Fly Sense

Team C

Nicholas Crispie, Shivang Baveja, Sai Nihar Tadichetty, Harikrishnan Suresh, João Fonseca Reis
Advisors: Vishal Dugar, Sebastian Scherer
Mentors: Dimi Apostolopoulos, John M. Dolan

29th September 2017

Table of Contents

Table of Contents

1. Project Description

Helicopter pilots have one of the toughest jobs in the world. Their jobs are usually task and sensory saturated, with limited ability to process new information and many different controls to be used at an instant. However, there aren't many aides for helicopter pilots that present useful information in a relevant way. The U.S. military has invested millions of dollars in state-of-the-art headsets for conveying all sorts of information to fighter pilots in real-time, but nothing close to that technology has been introduced in the commercial domain given the current price point and the focus on assisting firing and targeting systems.

Helicopter pilots face difficulties in different phases of flight and mission types. Some of these are low-altitude flights, landing in tight spaces with fixed structures and navigation in low visibility scenarios. Out of the listed flight stages, one of the most critical is flight at an altitude below 200ft AGL (Above Ground Level) [1] where, unlike commercial airplanes, there is no autonomous piloting features in place to aid with landing.

Helicopter pilots resort to their instruments, but above all look for landmarks they know the size of to judge how far they are from obstacles. Nonetheless, this can be even more difficult when flying in unfamiliar environments, in areas where the landscape is monotonous (e.g. desert, or a grass field) or in situations where it's hard to judge obstacles that can cause a crash (e.g. a pole near the tail rotor).

The project's focus is to assist helicopter pilots in the difficult flight phases mentioned above by creating a system that gives needed information to the pilot in the least intrusive way. The project aims to bridge the gap between the risky status quo and the technologies yet to be seen in this domain. We want to develop a core set of versatile technologies that accomplish a few of the important objectives, outlined in Figure 1.2.

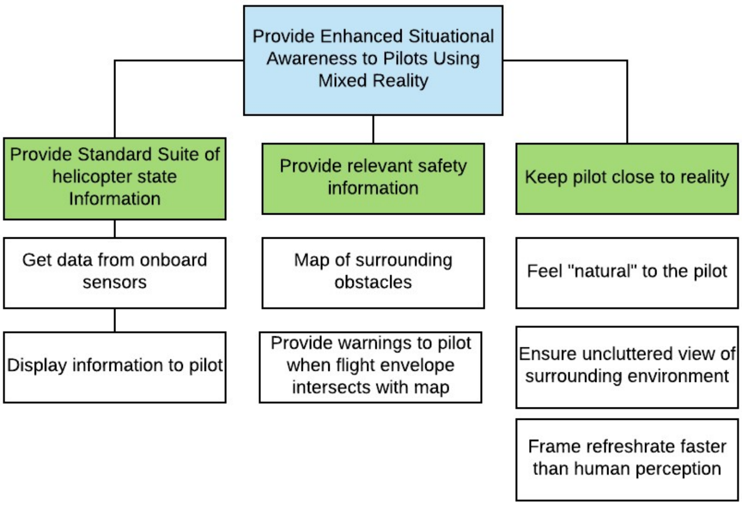


Figure 1.1: Objectives tree for FlySense.

2. Use Case for FlySense

Spencer is a search and rescue helicopter pilot working out of the Rostraver Airport just outside of Pittsburgh, PA. Today reports are trickling in about some significant damage made to a few buildings by fallen trees from thunderstorms, but it's unclear whether there were any personal injuries that need to be transported by helicopter. If there have been such occurrences in remote areas, Spencer will for sure be called in any time soon.

Just as he has grabbed a cup of coffee, a dispatcher gives him a call. "We need a helicopter in Cranberry Township right away! We have a TBI (Traumatic Brain Injury) and extensive bleeding at a construction site. I'm passing off the GPS coordinates to you right now."

encer grabs his gear from his locker and walks briskly out to the helicopter. Upon reaching the helicopter, he does a quick walk around and proceeds to the cockpit for his preflight checks. He grabs his FlySense visor and turns it on. Within seconds the headset boots up. After a 10 second calibration procedure, a couple options pop up. With a quick wave of the hand, Spencer selects a trip planning view.

Spencer reads off the coordinates that he was given "40 point 6858 degrees North, 80 point 1032 degrees west." and selects it on the map. Spencer then turns on a 360-degree view of the helicopter's surroundings by telling the flight computer "Bird's Eye View" and begins his takeoff procedure. It's a good thing he is in the parking warnings mode. In their haste to get things ready, a ladder was left by the crew. Spencer gets a warning that the ladder is there, and alerts the ground crew to clear it out before he lifts off.

Once Spencer over 200 ft, he can engage the autopilot. This makes his job a lot easier, but his FlySense display is still active on "Standard Instruments" view. This gives him constant updates on his attitude, altitude, and a view of the horizon, even as he goes through a couple low hanging fog banks. It also shows him where other flying vehicles are located and what path to take to his destination all within the "FPV Overlay" view.

Figure 2.1: The tight spot where Spencer needs to land (Source: vertikal.net)

All goes smoothly until he gets to the desired landing spot. On a quick circle around, Spencer sees many potential problems. It's a tight fit on the flat section of ground next to the construction site, with a lot of trees, with some large branches strewn around. Matters are made worse by a crane leaning over the area, which was probably knocked down by the high winds. Spencer figures he's going to have to back in underneath the overhanging crane, a very dangerous operation.

He switches from FVP Overlay to Bird's Eye View mode with a quick verbal "Bird's Eye" command. Once he descends to 200ft in altitude, the mapping system in the helicopter within seconds has mapped out the entire area and immediately starts to provide feedback in the visual display. He goes down right above the tree level, and slowly backs in towards the landing spot he's picked. Suddenly, the Bird's Eye visuals start flashing in red. He's getting too close to the crane right below the helicopter. He rotates the helicopter slightly to the left and the warnings go away. He backs in a little farther and drops down to the ground.

