused on developing batteries that are able to provide the same level of power as Li-Po batteries, but contain much higher source specific energy.

To determine how source specific energy could affect the range of a UAV, I backed out delivery range from the estimates from D'Angelo in Section 3. I aimed to see how source specific energy of a battery would affect the range of a UAV with a 2 kg battery, 2 kg frame, and 2 kg payload (either sensors or package).

D'Angelo estimates the energy requirement to be :

$$\frac{d}{1 - v_r}\left(\frac{m_p + m_v}{370\eta r} + \frac{p}{v_c}\right)$$

I backed out the range, d in km, to be:

$$d = \frac{2 * E_b(1 - v_r)}{\left(\frac{m_p + m_v}{370\eta r} + \frac{p}{v_c}\right)}$$

Where E_b is the source specific energy of the battery, m_p = payload mass in kg, m_v = vehicle mass in kg, r = lift-to-drag ratio, η = power transfer efficiency for motor and propeller, p = power consumption of electronics in kW, and v_c = cruising velocity of the aircraft in km/h. As noted in Section 3, we assume η to be .5, and r to be 3.

Using the estimates, I modeled how the range, in km, of a UAV could change over changes in payload mass and source specific energy:

Package Mass (kg)	Source Specific Energy (kWh/kg) 0.10	0.15	0.20	0.25	0.40	0.65	0.80	0.95
0.5	6.45	9.68	12.91	16.13	25.81	41.95	51.63	61.31
0.75	6.18	9.28	12.37	15.46	24.74	40.19	49.47	58.75
1	5.94	8.90	11.87	14.84	23.74	38.58	47.49	56.39
1.25	5.71	8.56	11.41	14.27	22.83	37.10	45.66	54.22
1.5	5.50	8.24	10.99	13.74	21.98	35.72	43.96	52.20
1.75	5.30	7.95	10.60	13.25	21.19	34.44	42.39	50.33
2	5.12	7.67	10.23	**12.79**	20.46	33.25	40.92	48.59
2.25	4.94	7.42	9.89	12.36	19.78	32.14	39.55	46.97
2.5	4.78	7.18	9.57	11.96	19.14	31.10	38.28	45.45
2.75	4.63	6.95	9.27	11.59	18.54	30.13	37.08	44.03

FIGURE 39: SENSITIVITY OF DELIVERY RANGE TO PACKAGE MASS AND SOURCE SPECIFIC ENERGY

Though these estimates make the assumption that the battery would be able to provide sufficient power for operation, it is clear that advances in battery source specific energy will drastically increase flight time and range.

B) Future Battery Technology

The most promising candidates for replacing Li-Po are currently Lithium Sulfur (Li-S), $LiMnPO_4$, and carbon nanotube based batteries.

Li-S research is led by Sion Power, and has shown the ability to achieve nearly 2x improvements over Li-Po in source specific energy, without giving up source specific power.[107] However, issues with the battery lifecycle, and the stability of the battery (some report Li-S batteries lighting on fire randomly) must be resolved.

$LiMnPO_4$ have also been seen as a viable alternative with minimal issues with stability, and greatly increased source specific energy.[108] However, $LiMnPO_4$ technology is still in the early stages of development, and very limited research is available.

Finally, nanotubes have been proposed for years as a viable way to provide far higher specific energy and power. Research efforts are at their early stages, but there have been recent positive trends. Specifically, the recent $15 million venture funding, led by IDG Capital Parnters, into China based CNano Technologies is a positive sign.[109]

C) Alternatives

Alternative solutions include use of fueled systems to provide power to the UAV. These UAVs would use traditional gasoline systems to enable flight. Though these systems can provide significantly higher specific energy than Li-Po alternatives (some estimate >.6 kWh/kg), they require significant engine components. These components, including pumps and fans for the engine, are difficult to scale down to small sizes for UAV use, and are considerably heavy. In addition to the difficulty in weight, the overhead of engine components typically leads to low source specific power, in the range of .1 kW/kg. As UAVs have high power requirements, most fueled systems are 1-2 orders of magnitude shy of quadcopter needs.

However, Top Flight technologies recently introduced a gas-powered UAV that they claim can achieve a flight time of 2 hours on 1 gallon of gasoline.[110] The validity of these claims has not yet been tested, but it is a positive sign for gas-powered systems.

D) In-Mission Recharging

As carrying all the necessary energy for a mission can prove to be difficult, in-mission recharging can be a viable alternative.

A battery swapping and charging technology, similar to those proposed by Mulgaonkar[97] and Toksoz[111] could prove to be effective. These technologies hold and charge multiple batteries, and are capable of removing a battery from a functioning UAV, inserting a new, charged battery, and recharging the removed battery. They can accomplish this battery swapping process while keeping the UAV powered. Strategically placing these battery swapping stations around a dense urban environment could allow UAVs to embark on long missions without returning to a central base.

Additionally, solar powered systems could be helpful in improving UAV flight time. By placing solar cells on a UAV, solar power could recharge the UAV. However, there is very limited space on multirotors for solar cells, and UAVs operating in dense urban environments are only sporadically exposed to sunlight.

