EUROGRAPHICS Symposium on Sketch-Based Interfaces and Modeling (2012)
L. B. Kara and K. Singh (Editors)

RedDog: A Smart Sketch Interface for Autonomous Aerial Systems

D. Cummmings[1] & S. Fymat[2] & T. Hammond[1]

[1]Department of Computer Science & Engineering
Texas A&M University, College Station, TX, USA
[2]Polarity Labs, 801 Century Park East, Los Angeles, CA USA

Abstract

In order to decrease the number of casualties and limit the number of potentially dangerous situations that Soldiers encounter, the US military is exploring the use of autonomous Unmanned Aircraft Systems (UAS) to fulfill air support requests (ASR) from the field. The interface for such a system must provide interaction in modes that facilitate the completion of the support request in various scenarios, and it must be usable by operators of all skill levels, without requiring extensive training or considerable expertise. Sketches are a simple and natural way to exchange graphical information and ideas. In this paper we present the development of an interface that that allows the user to plan an ASR using sketch and other inputs while conforming to the userŠs mental model of natural interaction.

Categories and Subject Descriptors (according to ACM CCS): Software [D.2.2]: SOFTWARE ENGINEERING—Design Tools and TechniquesUser Interfaces

1. Introduction

Unmanned Aerial Systems add a new dimension to the modern battlefield. Today's warfighter depends on accurate, up-to-date, and easily accessible intelligence to ensure successful execution. The military wishes to use unmanned helicopters to replace road convoys and manned helicopters in hostile environments [The11]. UASs can provide intelligence quickly and safely and thus have become a vital asset to the military. Today, the military uses UASs at every level, from the small, squad-level Raven to the theater-level Predator. However, current UASs rely on dedicated hardware, counter-intuitive user interfaces, and user proficiency (Figure 1). As such, there is a need for an interface that will allow Soldiers [web] to interact directly with and control unmanned air vehicles.

UASs can provide a reasonable alternative to a human pilot and thus have become a vital asset to the military [GO10]. In combat or disaster situations, the requester of support may be occupied with life-critical activities and therefore needs a method of interaction that is both unobtrusive and easy to learn. Sketch recognition can be used to create user-friendly interfaces that can accommodate natural drawing as a means of creating mission plans.

A hand-drawn military Course-of-Action (COA) diagram provides a more efficient means of communicating mission planning information than verbal exchange (going back to the old adage "a picture is worth 1000 words"). The integration of map objects that can contain both GPS and geographical information greatly enhances the potential areas for application. With input from a pen-based interface, sketch recognition can be used to decipher sketches as a means of translating graphic inputs into actions on map objects. By recognizing course of action symbols on an interactive map, we can create various interfaces that will allow users to describe a complicated scenario with a simple drawn image. This scenario can then be translated into actions for other humans controlling a UAS, or even for a completely autonomous UAS. Working in conjunction with the Office of Naval Research (ONR), we have developed RedDog, a smart sketch interface for a UAS. RedDog is an Android-based prototype application that uses a various inputs (including sketch, touch and text) to assist the user in creating a detailed air support request for an autonomous UAS.

DOI: 10.2312/SBM/SBM12/021-028

Figure 1: *Old UAS interface*

2. Related Work

In the areas of UAS interface design, various UAS control and communication methods have been explored. Some of these methods include voice recognition, joystick control and even screen icon interaction to name a few [QGB04]. A sketch-based UAS interface is a logical extension in this realm. The use of quick sketches to show or clarify concepts is a natural and effective form of communication. Sketch recognition is the interpretation of hand-drawn sketches by a computer. It presents a unique problem because no two free-hand drawn sketches are alike. As a result, identifying important shapes and/or text and determining their meaning can be quite a difficult AI problem.

In applications involving geographic information systems (GIS), even minute changes and inputs are often complex and tedious as a result of the non-spatial input methods used by most programs [BSE00]. Various applications have been created in an attempt to capture sketches and translate them into meaningful data for programs.

For example, sketch-based programs such as nuSketch Battlespace and COASketch have been developed to simplify the creation of military course of action diagrams [FUC03], [HLP*10]. nuSketch Battlespace uses glyphs and spacial reasoning while COASketch uses a free-hand sketch recognition approach. In some domains, sketch interaction has been paired with the use of small unmanned air vehicles in order to annotate aerial video feeds taken during disaster situations [ZPM*09].