When the patient is safely inside the helicopter, Spencer then tells the FlySense system "Map mode…. UPMC" and the system switches to a map mode showing the overall direction he needs to go and what vector to use while approaching the hospital. Once in a stable position at a high enough altitude, he engages Autopilot and the helicopter flies off towards the hospital.

As Spencer gets closer to the medical center, there is much more air traffic in the area. He gets warnings and recommended adjustments to his route through the map and path planning interface, with vectors around the display showing the relative positions of other aircraft nearby. Spencer adjusts the trajectory, slowing down and hovering to allow another helicopter clear out of his approach path at the hospital. It seems like he's not the only one taking a patient to the hospital for treatment today. Now with a clear path, he is able to make his approach smoothly and safely, getting the patient directly to the helipad, where medics rush out to give the patient treatment. **Welcome to FlySense!!!**

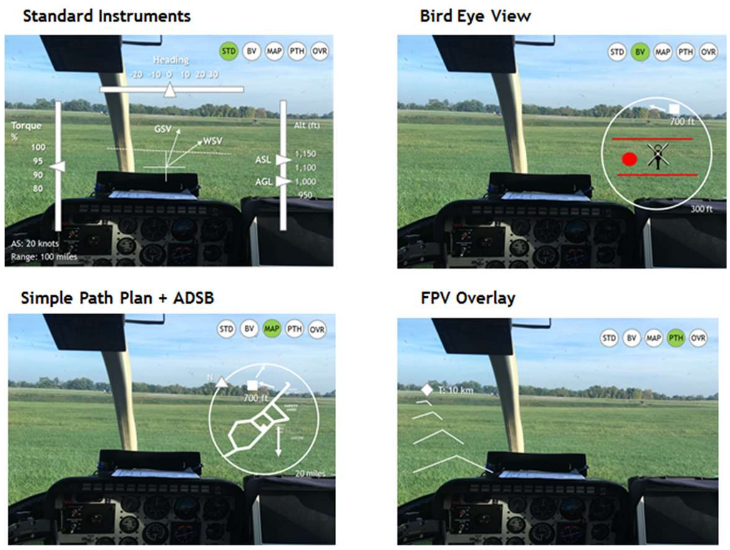


Figure 2.2: FlySense Look and Feel

3. System Level Requirements

In this section system requirements are presented and segmented according to priority (mandatory, mandatory for testing purposes and desirable).

3.1 Mandatory requirements for end-solution testing (in a helicopter)

Functional Requirements:

ID	Description	Target Performance
MR.01	"Standard User Interface" (typically to be used before pilot is airborne with hand gestures as interface)	User can turn features on/off, change key display configuration (e.g. size, position,)
MR.02	Bird's Eye View: Give Warnings for all obstacles in immediate flight envelope	Plot all obstacles in the flight envelope within x seconds of impact
MR.03	Bird's Eye View widget is automatically turned on/off	When obstacles are within/outside y seconds of impact (distance displayed in feet)
MR.04	Standard Instruments: Give Direct access to standard data displayed on the HUD headsets in the market	Ensure pilots can cruise without looking at the "physical" instruments dashboard

Non-functional Requirements:

MR.05	Easy to set up before flight is initiated	< 2 minutes to set up
MR.06	Refresh frame rate above human eye	25 Hz Refresh rate
MR.07	Low latency processing information	< .4s latency
MR.08	Display obstacles accurately	Identify obstacle within 5 deg error
MR.09	Readable display in multiple lightning (darkest night / brightest day)	Information readable in closed room and under incidence of a x lumen of light
MR.10	Capable of processing data in a rapidly changing environment	Process information while travelling at up to y knots
MR.11	Pilots can wear solution comfortably for extended periods of time.	AR Headset weight below 1 pound

3.2 Mandatory Requirements for Intermediate testing (before helicopter testing)

Requirements marked as "Mandatory for testing purposes" are not part of the final solution but needed to be tested in detail before the solution is deployed in a real-life scenario inside a helicopter.

ID	Description	Target Performance
MRIT.01	Perform full state estimation	Altitude, speed, attitude, position within x% accuracy

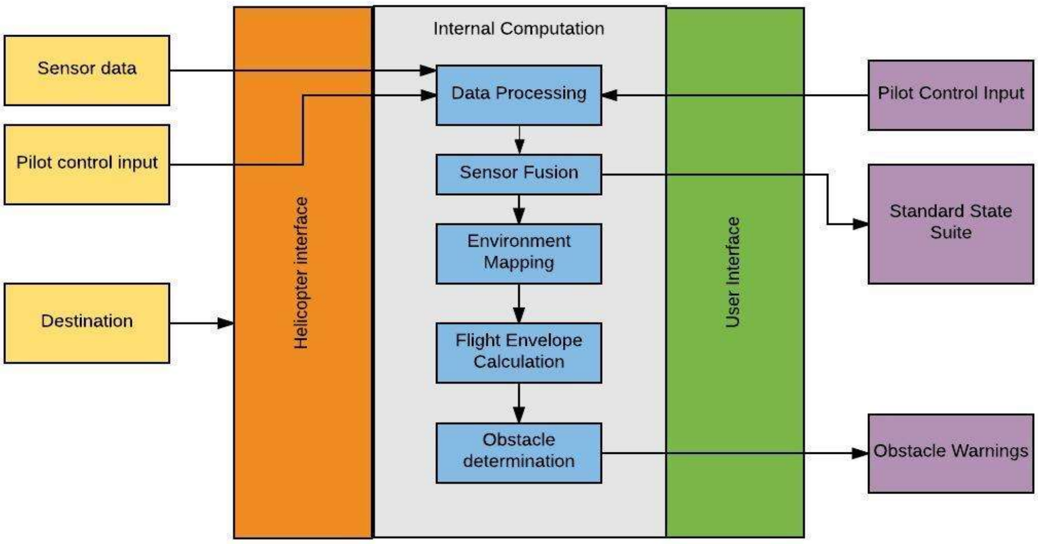
3.3 Desired Functional Requirements (non-mandatory)

The following requirements are not mandatory and will only be pursued after the mandatory requirements are achieved.