6.2 Sense and Avoid Performance

Sense and avoid technologies must improve to be both computationally faster (most of the state-of-the-art 3D mapping and avoidance systems take >1-2 seconds to process each iteration), and incredibly dependent. Since UAVs are under immense regulatory and public scrutiny, any failure, even at rates far lower than those of manned aircraft, would likely be perceived highly negatively by the public. Before introduction into the NAS, the complex sense and avoid functionality of a UAV, for both environmental obstacles and aircraft avoidance, must dramatically improve. Though the quality of currently proposed sense and avoid technologies is not yet known, the implementations sited in literature are not nearly advanced enough to date for larger-scale civil use.

6.3 Sensor Cost + Weight

The sensors used for SAA and SLAM functionality are currently expensive and heavy. The simplest Lidar sensor available today weighs 210 g, and costs $2,500. The more advanced Lidar sensors used to provide complex 3D modeling and long-distance scans are currently far more expensive (>$10,000) Radar and IR technologies are equally as heavy and expensive. Improvements in the cost and weight of almost all of the sensors described above will be integral to advancements in UAV functionality.

6.4 Insurance & Liability

Though regulation must be in place before any operation in the NAS, firms must also be willing to deal with the immense liability associated with deploying automated UAVs. Usually, corporations are uneasy with the large potential liabilities, and instead choose to pay premiums to insurance companies to transfer the risk off their balance sheet. Even when the risk of liability is small, companies prefer the fixed cost of insurance premiums because they would rather budget their operations with a known, constant cost instead of constantly worrying about a major cash outlay caused by a freak accident. However, insurance companies (more precisely, the actuaries within the insurance companies), like regulators, are unsure how to quantify the risk of autonomous UAV operation. Extensive simulation and test data from both piloted and unmanned UAV operation is needed for insurance companies to feel comfortable issuing coverage for UAV operation. Early issuers of UAV insurance will either reap the major first-mover advantages of entering into sticky, long-term insurance agreements with UAV market leaders, or miserably fail by drastically underestimating the true risks of autonomous UAV operation. Only time will tell, but affordable UAV insurance is imperative for autonomous UAV adoption to take off.

6.5 Emergency Planning / Fault Recovery

UAVs, like all machines, can fail. Innovations that can prevent catastrophes, like the dangers of a heavy machine free-falling out of the sky, will be crucial to limiting UAV operational liability and convincing the public of their safety. Even if UAVs advance to the point were machine failure is unheard of, simply having a working "backup plan" will be helpful with public sentiment. Some early ideas include the inclusion of a parachute, or the use of gliding wings to assist malfunctioning drones in a gentler fall from grace.

6.6 Privacy Considerations

As cameras are light and cheap, they are used in almost every autonomous UAV. Though these cameras might be used exclusively for navigation and sensing, they can easily "accidently" or maliciously be used to invade privacy. For the public to be comfortable with UAVs acting autonomously in urban environments, efforts must be made to limit their ability to invade personal privacy.

7. Conclusion

We are on the cusp of a drone revolution. Enabled by tremendous improvements in computing power, wireless connectivity, software functionality, and sensor capability, UAVs have the potential to become a common and extremely valuable part of our skyline. With considerable drone applications in precision agriculture, logistics, photography, energy production, and emergency services combined with niche industries forming around their repair, retail, manufacturing, and insurance, it's not crazy to think that new, highly valued technology companies could emerge from a multi-billion dollar global drone market. Similar to how Jobs and Gates revolutionized the computing industry in the 70's and 80's, young, ambitious founders with visionary ideas are building brands in a new market by commercializing technology originally developed and funded exclusively by the military. Even better, access to capital to fuel innovation has never been higher. Traditional capital sources like venture capital are near all-time funding highs, and non-traditional funding from cash-flush corporate venture arms, investment funds, crowdsourcing vehicles, and big banks has become common as public market investment returns have lagged.

However, the industry is at a crossroad – though it's clearly possible that drones could innovate to where they are an integral part of our economy and society, regulators have the distinct ability to swiftly halt the drone craze with the passage of an overly conservative or prohibitive law. A regulator's worst nightmare is to pass laws today that could later be viewed as far too lenient tomorrow. So, they almost always err on the side of conservatism as they presumably have no economic skin in the game - the FAA's goal is not to maximize aviation's economic impact, but rather to maintain the safest aviation system in the world. Compound this with the generally accepted sentiment that it is far harder to repeal stringent regulation than prevent it and the importance of the upcoming regulation becomes clear.

For drones to be permitted to operate in the NAS in a manner that would allow the many promising applications to actually take place, regulators must be convinced that these early entrants will achieve their grand visions of drone integration in a safe, reliable, and ethical manner *every* step of the way. To bring about the most drone-friendly legislation and ensure the quickest and most accommodating integration into the NAS, I have offered the following six suggestions.