RedDog combines the sketch recognition capabilities of COASketch (which recognizes over 5,900 hand-drawn symbols with 87% accuracy) and the mapping capabilities available through the Google Map API and the OpenMap package, a toolkit for viewing and manipulating geospatial data [HLP*10], [BBN09]. As such, RedDog can recognize course of action symbols sketched on an interactive map and use the symbol locations to create an action narrative that contains both situational and navigation information that can be used for mission planning.

3. Existing Practices

In the first phase of this research project we sought to understand current practices for interacting with manned helicopters when requesting cargo re-supply. We wanted to identify processes that are applicable to the unmanned scenario but also harness existing knowledge in order to build for Soldiers that requires little additional training. To gather this information, we interviewed 4 Navy personnel and 1 Army officer, who all had involvement in cargo supply and/or medical evacuation procedures either through field operations and/or piloting experience. Our goal was to obtain feedback in order to develop an interaction approach to meet their needs.

Today, the initial supply request is typically made by Soldiers using radio communication, or could also conceivably be made by data communication over the Global Information Grid. In both cases, the request is interpreted by a human. Soldiers load the supplies onto convoy trucks or helicopters, which must then navigate to their destination for dropoff or pickup and return to base, while avoiding threats and obstacles along the way.

We assume that humans will still receive the request for the requested supplies, pick and pack them, and manually load the supplies onto the Cargo UAS, but will not perform traditional piloting activities. We asked the interviewees to assist us in creating an extensive list of user scenarios which we consolidated into 13 examples, two of which are listed below:

1. *A Soldier needs a resupply of bullets and food. Her team is in hiding, waiting for an assault planned for a day from now. The team is located outside, in the mountains, and are without tents. She makes the request at night, so as not to give away her position with bright lights or loud noises. She's probably wearing gloves that are big and clumsy. She's tired and may not be thinking as precisely as normal.*

2. *A vehicle convoy is traveling on a humanitarian mission to a city. At some point the vehicle can no longer progress through city streets. The Soldiers park the vehicle, leave a small security detail to guard the caravan and proceed on foot. Upon arrival at their destination, a Soldier requests Humanitarian assistance packages (10 bags wheat, 10 bags of rice, 10 bags of oil, etc.) from the UAS.*

Using this information, we then identified the requirements that derive from each use case scenario and developed functional specifications for the software prototype in response to these requirements. These requirements include rapidly providing a useful visualization of relevant information of various amounts to the operator. This information must be usable in both benign and harsh environmental conditions, and the system must be usable on a wide range of hardware, from handheld mobile devices, to laptops and workstations. In the next section we will discuss the design decisions that resulted from these requirements.

Based on our interviews, we determined that the system should demonstrate situational awareness. For example, Soldiers can indicate where hostile forces are located. However, the system should not require tedious data entry and should automatically determine how much distance to keep from their location. The UAS should deduce a route satisfying those constraints based upon the flight path, terrain topology, line of sight, current visibility conditions, an on-board model of its own noise signature, etc.

The reasoning behind our initial design approach is that a simple sketch of a support scenario drawn in a few seconds on a map can save minutes of verbal instructions.

4. Implementation

We implemented a sketch-based interface with an interactive mapping feature and modified it though multiple iterative development phases with feedback from domain experts at each level.

4.1. GeoSketch

Geospatial analysis involves the evaluation and processing of geographic datasets from many applicable areas including flight planning, military mission planning, land development, etc. Geospatial analysis, although largely image dependent, has very few sketch-based systems that obtain meaningful input. As a result, there are limits to the level of unrestricted human-computer interaction between geographical information systems and their users. Sketch recognition builds upon the activity of using sketched markups to identify actions on a map.

The concept for RedDog was originally developed as a desktop application which merged a Course of Action Symbol recognition system and the OpenMap package, a toolkit for viewing and manipulating geospatial data [HLP*10, BBN09]. The application, called GeoSketch, can recognize course of action symbols sketched on an interactive map and use the symbol locations to create an action narrative for mission planning (Figure 2). The project's UI contains several layers for interpreting sketch input and an OpenMap layer for delivering geospatial data. Using the OpenMap libraries, the application is capable of hosting all sorts of data from Digital Terrain Elevation Data (DTED) to ESRI shape files to ortho-rectified imagery. The current implementation hosts DTED 0 data (i.e. height elevation data sampled at 30 arc-second intervals) provided by the National Geo-spatial Intelligence Agency. This data format was chosen because it is the standard elevation data format used in Department of Defense (DoD) operational planning, and thus was an appropriate choice for the UAS interface. While this resolution does not provide an impressive amount of visual detail at smaller scales, DTED 0 elevation data for the entire planet is publicly available. Following this implementation, the next goal was to create a similar application on a mobile computing platform.