ID	Description	Target Performance
DR.01	"**Quick Access User Interface**" (use case: airborne period)	Pilot can turn features on/off (but not customize them)
DR.02	**Simple Path Plan + ADSB**: Define a destination using maps overlaid with air traffic to choose destinations	Present all aircraft within x miles radius. Accepts one set of coordinates and display approach vector for the target destination
DR.03	**FPV Overlay:** Plot obstacles/ADSB information on a first-person view	Plot all obstacles in flight envelope within x seconds of impact
DR.04	**Full Path Planning**: Allow for full path planning capabilities (expanding the previous requirement)	Plots full path to destination including adequate cruising altitude to avoid all obstacles ensuring a safety envelope of x seconds

Note: Requirement DR.01 is not mandatory at this phase given that only the "Standard Instruments" and the "Bird's Eye View" requirements are mandatory at this stage. "Standard Instruments" are typically turned on while still on the ground and "Bird's Eye View" is automatically turned on when relevant.

4. Functional Architecture



The above figure captures the basic function of the sub-systems on a high level. The system is divided globally into three main parts:

1) Helicopter interface: Interface to the Helicopter flight computer for sensor data, and pilot inputs.
2) Processing: Generate obstacle map, estimate state, generate warnings, etc
3) User interface: communicating with pilot (inputs and outputs)

Each of the three parts described above has two components relating to our two most mandatory features:

a) Conveying the standard set of aviation information to the pilot, from sensing to display
b) Communication of potential hazards, again from sensing to display

5. System Level Trade Studies

Trade studies were conducting for following major subsystems:

1. On-board computer
2. Sensing system
3. AR headset – Controller
4. User Interface

Every component listed above plays a crucial role in the project's functionality. Every parameter that has been used for comparison has been carefully discussed and evaluated by the whole team along with valuable inputs from key stakeholders.

5.1 FlySense On-Board Computer

Our objective is to develop a system which goes on-board the helicopter. The key criteria for selecting the processing unit is its processing capabilities (GHz) and memory (GB) given that we want to have as much richness of information coming from the sensors as possible.

Lastly, since our project is to enhance a pilot's sensing capabilities, it must be reliable. It must work in any condition, process the data without any lag and never get stuck.

Below are the scores of each of the components in our trade study for FlySense onboard computer:

	Products / Weights	Raspberry Pi 3	BeagleBone Black	Jetson TX1	Odroid-C2	Asus-Tinker-board
Processing capability	20%	2.3	1.9	3.3	3.8	5.0
RAM	20%	1.3	0.6	5.0	2.5	2.5
Reliability	20%	1.7	3.3	1.7	5.0	5.0
Programming Support	15%	1.0	2.0	5.0	3.0	4.0
Ports (LAN,…)	10%	5.0	5.0	5.0	5.0	5.0
Form Factor	5%	2.9	2.9	0.0	2.5	2.9
Technical Support	5%	1.0	2.0	5.0	3.0	4.0
Serviceability	5%	5.0	1.3	2.5	1.3	2.5
Total	**100%**	**2.1**	**2.3**	**3.6**	**3.6**	**4.1**

5.2 Sensing System

Our system utilizes data from a sensor suite which includes GPS, Magnetometer, IMU, Barometer and LIDAR. LIDAR plays an important role and is by far the most expensive component.

	Product	SONAR	LIDAR			Stereo	RGB-D Camera	
	Weights	Maxbotix	SICK LD-MRS	Velodyne PUCK VLP-16	Hokuyo UTM-LX	Zed	Kinect	Asus Xtion-Pro
Range	20%	0.2	2.5	5.0	0.8	0.1	0.1	0.1
Environmental Robustness	20%	2.0	5.0	5.0	4.0	1.0	2.0	2.0
Richness of Information	15%	3.0	2.5	2.5	3.0	5.0	3.5	3.0
Field of View (vertical)	15%	0.9	0.2	1.9	0.0	5.0	4.4	2.8
Field of View (horizontal)	15%	0.2	5.0	5.0	3.8	1.5	0.8	0.8
Cost	10%	5.0	0.0	1.9	3.1	4.8	4.5	4.5
Weight of Product	5%	2.3	0.0	0.5	4.0	4.0	1.8	3.5
Total	**100%**	**1.7**	**2.7**	**3.6**	**2.5**	**2.6**	**2.3**	**2.0**

The most important features on the LIDAR equipment are range, field of view and robustness. The 3D obstacle mapping provides pilots with complete situational awareness during various flight scenarios like landing in constrained spaces, flying in low visibility environments etc. As the helicopter might have to encounter extreme weather conditions like low light or heavy rain, the 3D mapping sensor must be accurate and robust.

Based on the trade study, it is evident that the Velodyne Puck VLP-16 LIDAR sensor outperforms the others in all the main criteria. Drawing conclusion from the trade study, we plan to use a Velodyne LIDAR based on the availability from our sponsor.

5.3 AR Headset – Controller

Our system heavily depends on the quality of user experience, for this we absolutely need to make sure that there are no hiccups in the setup or visualization process. We have zeroed in on the capability and ease of programming as the most important factors.