First, improve their public image. Donate your hardware to local emergency responders, non-profits, and schools and engage in pilot testing or case studies with social impact in mind. Work on press that focuses on the positive drones can do, not the negative. Continue the "know before you fly" campaigns, and make certain your position on unsafe flying and unethical use is clear. Convincing the public that civil UAVs are a completely different beast than military drones will be a complex challenge, but top of mind to regulators who are heavily impacted (and voted for) by public sentiment.

Second, focus on safety and reliability, not cash flow in the short term. Though aiming for high gross margins is still important to ensure future profitability is an option, building out the safest hardware and software humanly possible before regulators implement legislation is far more important to long-term market success. Do whatever it takes to surround your engineering teams with the most talented and capable people in the world, and deploy resources focused on safe autonomous operation until the marginal dollar spent ceases to lead to more progress. Again, time is more important than ever, so don't worry about the "dollarized return" on R&D investment, but rather measure success in improvements of a tenth of a mid air collisions avoid per million flight hours for your sense-and-avoid software, seconds of processing speed of your SLAM algorithms, or fraction of a kWh / kg for your batteries. Test and simulate your systems as often and extensively as possible, and be sure to validate the tests with 3rd parties. This will be crucial not only for shaping regulators opinion, but also in the cost and availability of enabling industries like drone insurance.

Third, opportunistically raise capital to amass a war chest large enough to put off profitability until after the regulatory cloud passes. Capital is cheap and valuations have never been higher. Fundraising is incredibly time consuming and often affected by forces outside your control, so raise enough to weather a storm. This will allow you to focus on incremental innovation and brand building, not profit, over the next few years. Focus on raising from those who share your long-term vision, and understand that cash flow in the short term could and will directly trade off with the long size and profitability of your operation in the long-term. If R&D and testing is the only reason you are not profitable, cutting cost and turning a profit when it is necessary will be easy; however, this overhead is a necessary and important part of your business today.

Fourth, engage heavily in lobbying efforts collectively with your competitors. The size and scope of the entire drone market depends on the decisions of regulators, so continue to join together in aggressive lobbying efforts like the Small UAV Coalition and the Association for Unmanned Vehicle Systems International (AUVSI) to get in front of regulatory stakeholders proactively.

Fifth, the roads or "corridors" for dense UAV travel in NASA's UTM must be well defined, and open to public discussion. As many don't want the noise or image of drones invading their personal space, property owners should be given the option to geo-fence their property from low altitude drone operation or at least be allowed to vote on proposed UAV corridors in their locale. UAV travel should also only happen at specific times during the day (probably during the most active work hours), and should aim to minimize disturbances to households specifically.

Finally, engage 3rd party research centers and universities. Actively issue grants, host competitions with big cash prizes, sponsor hackathons, or fund entire research centers. Even go as far as to use research universities to independently test and validate your claims. Enable them with API's, free hardware, and open-sourced software, and reward them with what universities and college students love most – a little cash. Today's engineering students can be unique sources of innovation and aligning yourself with them can be tremendously helpful with public sentiment.

8. Figures Cited

Figure 1) http://s3-blogs.mentor.com/jvandomelen/files/2012/06/SperryATorpedo1918.jpg
Figure 2) http://en.wikipedia.org/wiki/Radioplane_Company#/media/File:MarilynMonroe_-_YankArmyWeekly.jpg
Figure 3) http://en.wikipedia.org/wiki/Radioplane_Company#/media/File:WTL_and_Radioplane_OQ-3_(4978679254).jpg
Figure 4) http://www.dji.com/product/phantom
Figure 5) http://whyrll.com/parrot-ar-drone-cool-smartphone-controlled-quadcopter/
Figure 6) http://www.mpoweruk.com/motorsbrushless.htm
Figure 7) http://theboredengineers.com/2012/05/the-quadcopter-basics/
Figure 8) http://www.socialledge.com/sjsu/images/thumb/2/26/CmpE244_S14_Quadcopter_Quad_motion1.JPG/450px-CmpE244_S14_Quadcopter_Quad_motion1.JPG
Figure 9) A) https://store.3drobotics.com/products/3dr-pixhawk B) http://www.dji.com/product/a2 C) http://media.precisionhawk.com/topic/say-hello-to-latas/ D) http://www.airware.com/products/flight-core
Figure 10) http://en.wikipedia.org/wiki/Kalman_filter#/media/File:Basic_concept_of_Kalman_filtering.svg
Figure 11) Fraundorfer, F., & Scaramuzza, D, 2012, p. 2
Figure 12) Rosten, E., & Drummond, T., 2006, p. 436.
Figure 13) Bachrach, A. G. 2009, Pg. 51.
Figure 14) Scaramuzza, D., & Fraundorfer, F. , 2011, p. 6.
Figure 15) Bachrach, A. G. 2009, Pg. 33.
Figure 16) Bachrach, A. G. 2009, Pg. 34.
Figure 17) Bachrach, A. G. 2009, Pg. 36.
Figure 18) Bachrach, A., et al ,2011, p. 8.
Figure 19) Bachrach, A., et al ,2011, p. 11.
Figure 20) Dryanovski, I., et al, 2010, p. 1554.
Figure 21) Dryanovski, I., et al, 2010, p. 1557.
Figure 22) Shen, S. ,2014, p. 25.
Figure 23) Sucan, I. A., et al ,2010, p. 1.
Figure 24) Sucan, I. A., et al ,2010, p. 2.
Figure 25) Karaman, S., et al. ,2010, p. 8.
Figure 26) Karaman, S., et al. ,2010, p. 8.
Figure 27) Bry, A., et al, 2011, p. 7.
Figure 28) Bachrach, A., Roy, N., & He, R. ,2009, p. 4
Figure 29) https://www.cbinsights.com/blog/drone-startups-venture-capital/
Figure 30) Self-created
Figure 31) Self-created
Figure 32) http://ark-invest.com/industrial-innovation/amazon-drones
Figure 33) https://www.datamapper.com/
Figure 34) https://pix4d.com/wp-content/uploads/2013/04/Projeto_Redentor_Pix4D_AeryonLabs_Whitepaper_2015.pdf
Figure 35) https://pix4d.com/wp-content/uploads/2013/04/Projeto_Redentor_Pix4D_AeryonLabs_Whitepaper_2015.pdf
Figure 36) http://www.faa.gov/regulations_policies/handbooks_manuals/aviation/pilot_handbook/media/phak%20-%20chapter%2014.pdf
Figure 37) Lester, T., et al, 2014, p. 12
Figure 38) https://prod.nais.nasa.gov/eps/eps_data/162405-OTHER-001-001.pdf