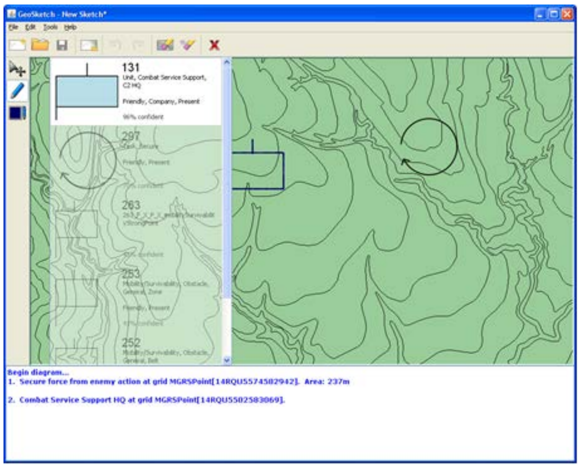

Figure 2: *GeoSketch application with action narrative shown in bottom window*

4.2. Unmanned Air System

In order to create a portable, light weight system, we chose to create the RedDog application on the Android platform with the Galaxy Tab used as the target device. The Android platform has been selected as the platform of choice for the military. The Galaxy Tab was chosen over the iPad due to its smaller size; we wanted to have a device which can be held easily with one hand and sketched on with the other. RedDog was designed to allow the user to quickly create a plan for an ASR with minimal interaction. As a result, a large portion of the interface is created using large button menus. This allows the user to quickly make selections related to each portion of the support request.


4. Type Mission ☐ Tactical
5. Mission Is
A ☐ Assault Transport B ☐ Logistic Support
E ☐ Aerial Delivery F ☐ C2
I ☐ Illumination J ☐ Special Ops
6. Payload Is
A Troops ___
C Internal Cargo Wt/CU ___
7. Instructions
Pickup Time/Date Coordinates
A ___ ___
B ___ ___
C ___ ___
D ___ ___
8. LZ Description
A Wind Direction / Velocity ___
C Size ___
E Friendly Pos ___
F Enemy Pos ___

Figure 3: *Sample ASR Form*

The UAS flow of action is based on a paper ASR form that is currently filled out when a Soldier needs a pickup or delivery mission in the field (Figure 3) [The98]. The form includes various sections that are used to give a human pilot a detailed description about the mission. The goal of RedDog is to facilitate the acquisition of this data through sketch and touch and overlap the sequential inputs wherever possible. In some scenarios, sketch is more appropriate (i.e. when identifying landing zone locations), as opposed to a menu selection action where touch is more efficient.

The interface includes a menu that corresponds to the sections of the ASR form and displays check boxes to keep a record of relevant information that has been retrieved either from the device or from the user. In order to facilitate minimal data entry from the user, information from the device, such as the userŠs location and the current time, are automatically saved and used to help provide information about the support request and/or describe the drop zone. Figure 4 and Figure 5 show how entered information is displayed within the menu in order to maximize space in the interface.

The system provides an interactive map via Android's Google map view. The view allows the user to adjust the focus to her location of choice using one or two fingers to pan and zoom the map, starting from the device's physical location.

The requester begins by selecting a support request from the menu (Figure 4). If the requester selects cargo, she can enter a list of supplies and then proceed to entering information about the cargo drop location. The minimum information required for a cargo drop is what, when and where. Additional information can be provided if available.