Parameters	Weights	Microsoft Hololens	Google Glass	Vuzix	Meta 2.0	Recon Jet	Optivet ORA	Epson BT300
Capability	**25%**	5.0	2.0	2.0	4.0	3.0	2.0	2.5
Ease of Programming	**25%**	5.0	4.0	1.5	5.0	1.5	1.0	4.0
Cost	**10%**	4.1	0.3	2.5	0.3	3.4	0.0	2.5
Reliability	**10%**	5.0	4.4	3.3	5.0	5.0	2.8	4.4
Weight	**10%**	0.0	4.7	4.6	0.7	4.3	4.2	4.2
Hand Tracking	**10%**	5.0	5.0	0.0	5.0	0.0	5.0	5.0
Head Tracking	**10%**	5.0	2.5	5.0	5.0	0.0	2.5	5.0
Total	**100%**	**4.4**	**3.2**	**2.4**	**3.8**	**2.4**	**2.2**	**3.7**

Conclusion: Hololens seems the best candidate but we need to validate its performance working inside the cockpit of a helicopter.

5.4 User Interface

While deciding which hardware to procure for the project, it is essential to decide how to proceed into the development phase, it is equally important to do studies on how the end user will interact with the product. This study gives us futuristic insights into what we might be missing during the planning phase, this is a valuable result that might save a lot of time and effort as we can have the complete picture of how to build the product while keeping the end user in mind.

We generalized the term ‘trade study’ and implemented it to decide how the pilot uses our system. There are a certain set of modes that allow the pilot to interact with the system, we have finalized these based-on feedbacks from the pilot workshop and other key stakeholders. The whole point of this study is to minimize pilot’s cognitive needs and keep him focused on doing his job more efficiently than before.

Virtual Air Grab: The pilot will be grabbing icons "out of thin air". While this is easy while the helicopter is on the ground, it is not feasible to use while the helicopter is in the air where the pilot is already cognitively saturated. Therefore, a lower score.

Buttons: This is a classical robust way to interact with any system. Obviously, this is very reliable but the pilot is already burdened with a million buttons on his dashboard. This also goes against the main idea of our project which is to reduce the cognitive load, so, the last thing we would want to do is add more buttons. Therefore, a lower score.

Smart Gloves: This is an interesting way to solve this problem, we are still considering and once we have all the technical know-how, we will decide on this.

Voice Commands: This is the simplest solution to the problem; the pilot can interact with system with just his voice commands. There are some issues with reliability, which might be solved (we are still considering the technical aspects). It just needs a small microphone, so therefore, high overall score.

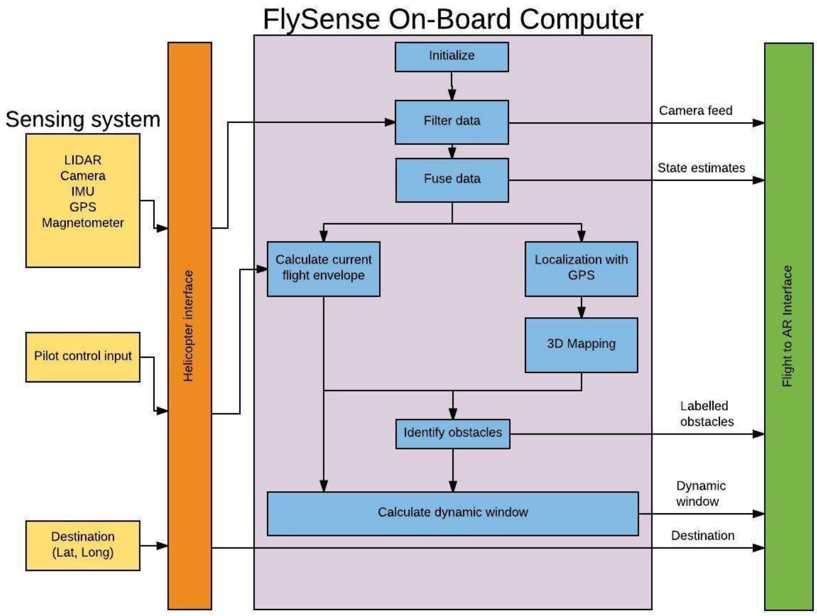
Eye Tracking: This is a very interesting solution with a lot of complexity in implementation and lacks reliability. After discussion, it was given the lowest score and we decided that we will not be pursuing it.

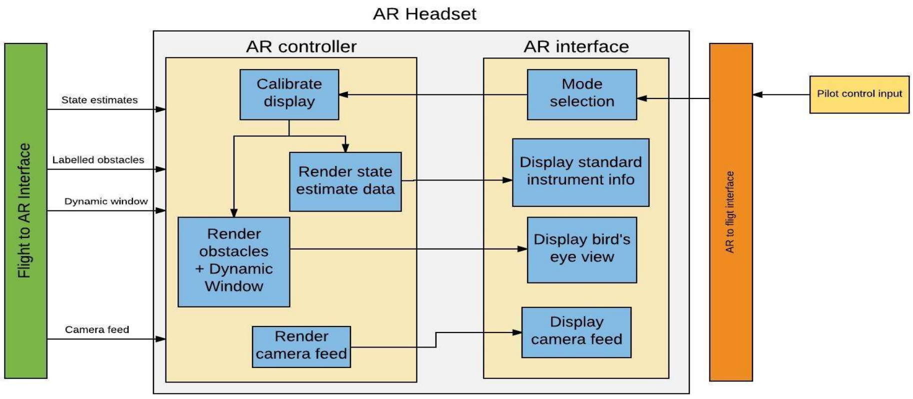
Figure 5.4 Trade Study on AR Interface Design

	Ease of Use	Reliability	Ease of Integration	Development Resources	**Total**
VIrtual Air Grab	2	4	5	3	**14**
Buttons	2	5	2	4	**13**
Smart Gloves	4	3	4	2	**13**
Voice Commands	5	2	4	3	**14**
Eye Tracking	1	2	1	1	**5**

Conclusion: No clear resolution on what option to choose going forward.
Future Action: Conduct additional research on the options available to delineate some of the numbers, as well as generate concepts and put those concepts in front of pilots. We will be designing concept walk-throughs to get specific feedback on what works best for pilots.