9. Endnotes

---

[1] Dillow, C. (2014, October 09). A brief history of drones. Retrieved from http://fortune.com/2014/10/09/a-brief-history-of-

[2] Tesla, N. (1898). *U.S. Patent No. US613809 A*. Washington, DC: U.S. Patent and Trademark Office.

[3] Sperry, E. A. (1917). *U.S. Patent No. US 1279471 A*. Washington, DC: U.S. Patent and Trademark Office.

[4] Shaw, I. G. (2014). The Rise of the Predator Empire: Tracing the History of U.S. Drones. Retrieved from https://understandingempire.wordpress.com/2-0-a-brief-history-of-u-s-drones/

[5] Sorensen, E. M. (1946). *U.S. Patent No. US 2408819 A*. Washington, DC: U.S. Patent and Trademark Office.

[6] Wagner, W., & Sloan, W. P. (1992). *Fireflies and other UAVs (unmanned aerial vehicles)*. Arlington, TX: Aerofax.

[7] TIME. (1964, November 27). Nation: The Firebee. *TIME*, *84*(23). Retrieved April 20, 2015, from http://content.time.com/time/magazine/article/0,9171,871380,00.html

[8] Whittle, R. (2014). *Predator: The secret origins of the drone revolution*. Henry Holt.

[9] General Atomics Aeronautical. (2014, April 9). Bringing Together the Pieces: An Industry Perspective on Integrating UAS into the NAS. *ICNS 2014*.

[10] Anderson, C. (2010, January 6). Parrot AR.Drones specs. Retrieved from http://diydrones.com/profiles/blogs/parrot-ardrones-specs-arm9

[11] United States, Federal Aviation Administration. (2015, February 23). *Small UAS Notice of Proposed Rulemaking*. Retrieved April 20, 2015, from https://www.faa.gov/regulations_policies/rulemaking/recently_published/media/2120-AJ60_NPRM_2-15-2015_joint_signature.pdf

[12] United States. Federal Aviation Administration. Air Traffic Organization Operations Planning Office of Aviation Research and Development.*Literature Review on Detect, Sense, and Avoid Technology for Unmanned Aircraft Systems*. By S. B. Hottman, K. R. Hansen, and M. Berry. Washington, DC: U.S. Dept. of Transportation, Federal Aviation Administration, September, 2009.

[13] Kendoul, F. (2012). Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems. *Journal of Field Robotics, 29*(2), 315-378. doi:10.1002/rob.20414

[14] Airware. (2014, December 17). High-Accuracy GPS | Aerial Insights. Retrieved from https://www.youtube.com/watch?v=-c4syPUkens

[15] Mulgaonkar, Y., Kumar, V., Shen, S., & Michael, N. (2014, June 7). Multi-Sensor Fusion for Robust Autonomous Flight in Indoor and Outdoor Environments with a Rotorcraft MAV. *2014 IEEE International Conference on Robotics & Automation (ICRA)*.

[16] Shen, S., Michael, N., & Kumar, V. (2013). Obtaining Liftoff Indoors: Autonomous Navigation in Confined Indoor Environments. *IEEE Robotics & Automation Magazine, 20*(4), 40-48. Retrieved from IEEE Xplore.