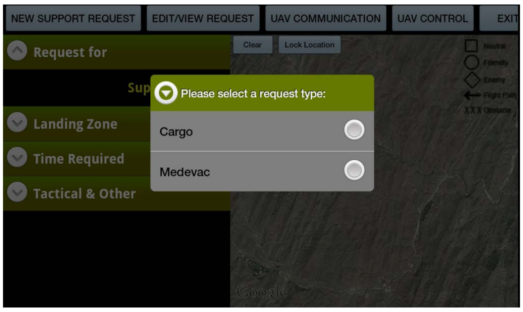

Figure 4: *New Support Request*

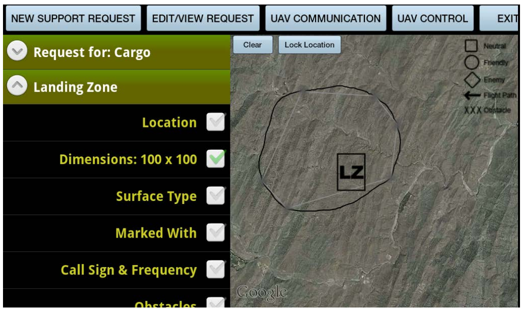

Figure 5: *Landing Zone Area Identified*

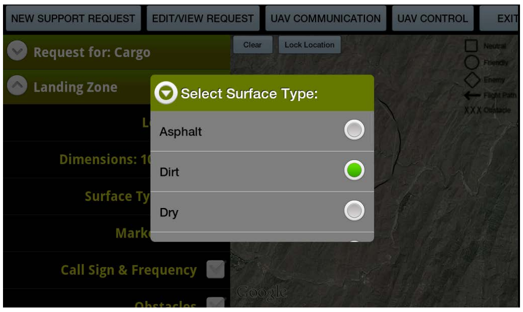

Figure 6: *Sketched Forces and Flight Path Recognized*

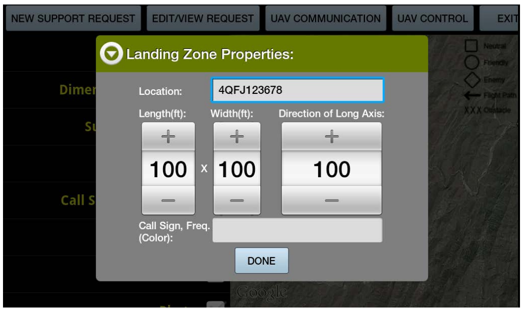

Figure 7: *Landing Zone Properties*

In the ASR form, the description of the landing zone is the most detailed element of the form and therefore requires the most amount of information and time to complete. We have designed RedDog to expedite this process. The device's GPS is used to get the location of the requester and orient the map. It is much easier to sketch the boundary of a landing zone than to verbally describe it. Using RedDog, the requester can outline the landing zone location relative to her position directly on map (Figure 5). Using a stylus or their finger, the requester can draw the boundary of the general area for the landing zone on the map. This gives the system an overall region of land to validate as a potential landing zone based on map terrain information. This is an improvement over the ASR form which only allows the Soldier to identify the center and the size of the landing zone. The map terrain information can also be useful in determining alternate landing zones based on variables such as enemy fire.

After the requester specifies the possible landing zone area through a boundary sketch, RedDog then then gives her an icon representing the ideal landing zone that she can touch and re-orient within the drawn boundary. The coordinates of the landing zone icon are also saved as part of the ASR data. The landing zone icon itself functions as a separate interactive view; when clicked it provides a button menu interface that allows the requester to select properties of the landing zone area such as the the length and width, the surface type, weather, and markers that will be used to identify the area (Figure 8). This information may be used by the UAV to facilitate tracking, targeting and landing functions.

Originally RedDog was developed to recognize the basic military course of action symbols outlined in the MIL-STD-2525B document (Military Symbols for Land-Based Systems). With this feature, users can identify obstacles and/or possible threats to the UAS's flight plan. However, based on our interviews and test scenarios, we determined that many military personnel have not memorized all COA symbols. Therefore for simplicity and ease of use, we've reduced the targeted symbols to primitive shapes that the user can easily remember. She can identify the location of obstacles such as buildings, trees, and fences by sketching Xs directly on the map and assigning properties (Figure 6). In addition she can identify the location of friendly, civilian and hostile forces by drawing circles, squares and diamonds, respectively, on the map (Figure 9). Once drawn and recognized, the images change to the corresponding COA symbol color: red for enemy, green for neutral, and cyan for friendly. This optional situational awareness information can be used by the UAS to complete the mission and avoid any potential threats at the same time.