6. Cyber-Physical Architecture





The FlySense system is shown as two major systems. These are the FlySense Onboard Computer and the AR Headset. The functions carried by them are detailed in the sub-system description section.

Further the architecture shows the flow of data between the two systems. This interface is called Flight to AR interface. The AR Headset is receiving following data from the Onboard computer:

a) State Estimate
b) Camera Feed
c) Labelled Obstacles Map
d) Path Information

The Helicopter interface is shown separate as we will be receiving the sensor data, pilot control inputs from the Helicopter flight computer.

7. Subsystems Description

7.1 Sensing system

The sensing system contains the following components:

- LIDAR for 3D point cloud information
- Camera for live video feed during operation (TBD)
- Inertial Measurement Unit for determining attitude, angular rates and specific force acting on the helicopter
- GPS for localization
- Magnetometer for determining heading of helicopter

Along with the sensors, the pilot's control inputs and the target destination (latitude and longitude) are provided to the onboard computer.

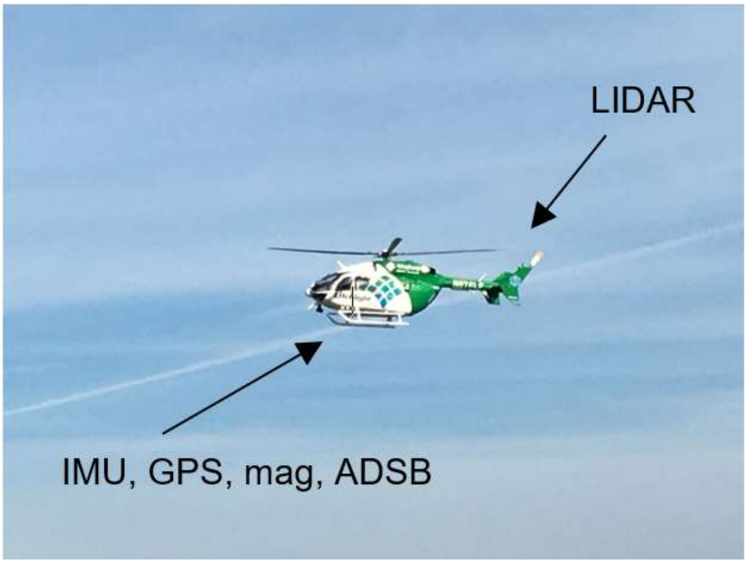


Figure 7.1: Location of each sensing modality

7.2 FlySense On-board computer

The on-board computer performs five major functions:

- State estimation: The data coming from the sensing system is processed to extract valuable information. This information coming from various sources is then fused together in an estimator using one of the following algorithms (Kalman Filter, Bayesian Networks etc.) to obtain the position, attitude, velocity, and acceleration of the helicopter and sensor errors.

- Mapping and Localization: Localization is performed completely using GPS and is used to build a 3D map of the surrounding area. 3D maps are built using the point cloud information and updated in real time based on the localization information.
- Flight envelope calculation: The control inputs from the pilot, the estimated states and underlying dynamics of the helicopter are used to calculate the current flight envelope.
- Obstacle identification: Based on the 3D map, the obstacles in the environment are identified and the environment is segmented into different zones (safe, alert, and dangerous). Obstacle identification is of primary interest here, especially to address the tail rotor issue. Environment segmentation is planned only for the future upgrades like overlay and flight path planning.
- Dynamic window calculation: Based on the current flight dynamics, the dynamic window is calculated and obstacles inside the dynamic window are labelled based on proximity.

7.3 Augmented Reality Controller

The Augmented Reality Headset takes input from the on-board computer and the pilot and provides the desired functionality. It contains two major components:

- The controller detects when the pilot has made a selection
- Based on the mode selection by the pilot, the display is calibrated.
- For the standard instruments mode, the information from state estimator is rendered for display.
- For the bird's eye view, the calculated dynamic window and identified obstacles are converted to top view mode and rendered.

7.4 AR interface segmented by functional requirement

The major function of the interface is to provide relevant information to the pilot in a seamless manner. The overlay will be robust with respect to various operating conditions and environmental factors.

Standard User Interface (MR.08)

Standard user interface is done through hand gestures interacting with the virtual reality features. The user should be able to both turn features on/off and to customize the way they are presented on the screen in terms of size and orientation. By default, all widgets are turned off to ensure uncluttered view.

Whenever a pilot "touches" a feature that is not yet turned on, the control radio buttons show be illuminated and the featured turned on. Should the feature be a first person overlay, it should be overlaid on top of the reality as the pilot sees it. Should it be a widget, it should pop up in the screen in the latest relative location with the latest relative size introduced.

When a pilot "touches" a feature that is currently turned on, the control radio buttons show be turned off together with the feature.

Note: This interface can only be used effectively and safely before the pilot is airborne as interacting with it implies removing one of the hands from the helicopter controls (unlike the case of an airplane, a helicopter pilot controls two different levers with the hands: the cyclic to control the angle of attack of the rotor blades and the throttle to control the rotor power). In addition, pilot feet are also used to control the hovering pedals.

Quick Access User Interface (DR.01)

This interface is to be used during flight to turn features on/off and does not have the ability to change the presentation parameters of a given feature. To design it properly, we will have to do simulations/mock-ups/focus groups with pilots in order to analyze in detail what would be the least intrusive way of doing this. In addition, the solution needs to be reliable.

The following technologies are to be studied in detail (all of them have potential upsides and downsides that need to be validated extensively and that we cannot do now):

a) Additional buttons on the cockpit
b) Smart gloves on the pilot hand
c) Iris tracking to point to widget
d) Simple voice commands

Standard Instruments (MR.10)

This feature is needed to ensure the solution is "acceptable" to the pilots and needs to be in line with the standard practices of HUD displays currently in the market. Standard instruments should allow pilots to cruise without the need to look at any of the currently physical instruments.