[17] Kalman, R. E. (1960). A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering, 82*(1), 35-45. doi:10.1115/1.3662552

[18] Wikipedia. (n.d.). Kalman Filter. Retrieved from http://en.wikipedia.org/wiki/Kalman_filter#Overview_of_the_calculation

[19] Faragher, R. (2012). Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation [Lecture Notes]. *IEEE Signal Processing Magazine, 29*(5), 128-132. doi:10.1109/MSP.2012.2203621

[20] Bry, A., Bachrach, A., & Roy, N. (2011). State Estimation for Aggressive Flight in GPS-Denied Environments Using Onboard Sensing. *IEEE International Conference on Robotics and Automation*. Retrieved January 18, 2015, from http://groups.csail.mit.edu/rrg/papers/icra12_aggressive_flight.pdf

[21] Julier, S. J., & Uhlmann, J. K. (1997). New extension of the Kalman filter to nonlinear systems. *SPIE Proceedings, 3068*. doi:10.1117/12.280797

[22] Bachrach, A. G. (2009). *Autonomous Flight in Unstructured and Unknown Indoor Environments* (Unpublished master's thesis). Massachussetts Institute of Technology. Retrieved from http://dspace.mit.edu/bitstream/handle/1721.1/54222/599812275.pdf

[23] Thrun, S., Burgard, W., & Fox, D. (2005). *Probabilistic Robotics* (1st ed.). Cambridge, MA: MIT Press.

[24] Fraundorfer, F., & Scaramuzza, D. (2012). Visual Odometry: Part II Matching, Robustness, Optimization, and Applications. *IEEE Robotics & Automation Magazine*. doi:10.1109/MRA.2012.2182810

[25] Harris, C., & Stephens, M. (1988). A Combined Corner and Edge Detector. Retrieved from http://www.ee.iisc.ernet.in/new/people/faculty/soma.biswas/pdf/harris1988.pdf

[26] Shi, J., & Tomasi, C. (1994). Good Features to Track. In *1994 IEEE Computer Society Conference on Computer Vision and Recognition*. Retrieved from http://movement.nyu.edu/mocap11f/papers/lec03_OpenCV_FeaturesFinding.pdf doi:10.1109/CVPR.1994.323794

[27] Rosten, E., & Drummond, T. (2006). Machine learning for high-speed corner detection. *European Conference on Computer Vision* (Vol. 1, pp. 430-443).

[28] Lowe, D. (2003). Distinctive image features from scale-invariant key- points. *International Journal on Computer Vision, 20*(2), 91-110.

[29] Bay, H., Tuytelaars, T., & Gool, L. (2006). Surf: Speeded up robust features. In *ECCV*. Retrieved from http://www.vision.ee.ethz.ch/~surf/eccv06.pdf

[30] Agrawal, M., Konolige, K., & Blas, M. (2008). Censure: Center surround extremas for realtime feature detection and matching. *Proceedings of European Conference on Computer Vision* (pp. 102-115).

[31] Bouguet, J., & Intel Corporation. (2000). Pyramidal Implementation of the Lucas Kanade Feature Tracker Description of the algorithm. Retrieved from http://robots.stanford.edu/cs223b04/algo_tracking.pdf

[32] Scaramuzza, D., & Fraundorfer, F. (2011). Visual Odometry Part I: The First 30 Years and Fundamentals. *IEEE Robotics and Automation Magazine, 18*(4). Retrieved from http://rpg.ifi.uzh.ch/docs/VO_Part_I_Scaramuzza.pdf

[33] Cross, G. B. (2015). *Sparse Monocular Visual Odometry* (Unpublished master's thesis). University of Pennsylvania.

[34] Klein, G., & Murray, D. (2007, November 17). Parallel Tracking and Mapping for Small AR Workspaces. In *6th ICEE and ACM International Symposium on Mixed and Augmented Reality*. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4538852&tag=1

[35] Achtelik, M. (2009). *Vision-based pose estimation for autonomous micro aerial vehicles in gps-denied areas* (Master's thesis, Technische Universitat Munchen, 2009). Munich.

[36] Umeyama, S. (1991). Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 13*(4), 376-380. doi:10.1109/34.88573

[37] Zhang, Z. (1994). Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision, 13*(2), 119-152. doi:10.1007/BF01427149

[38] CARMEN Robot Navigation Toolkit. (n.d.). Retrieved from http://carmen.sourceforge.net/

[39] Bachrach, A., Roy, N., & He, R. (2009). Autonomous Flight in Unknown Indoor Environments. *International Journal of Micro Air Vehicles, 1*, 217-228. Retrieved January 18, 2015, from http://dspace.mit.edu/handle/1721.1/66162

[40] Bachrach, A., Prentice, S., He, R., & Roy, N. (2011). RANGE-Robust autonomous navigation in GPS-denied environments. *Journal of Field Robotics, 28*(5), 644-666. doi:10.1002/rob.20400

[41] Shen, S. (2014). *Autonomous navigation in complex indoor and outdoor environments with micro aerial vehicles* (Unpublished doctoral dissertation). University of Pennsylvania, Department of Electrical and Systems Engineering.