The primitive sketch recognition was implemented using PaleoSketch [PH08]. RedDog contains a sketch panel object that listens for a stroke action and records each individual stroke as input. PaleoSketch finds and removes consecutive duplicate points from the stroke, and then a set of

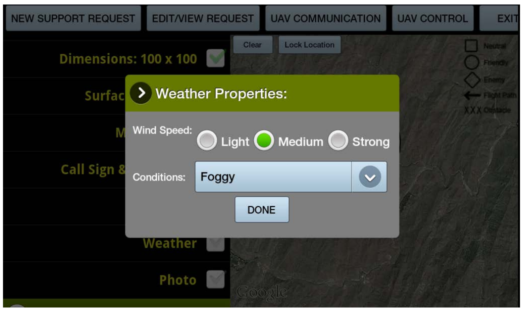

Figure 8: *Landing Zone Properties*

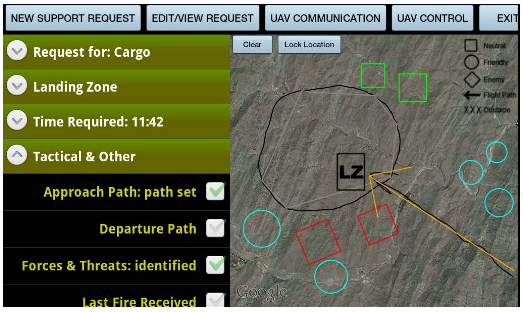

Figure 9: *Sketched Forces and Flight Path Recognized*

features, such as speed, direction and corners, are calculated [WFH11]. These features allow PaleoSketch to represent the original stroke as a set of polylines prior to recognition. Once this beautified shape is returned, Paleo proceeds to run a series of tests to identify low-level shapes. These shapes are then passed to high-level recognizer that can recognize the symbols used in RedDog.

To determine if a stroke represents a landing zone area, or any closed figure, we compute the distance between the stroke's endpoints and divide it by the stroke length. This ratio must be less than some threshold (the level of constraint is depends on the usage) in order to identify the stroke as a closed shape [VFSH11].

To determine if a primitive shape forms a rectangle, we take a group of strokes that create some polygon and assign each stroke to one of the four sides of the shape's bounding box. We then compute the feature area error metric as well as the ratio between the stroke length, the bounding box perimeter, and the ratio of the distance between the 2 furthest points on the stroke and the diagonal of the bounding box. The recognition of diamonds requires the same approach, we simply rotate the stroke 45 degrees prior to calculating the features.

To recognize the single-stroke arrow symbol, we first perform corner-finding on the stroke which should produce 4 segments: one for the arrow body, two for the head and one retrace stroke to change direction on the head. We used the

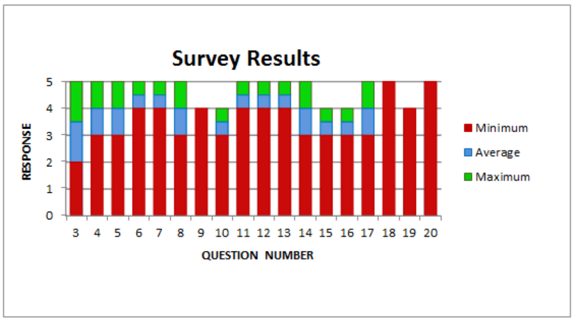

Figure 10: *User responses from questionnaire*

number of segments produced as one of the features, along with the ratio of the size of the arrowhead segments, the distance between the stroke point on the arrowhead and the stroke point on the arrow body.

For a circle we first calculate the average distance between each stroke point and the ideal center of the shape; this feature gives us the ideal radius. We then calculate the normalized distance between direction extremes (NDDE), this is the length of the stroke between direction extremes divided by the entire stroke length. We then verify that the shape is a closed shape, using the method mentioned earlier, and that the stroke's NDDE value is high.

5. Evaluation

We conducted repeated evaluations of RedDog in order to gain feedback and determine its level of usability. The system was presented for feedback to the initial interviewees. In addition the system was evaluated by 2 employees of the U.S. Army that had both flight and field mission experience, and 3 ROTC cadets with mission planning experience. Participants were given a complex field scenario in which they needed to request air support for supplies. They were provided with an aerial image of their location as well as the location of buildings and enemy and neutral forces in the area. Using this information, the participants were asked to walk through two scenarios to complete a cargo request: a radio scenario and a RedDog scenario.