The system will receive from the helicopter computer the cleaned state estimation data for:

a) Navigation: Heading vector, current altitude, average altitude above ground
b) Dynamics: Ground Vector Speed (GVS) and Wind Vector Speed (WVS), Air Speed (AS)
c) Performance: Engine Torque, estimated range left (with current conditions)

Note: Whenever the wind is blowing from behind the tail rotor effect can be offset and the helicopter can start to spin.

Bird's Eye View (MR.09)

The "Bird's Eye View" receives LIDAR data (or alternative source) from the helicopter computer and displays it to the pilot in a 2D format resembling the systems currently used for car parking.

Unlike the case of a car parking aid, a helicopter travels in a 3D environment and the range of speeds where this command can be useful is substantial as it can be useful cruising and not only parking a helicopter. To cater to this, the position of the helicopter symbol and the scale of the map will be managed dynamically:

a) Position of the symbol inside the widget: the helicopter symbol will be moved in opposite direction to increase the field of view in the direction of the movement (e.g. if the helicopter is going forward the symbol will be moved backwards in the widget)
b) Managing the dynamic window for what will be displayed:
   i) In the plain of the movement the selected area will be defined by a constant (x seconds of travel) times the current speed.
   ii) In the orthogonal direction, it will be proportional to the distance that can be travelled with the maximum pilot input.

iii) In both planes there will be a minimum distance threshold so that data is still displayed when the helicopter is parked.

c) Coloring code: obstacles fairly close (e.g. 5 seconds before contact) will be presented in red with colors than evolving proportionally into yellow and white

Note: In addition to the graphical information, 3D sound aids may be deployed to better assist pilots in identifying the nearest object (above/below, left/right, front/back).

Simple Path Plan + ADSB (DR.02)

The simple path planner is widget map where the target destination can be selected either by GPS coordinates or by tapping on the map. It should also be able to accept an approach vector to that destination and to show current air traffic around the destination.

The map and GPS coordinates will come from an off-the-shelf tool (either online or offline), while the air traffic positions and speeds will come the ADSB through the helicopter computer.

After the destination and approach vector have been introduced, the pilot should be able to follow them in the “Path” feature.

FPV Overlay (DR.03)

The First-Person View overlay allows the pilot to track obstacles in a manner consistent with the direction of the current gaze. The data displayed is received from the helicopter computer and the following information will be overlaid in sync with reality taking into account the pilot gaze and the relative position:

a) Destination: location in the horizon with distance overlaid
b) Airborne vehicles: location in the horizon with difference in altitude overlaid
c) Obstacles in solid angle: with color coding dependent on dynamic window and only showing objects in yellow or red directly overlaid in current view
d) Obstacles outside solid angle: standard symbol to be displayed at the extremes of the image to catch the attention of the pilot to look up/down, left/right

This feature does not address directly the most urgent need identified in the interviews, but can also be used instead of the bird's eye view for parking a helicopter. While it will be a crucial feature in future evolutions, it is not mandatory for this phase.

Full Path Planning (DR.04)

This feature is an evolved version of the path planner that will include obstacle information in order to generate a path free of obstacles (e.g. including cruising altitude). It will be detailed in the future as this is not a mandatory requirement and is extremely complex.

8. Project Management

One of the major risk to the project as mentioned in the risk management section is availability of an actual helicopter for final integration and testing. To mitigate that risk, the team has developed two plans which coincide till end of the fall semester and diverge for the spring semester. This would give the team time to work out the availability of helicopter with the sponsor (Near Earth Autonomy) without deviating from the project requirements.

Plan A (if Helicopter is confirmed to be available for spring validation)

The team will work focused towards helicopter integration from the beginning. This will involve carefully studying the helicopter interfaces and developing our system to match them, and ensuring all the sub-system would work in flight. The goal here would to test a first prototype in the helicopter by mid-March and from there work on further improvement and testing to make it as robust as possible by spring validation.

Plan B (if helicopter is not going to be available for spring validation)

The team will work focused on integrating the system on a quadcopter to validate the requirements. To simulate helicopter flight, a light weight tail (or a virtual tail) will be attached to the quadcopter. The system would be developed keeping in mind some of the helicopter constraints but actual integration can only be done after completion of the project.

Our team will be selecting one of the above-mentioned plans by the end of Fall semester after we have gathered more information, in particular at the level of commitment that our sponsor is willing to give to this project.

The two plans overlap more than 70% implying that the switching costs from one approach to the other will be limited, although we would like to focus only in one of the approaches from the beginning to avoid unneeded rework and focusing our efforts towards the final goal. All the major tasks identified have been divided into 4 tracks which will go in parallel.

8.1 Work plan

I. Augmented Reality controller (“Standard Interface”)
   a. Setup of the FlySense system
   b. Real-time Heads-up display
   c. Hand movement detection
   d. Head movement detection
   e. Widgets enable/disable by pilot
   f. Widgets customization by pilot
   g. Standard instruments view
   h. Bird’s-eye view
   i. Interfacing to onboard computer.

II. Sensing
   a. Testing: Real time sensor interface (Inertial, Altimeter, GPS & Magnetometer).
   b. Testing: State estimation
   c. 3-D map generation using Ros package
   d. Bird’s-eye view transformations
   e. Making it robust for aerial vehicle integration
   f. Real time sensor interfacing through helicopter flight computer

III. UI/UX
   a. Standard user interface (based on Pilot interviews and other criterias).
   b. 3d sound warning for object proximity
   c. Develop mock-up/simulation (TBD) to validate the design with stakeholders.
   d. Alternative quick user interface (non-mandatory: TBD)

IV. System Integration and testing
   a. Procurement of sensor and equipment
   b. Development of test support systems (Fall & Spring validation experiments)
   c. Integration of AR headset, Sensing systems and UI/UX.
   d. Unit level and system level tests to improve the accuracy and performance.
   e. Track mandatory requirements to drive development towards compliance.