[42] Meagher, D. (1982). Geometric modeling using octree encoding. *Computer Graphics and Image Processing, 19*(1), 85. doi:10.1016/0146-664X(82)90128-9

[43] Dryanovski, I., Morris, W., & Xiao, J. (2010). Multi-Volume Occupancy Grids: An Efficient Probabilistic 3D Mapping Model for Micro Aerial Vehicles. *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. doi:10.1109/IROS.2010.5652494

[44] "Oxford PTAM Github," accessed: 30-03-2015. [Online]. Available: https: //github.com/Oxford-PTAM/PTAM-GPL

[45] Grisetti, G., Stachniss, C., & Burgard, W. (2005). Improved techniques for grid mapping with Rao-Blackwellized particle filters. In *2005 IEEE International Conference on Robotics and Automation*. Barcelona, Spain: IEEE. doi:10.1109/ROBOT.2005.1570477

[46] OpenSLAM.org. (n.d.). Retrieved from http://openslam.org/

[47] Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik, 1*(1), 269-271. doi:10.1007/BF01386390

[48] Bellman, R. (1956). *On a routing problem*. Santa Monica, CA: Rand.

[49] Sucan, I. A., & Kavraki, L. E. (2010, May 7). On the Implementation of Single-Query Sampling-Based Motion Planners. *2010 IEEE International Conference on Robotics and Automation (ICRA)*. doi:10.1109/ROBOT.2010.5509172

[50] LaValle, S. M. (1998). *Rapidly-Exploring Random Belief Trees: A New Tool for Path Planning* (Tech.). Retrieved http://msl.cs.uiuc.edu/~lavalle/papers/Lav98c.pdf

[51] Karaman, S., & Frazzoli, E. (2010). Incremental Sampling-based Algorithms for Optimal Motion Planning. *Robotics: Science and Systesm*.

[52] Bry, A., & Roy, N. (2011). Rapidly-exploring Random Belief Trees for Motion Planning Under Uncertainty. *IEEE International Conference on Robotics and Automation*. Retrieved January 18, 2015, from http://groups.csail.mit.edu/rrg/papers/abry_icra11.pdf

[53] Sucan, I. A., Moll, M., & Kavraki, L. E. (2012). The Open Motion Planning Library. *IEEE Robot. Autom. Mag, 19*(4), 72-82. Retrieved from http://ompl. kavrakilab.org

[54] Meeker, M., & KPCB. (2014, May 28). *Internet Trends 2014* (Publication). Retrieved http://www.kpcb.com/internet-trends

[55] Google. (n.d.). ATAP Project Tango. Retrieved from https://www.google.com/atap/project-tango/

[56] Rose, D. (2014, October 14). Dudes With Drones. *The Atlantic*. Retrieved from http://www.theatlantic.com/magazine/archive/2014/11/dudes-with-drones/380783/

[57] Nicas, J., & Macmillan, D. (2015, April 14). Chinese Drone Maker DJI in Talks to Raise Funding at Valuation of $10 Billion. *The Wall Street Journal*. Retrieved from http://www.wsj.com/articles/chinese-drone-maker-dji-in-talks-to-raise-funding-at-valuation-of-10-billion-1429050685

[58] Teal Group. (2014, July 17). *Teal Group Predicts Worldwide UAV Market Will Total $91 Billion in Its 2014 UAV Market Profile and Forecast* [Press release]. Retrieved from http://www.tealgroup.com/index.php/about-teal-group-corporation/press-releases/118-2014-uav-press-release

[59] CBInsights. (2015, February 14). Flying High – VC Funding to Drones Tops $100 Million in 2014. Retrieved from https://www.cbinsights.com/blog/drone-startups-venture-capital/

[60] Teal Group. (2014, July 17). *Teal Group Predicts Worldwide UAV Market Will Total $91 Billion in Its 2014 UAV Market Profile and Forecast* [Press release]. Retrieved from http://www.tealgroup.com/index.php/about-teal-group-corporation/press-releases/118-2014-uav-press-release

[61] Jenkins, D., & Vasigh, B. (2013, March). *The Economic Impact of Unmanned Aircraft Systems Integration in the United States* (Rep.). Retrieved http://www.auvsi.org/econreport

[62] *Testimony of Jesse Kallman Before the House Transportation and Infrastructure Subcommittee on Aviation* (2014) (testimony of Jesse Kallman, Airware).

[63] Pilkington, E. (2015, March 30). Amazon tests delivery drones at secret Canada site after US frustration. *The Guardian*. Retrieved from http://www.theguardian.com/technology/2015/mar/30/amazon-tests-drones-secret-site-canada-us-faa

[64] Dean, J. (2014, December 03). Aerial Video 79.6% Savings : Drone vs Helicopter. Retrieved from http://ihumanmedia.com/2014/12/03/aerial-video-79-6-savings-drone-vs-helicopter/

[65] DHL, & Heutger, M. (2014). *Unmanned Aerial Vehicles in Logistics: A DHL Perspective on implications and use cases for the logistics industry* (Rep.). Retrieved http://www.dhl.com/content/dam/downloads/g0/about_us/logistics_insights/dhl_trend_report_uav.pdf

[66] Bezos, J. (2013, December 1). Amazon's Jeff Bezos looks to the future [Interview by C. Rose]. Retrieved from http://www.cbsnews.com/news/amazons-jeff-bezos-looks-to-the-future/

[67] Amazon, & Misener, P. (2014, July 10). Amazon Petition for Exemption [Letter written July 9, 2014 to Federal Aviation Administration]. Retrieved from http://www.regulations.gov/#!documentDetail;D=FAA-2014-0474-0001