Fall 2017 schedule

Tracks	PR 1 20 Oct 2017	PR 2 27 Oct 2017	PDR 31 Oct	PR3 Nov 10	PR4 Nov 22	PR5 Nov 30	PR6 Dec 7	Critical Review Dec 14
AR	First visualization of sensor suite	Head tracking for alignment	Interface to onboard computer	Bird's Eye view	Interface to onboard computer	Bird's Eye with real-time LIDAR and ATT	Additional testing and refinement	Planning for next phase
UI/UX	First draft		Mock-up /simulation (TBD)		Design Phase -1 complete			
Sensing	Point cloud of Lidar Data set	Sensor interfacing on onboard computer	Interface to AR headset	3D Map Phase 1	Interface to AR			
System Int. & Testing	Procurement	Design of Fall verification experiment	Integration	Testing HUD	Integration	Testing 3D Map and HUD		

Progress Review 1: 20th of October

- Procure IMU, GPS, LIDAR and Onboard computer.
- First visualization of Heads-up display with fake sensor data.
- Use data from Lidar data set to create and visualize point cloud.
- First draft of UI/UX.

Progress Review 2: 27th of October

- Projection Alignment on AR
- Sensor (LIDAR, IMU, MAG and GPS) interfacing on onboard computer.
- Develop the apparatus for Fall validation experiment.

Spring 2018 schedule

<table>
<tr><th>Tracks</th><th>31 Jan 2018</th><th>28 Feb 2018</th><th>31 March 2018</th><th>30 April 2018</th></tr>
<tr><td>AR</td><td>Hand gesture detection
Head tracking</td><td>Widget setup
Widget enable/disable</td><td rowspan="3">Fixing issues seen during integration and testing</td><td></td></tr>
<tr><td>UI/UX</td><td>Sound generation based on obstacle distance</td><td>3-d sound generation</td><td></td></tr>
<tr><td>Sensing</td><td>Interfacing to Flight computer
(sensor data)</td><td>3-d map generation and flight simulation in rviz</td><td></td></tr>
<tr><td>System Integration and Testing</td><td colspan="2">Spring validation experiment development (Quadcopter/helicopter/Helicopter simulation)</td><td>Integration of all the subsystems and unit-level testing</td><td>Full System Testing</td></tr>
</table>

8.2 System Validation experiments

Fall Validation Experiment

Objective	Demonstrate FlySense system is capable of giving information about the surrounding obstacles real-time using Augmented Reality.
Elements to be tested	Augmented Reality Headset, FlySense onboard computer, Lidar, other sensors (IMU, GPS, MAG)
Equipment	A person sitting in a chair (or Standing) with AR headset on. Lidar and other sensor would be mounted on the chair (on a pole in his hand). As chair (or the person) moves around, the person should be able to view a bird's-eye view of the surrounding obstacles.
Location	An open area with some obstacles and decent GPS reception.
Procedure	a. Switch on power to the onboard computer and sensor. b. Person sits on the chair (or picks up the system) and puts on the AR headset c. The Person rolls the chair around (or moves) backwards towards obstacles. d. He/she can see his location and orientation with respect to obstacle and how close he is to bumping into it. e. Person in chair is moved back and forth and side to side, responds to questionnaire on what he/she sees
Verification Criteria	a. The Person should be able to prevent collision with the obstacles b. He/she should be able to view the information within maximum lag (.4s)

Spring Validation Experiment (Plan A)

Objective	Demonstrate complete functionality of FlySense as a helicopter pilot assistance system in field test (or Flight Simulator)
Elements to be tested	AR headset, FlySense OnBoard computer, 3d obstacle map generation, visual warning and audio warning
Equipment	Helicopter mounted with LIDAR and other sensors or (Flight Simulator for Helicopter), AR Headset, Onboard computer
Location	Flight testing location of NEA (or flight simulator) (TBD)
Procedure	1. AR and onboard computer system is powered up 2. Pilot completes the preflight checks 3. Pilot wears the AR Headset and does the setup 4. Pilot checks the HUD display for sensor readings 5. Pilot takes-off the aircraft 6. Pilot enables the Bird's-eye view (along with visual and audio obstacle proximity warning) 7. Helicopter hover near obstacles (within permissible limit) 8. Visual warnings check as aircraft goes slightly near the obstacle 9. Audio warnings check as aircraft goes slightly near the obstacle 10. Pilot comes in for landing and uses bird's-eye view to check how far he is from hangar(or some other obstacle). 11. Lands the helicopter safely 12. Removes the AR headset 13. Switches off Flight Sense system
Verification criteria	1. Setup should work under **1 minute** 2. Heads-up display should display horizon and other information correctly 3. Projection error less than **5 degrees** 4. Bird's-eye view shows top view of obstacles in red or yellow 5. Audio warnings are sounded when obstacle is nearby (vary based on flight envelope) and and warning goes away once obstacles are far (vary based on flight envelope). 6. Check Visual and audio warnings are in sync

Spring Validation Experiment (Plan B)

Objective	Demonstrate complete functionality of FlySense as a helicopter pilot assistance system using quadcopter with a tail as the test platform.
Elements to be tested	AR headset, OnBoard computer, 3d obstacle map generation, visual warning and audio warning
Equipment	Quadcopter mounted with LIDAR and other sensors, AR Headset, Onboard computer
Location	Open area with some obstacles, decent GPS reception
Procedure	1. Power-up the Quadcopter system (MOTORS DISARMED)

	2. Power up the AR Headset, ground control system (communication radios and RC controller) 3. Pilot wears the AR headset and does the setup 4. Check data and camera feed from Quadcopter to AR Headset 5. Preflight checks - Sensor data, communication, GPS accuracy (HDOP<1.0) 6. Motors are Armed 7. Pilot checks the HUD display for sensor data 8. Quadcopter takes-off in LOITER mode. 9. Pilot enables the Bird's-eye view (along with visual and audio obstacle proximity warning) 10. Quadcopter hovers with obstacles nearby on the rear side (within permissible limit) 11. Visual warnings check as quad goes slightly near the obstacle 12. Audio warnings check as aircraft goes slightly near the obstacle 13. Pilot brings in the quad for landing near an obstacle and uses bird's eye view for spatial reference. 14. Lands the quadcopter safely 15. Disarm the motors, power off 16. Pilot removes the AR headset 17. Switch off ground control system (including the AR headset)
Verification criteria	7. Setup should work under **1 minute** 8. Heads-up display should display horizon and other information correctly 9. Projection error less than **5 degrees** 10. Bird's-eye view shows top view of obstacles in red or yellow 11. Audio warnings are sounded when obstacle is nearby(vary based on flight envelope) and and warning goes away once obstacles are far (vary based on flight envelope). 12. Check Visual and audio warnings are in sync

8.3 Team Responsibilities

The tasks identified in the work plan were distributed among the team members based on their expertise and interest.

Role	Primary	Secondary
UI/UX concept	Joao	Nick, Nihar
AR Interface	Nihar	Nick
Sensing Hardware	Hari	Shivang
Mapping	Hari	Shivang, Hari, Nick
Flight Testing Hardware	Shivang	Hari
Flight Software	Shivang	Hari
System Integration	Nick	Shivang
Project Management	Joao	Shivang
Procurement	Nick	Shivang

9.4 Parts List and budget

The project budget will depend substantially on our capability to borrow material that will be need for the project, but that is not directly part of the end solution (e.g. Velodyne LIDAR).

Source	Description	Model	Source	Units	Cost
Borrowed Equipment	LIDAR	Velodyne Puck VLP-16*	Sebastian Scherer/ MRSD Inventory	0	8,000
Borrowed Equipment	AR Headset	Microsoft HoloLens*	Sebastian Scherer	0	3,000
Confirmed Budget	FlySense On-Board Computer	NVIDIA Jetson TX1	NVIDIA	1	500
Confirmed Budget	Camera	RunCam Swift 2	Amazon	1	40
Confirmed Budget	IMU	AdaFruit 9-DOF IMU	www.adafruit.com	1	35
Confirmed Budget	GPS	Hobbyking OSD	Hobbyking	1	43
Confirmed Budget	Magnetometer	SparkFun HMC5883L	MRSD Inventory	1	2
Reserve Budget (applicable Plan B)	Quadcopter	Phantom 2*	MRSD Inventory/ Sebastian Scherer	1	959
Reserve Budget (applicable Plan B)	FPV Goggle	Teleporter V5	MRSD Inventory	1	185

Total cost: 12,763 USD (worst case scenario where all material is purchased for the project)

To be purchase: 1,763 USD (accounting for material that may be borrowed for the project)

9.5 Risk Assessment

Risk for the project was assessed on likelihood of occurrence (1 (low) to 5 (high)) and severity of impact to the project (1 (low) to 5 (high)).

Item	Description	Likelihood of Occurrence	Severity	Category	Mitigation Strategies
RA.1	Difficulty in getting helicopter for testing	3	5	Testing	Backup strategy to test with quadcopter
RA.2	LIDAR sensor sourcing	3	5	Cost	Use less expensive LIDAR system
RA.3	Delay in integration for flight platform	3	3	Time	Build up simple test setup to develop features as much as possible of flight hardware
RA.4	Difficulty in procuring adequate COTS heads-up-display	2	3	Time	Have several options to choose from that we can make work

RA.5	Flight vehicle crashes, damages parts, (especially LIDAR)	2	5	Hardware Failure	Have replacement parts for drone. Low risk in helicopter. Plan flight prior to launch and design test platform to protect sensors from damage.
RA.6	Delays in getting feedback from pilots on user interface	1	3	Time	Pick a design freeze date and stick to that. Engage pilots early and often to get as close as possible to ideal solution
RA.7	Difficulty in scheduling outdoor flight tests (i.e. weather)	2	4	Testing, Time	Target numerous dates from field testing. Test system as much as possible in the lab to maximize field testing time.
RA.8	Mapping difficulties due to sensor limitations	2	5	Hardware Reliability	Add redundant/complementary sensing systems.
RA.9	Lighting variations make it difficult to see visual warnings	3	4	Hardware Reliability	Properly test and research hardware before full implementation. If needed, design mechanical solution for shading.

9. References

[1] Conversation with Spencer Spiker, David Murphy, and Kimber Rugg. NEA, 9/22/17.

[2] T.J. Chong, X.J. Tang, C.H. Leng, M. Yogeswaran, O.E. Ng, Y.Z. Chong, Sensor Technologies and Simultaneous Localization and Mapping (SLAM), In Procedia Computer Science, Volume 76, 2015, Pages 174-179, ISSN 1877-0509

[3] http://velodynelidar.com/vlp-16.html

[4] Product benchmarking research: https://glass.aero/

[5] Research on using AR for a helicopter: http://aviationweek.com/connected-aerospace/watch-can-augmented-reality-help-helicopter-pilots-stay-safe

[6] General AR for a cockpit: https://www.engineering.uiowa.edu/news/augmented-reality-cockpit

[7] Hololens information: https://www.microsoft.com/en-us/hololens/buy

[8] Epson product information: https://epson.com/moverio-augmented-reality?pg=1#sn

[9] Nvidia Jetson Tx1 information: http://www.nvidia.com/object/embedded-systems-dev-kits-modules.html

[10] BeagleBone Black information: https://beagleboard.org/black

[11] ArduPilot product information: http://ardupilot.org/copter/docs/common-apm25-and-26-overview.html

[12] Intel Galileo information: https://ark.intel.com/products/78919/Intel-Galileo-Board

[13] Raspberry pi 3 information: https://www.adafruit.com/product/3055