[68] D'Andrea, R. (2014). Guest Editorial: Can Drones Deliver? *IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, 11*(3), 647-648. doi:10.1109/TASE.2014.2326952

[69] Mulgaonkar, Y., Whitzer, M., Morgan, B., Kroninger, C. M., Harrington, A. M., & Kumar, V. (2014). Power and weight considerations in small, agile quadrotors. *Proc. SPIE 9083, Micro- and Nanotechnology Sensors, Systems, and Applications VI.*

[70] Tesla Motors. (2014). *Tesla Gigafactory* (Rep.). Retrieved http://www.teslamotors.com/sites/default/files/blog_attachments/gigafactory.pdf

[71] Matternet. (n.d.). Mttr.net. Retrieved from http://mttr.net/

[72] Keeney, T. (2015, April 13). *Amazon Drones Could Deliver a Package in Under Thirty Minutes for One Dollar* (Rep.). Retrieved Amazon Drones Could Deliver a Package in Under Thirty Minutes for One Dollar

[73] Raptopoulos, A. (2013, June). *Andreas Raptopoulos: No roads? There's a drone for that*. Speech presented at TED Global 2013. Retrieved from https://www.ted.com/talks/andreas_raptopoulos_no_roads_there_s_a_drone_for_that/transcript?language=en#t-322191

[74] Chen, L., Betschart, S., & Blaylock, A. (2015, February 18). *Projeto Redentor White Paper* (Rep.). Retrieved https://pix4d.com/wp-content/uploads/2013/04/Projeto_Redentor_Pix4D_AeryonLabs_Whitepaper_2015.pdf

[75] Associated Press. (2015, January 26). White House on lockdown after drone crashes into lawn. Retrieved from http://nypost.com/2015/01/26/possible-drone-found-on-white-house-grounds/

[76] Etherington, D. (2015, January 28). New DJI Drone Firmware Will Prevent White House Joy Rides, Border Drug Runs. Retrieved from http://techcrunch.com/2015/01/28/new-dji-drone-firmware-will-prevent-white-house-joy-rides-border-drug-runs/

[77] Whitlock, C. (2014, June 23). Close encounters on rise as small drones gain in popularity. *The Washington Post*. Retrieved from http://www.washingtonpost.com/sf/investigative/2014/06/23/close-encounters-with-small-drones-on-rise/

[78] University of Texas at Austin. (n.d.). Todd Humphreys' Research Team Demonstrates First Successful GPS Spoofing of UAV. Retrieved from http://www.ae.utexas.edu/news/features/todd-humphreys-research-team-demonstrates-first-successful-gps-spoofing-of-uav

[79] Goodin, D. (2013, December 3). Flying hacker contraption hunts other drones, turns them into zombies. Retrieved from http://arstechnica.com/security/2013/12/flying-hacker-contraption-hunts-other-drones-turns-them-into-zombies/

[80] Jmcminn (Youtube User) (Director). (2014, August 24). *Apple Campus 2 construction video - August 2014 - shot with GoPro* [Video file]. Retrieved from https://www.youtube.com/watch?v=pfZvimPkKio

[81] Ackerman, S. (2014, November 24). 41 men targeted but 1,147 people killed: US drone strikes – the facts on the ground. *The Guardian*.

[82] Diamond, J. (2015, April 23). U.S. drone strike accidentally killed 2 hostages. *CNN*.

[83] FAA Modernization and Reform Act of 2012: Federal Law 112-95, Title III, Subtitle B – Unmanned Aircraft Systems, 2012.

[84] Hampton, M. E. (2014, June 26). *FAA FACES SIGNIFICANT BARRIERS TO SAFELY INTEGRATE UNMANNED AIRCRAFT SYSTEMS INTO THE NATIONAL AIRSPACE SYSTEM* (United States, Department of Transportation, Inspector General).

[85] FAA. (2014). *Airspace Classifications* (United States, Federal Aviation Administration). Retrieved from http://www.faa.gov/regulations_policies/handbooks_manuals/aviation/pilot_handbook/media/phak%20-%20chapter%2014.pdf

[86] FAA. (2015, February 23). *Overview of Small UAS Notice of Proposed Rulemaking* (United States, Department of Transportation, Federal Aviation Administration). Retrieved from https://www.faa.gov/regulations_policies/rulemaking/recently_published/media/2120-AJ60_NPRM_2-15-2015_joint_signature.pdf

[87] Nicas, J. (2015, March 24). Amazon Says FAA Approval to Test Delivery Drones Already Obsolete. Retrieved from http://www.wsj.com/articles/amazon-says-its-approval-to-test-delivery-drones-already-obsolete-1427219327

[88] CFR 14, § 91.113.

[89] Park, C., Lee, S., & Mueller, E. R. (2014, June 20). Investigating Detect-and-Avoid Surveillance Performance for Unmanned Aircraft Systems. *14th AIAA Aviation Technology, Integration, and Operations Conference*.

[90] RTCA. (n.d.). SC-228 Minimum Operational Performance Standards for Unmanned Aircraft Systems. Retrieved from http://www.rtca.org/content.asp?contentid=178

[91] Swider, C. (2014, April 9). RTCA SC-228 & DAA Standards Development.*ICNS Conference*.

[92] RTCA. (2013). *TERMS OF REFERENCE RTCA Special Committee 228 Minimum Performance Standards for Unmanned Aircraft Systems* (Publication No. No. 109-13/PMC-1089). RTCA.

[93] FAA. (2013). *Integration of Civil Unmanned Aircraft Systems (UAS) in the National Airspace System (NAS) Roadmap* (United States, Federal Aviation Administration).

[94] RTCA. (2013). *Minimum Operational Performance Standards for Global Positioning System/Satellite-Based Augmentation System Airborne Equipment* (Tech. No. DO-229D). RTCA SC-159.

[95] ATSM. (2005). *Standard Specification for Design and Performance of an Airborne Sense-and-Avoid System* (Publication No. F2411-04e1). Retrieved http://compass.astm.org/download/F2411-04E1.14356.pdf

[96] Hottman, S. B., Hansen, K. R., & Berry, M. (2009). *Literature Review on Detect, Sense, and Avoid Technology for Unmanned Aircraft Systems* (Rep. No. DOT/FAA/AR-08/41). Washington, DC: Federal Aviation Administration.

[97] Euteneuer, E., Papageorgiou, G., & Honeywell. (2011, October). *UAS Insertion into Commercial Airspace: Europe and US Standards Perspective* (Rep.).

[98] Narinder, T., & Wiegmann, D. (2011). Analysis of Mid-Air Collisions in Civil Aviation. In *Proceedings of the 45th Annual Meeting of the Human Factors and Ergonomics Society*. Retrieved from http://pro.sagepub.com/content/45/2/153.full.pdf

[99] Higgins, C. R., & Higgins, J. (n.d.). The Role of Human Factors in Improving Aviation Safety - Text Only. Retrieved January 15, 2015, from http://www.boeing.com/commercial/aeromagazine/aero_08/human_textonly.html

[100] Lester, T., Cook, S., & North, K. (2014, January 31). *USAF Airborne Sense and Avoid (ABSAA) Airworthiness and Operational Approval Approach* (Rep. No. MTR130384).

[101] Hottman, S. B., Hansen, K. R., & Berry, M. (2009). *Literature Review on Detect, Sense, and Avoid Technology for Unmanned Aircraft Systems*(Rep. No. DOT/FAA/AR-08/41). Washington, DC: Federal Aviation Administration.

[102] Kerczewski, R., Wilson, J., & Bishop, W. (2013). *FREQUENCY SPECTRUM FOR INTEGRATION OF UNMANNED AIRCRAFT* (Rep.). 32nd Digital Avionics Systems Conference.

[103] *Statement of Dr. Edgar G. Waggoner before the Committee on Science, Space and Technology U.S. House of Representatives* (2015) (testimony of Dr. Edgar G. Waggoner).

[104] United States, NASA, AMES Research Center. (2014, September 3).*UNMANNED AERIAL SYSTEM (UAS) TRAFFIC MANAGEMENT (UTM)*. Retrieved from https://prod.nais.nasa.gov/eps/eps_data/162405-OTHER-001-001.pdf

[105] Mulgaonkar, Y., Whitzer, M., Morgan, B., Kroninger, C. M., Harrington, A. M., & Kumar, V. (2014). Power and weight considerations in small, agile quadrotors. *Proc. SPIE 9083, Micro- and Nanotechnology Sensors, Systems, and Applications VI.*

[106] Mulgaonkar, Y. (2012, May). *Automated recharging for persistence missions with multiple micro aerial vehicles* (Unpublished master's thesis, 2012). University of Pennsylvania. Retrieved from https://fling.seas.upenn.edu/~yashm/website/files/YashThesis.pdf

[107] Barghamadi, M., Kapoor, A., & Wen, C. (2013). A Review on Li-S Batteries as a High Efficiency Rechargeable Lithium Battery. *Journal of the Electrochemical Society, 160*(8), A1256-A1263. doi:10.1149/2.096308jes

[108] Bakenov, Z. (2011). LiMnPO4 Olivine as a Cathode for Lithium Batteries. *The Open Materials Science Journal, 5*(1), 222-227. doi:10.2174/1874088X01105010222

[109] Doom, J. (2015, April 21). Cnano Technology Gets $15 Million to Expand Nanotube Production. Retrieved from http://www.bloomberg.com/news/articles/2015-04-21/cnano-technology-gets-15-million-to-expand-nanotube-production

[110] Next Big Future: Hybrid gas electric drone has 13 times the range of a battery electric drone. (2015, March 19). Retrieved from http://nextbigfuture.com/2015/03/hybrid-gas-electric-drone-has-13-times.html

[111] Toksoz, T., Redding, J., Michini, M., Michini, B., How, J. P., Vavrina, M. A., & Vian, J. (2011). *Automated Battery Swap and Recharge to Enable Persistent UAV Missions*. St. Louis.