In the Radio Scenario, one participant (behaving as the requester) simulated radio communication to another participant (behaving as the pilot) who filled out the ASR form. This scenario was meant to mimic current standard operations procedures (SOP); any system that we build is meant to compare to the SOP. During this particular exercise, the method of communication was actually easier than a live radio situation since the participants were exchanging information within the same room. Next, participants were asked to perform the same scenario using RedDog. In the RedDog scenario participants were given little to no instruction on how to use the UI in order to identify possible needs for improvement. Upon completion of the exercise, users were given the following questionnaire to evaluate RedDog:

1. Were you able to complete the request using RedDog?
2. How much time did you spend making a request for the RedDog?
3. You have previous experience with computers. (using standard Likert 1-5 for questions 3-20)
4. You have previous experience with military aerial request forms.
5. You have previous experience with touch interfaces (iPads, Android phones, tablets, palmsĚ).
6. The software was easy to understand and use.
7. The software allowed me to input all the relevant information for the request.
8. The software did not contain major bugs or crashes
9. The software is visually appealing.
10. The system is efficient to use and could prove useful in real-life critical missions.
11. Select mission type.
12. Specify supply list.
13. Specify landing zone coordinates.
14. Specify landing zone size
15. Specify enemy's location
16. Specify obstacle locations
17. Specify friend's location
18. Specify marked/unmarked landing zone
19. Specify pickup zone call sign
20. Specify frequency (color code)

Every participant was able to complete the request using RedDog, therefore all answers to question 1 were "yes". For question 2, the average time to complete and verify the air support request using RedDog was 8 minutes. During the radio scenario, we noticed that several pieces of information had to be repeated, even though the participants were in the same room. For this reason, we expect a live radio scenario to require the same or more repetition. Even so, the amount of time to perform the two tasks were comparable even though the participants were not trained to use the RedDog interface. Additionally, given that the ASR data will need to be entered into a digital form in a live scenario, the expected time savings will be greater. Overall every participants completed the task within 3 min of the radio scenario and had took additional time to verify the request. We believe that continued use of RedDog will increase the user's level of confidence and therefore decrease the verification time.

Questions 3-10 asked the participants to assess their level of experience with technology and comfort with the software, with a level 5 being the most comfortable in both areas. Questions 11-20 asked the participants to assess the simplicity of completing the various scenario tasks using RedDog with a level 5 indicating "very simple". The results of their responses are shown in Figure 10. After completing the questionnaire, participants were lead in a group discussion to obtain additional feedback, specifically discussing the advantages of the radio communication vs. RedDog.

Every user that evaluated the system was familiar with the contents of the ASR form. As a result, many of them completed the sections in order of importance rather than in order of the menu options. Because we designed the flow of the interface to allow for non-linear data entry (i.e. the information can be input in any order) they were very satisfied with the menu interaction. Users stated that they were also very comfortable using the touch screens since most of them either owned or had used a smartphone with a touch screen prior to the study. One of the users, a former Apache pilot, had previous experience using touch tablets and was able to navigate through the various section of RedDog with ease.

Users also stated that once they were given the scenario, it did not take them long to understand where and how the data needed to be entered into the system; pop-up messages that gave information on input modes available for each step helped with this task. Upon completing the task, each user stated that the RedDog interface was very intuitive and easy to use.

6. Future Work

Our next step will be to perform a formal field study. We hope to simulate a support request scenario in the field first with human-to-human communication via radio and then using RedDog. We believe that this will allow us to create a more accurate scenario and obtain valuable information regarding interaction challenges.

Since the ultimate goal of RedDog will be to interact with a UAV, the next phase of development will include a flight demonstration with the AR Drone [Par12]. The AR Drone is a recreational quadrocopter that can be controlled via Wi-Fi from an Android device. The purpose of this phase is to demonstrate the functionality of the interface in controlling actual flight operation.

In addition, we plan to explore additional modes of input such as speech recognition as Soldiers are used to interacting with convoys and helicopter pilots via (voice) dialogue. We also plan to explore different modes of input response such as vibration feedback as Soldiers may not be able to devote full attention to the UAS because they are busy doing other things. These factors point to a dialogue framework where the UAS and the human "converse" with each other. This places the UAV in more of a peer-to-peer teaming relationship rather than in a master-slave relationship. Given the projected high level of autonomy for the UAV, it is reasonable to assume that it will be able to initiate interaction with the Soldiers in order to accomplish its mission (i.e. requesting clarification of previous communications, requesting additional information if not enough was provided to compute a flight route, requesting updated information as it is en route, etc). It will also be useful to be able to analyze conflicting instructions, or instructions that might put the UAV at risk and either seek clarification from the Soldiers and/or ignore certain Soldier instructions if they appear to be erroneous.

7. Conclusion

In the near future UASs may be used in place of road convoys and manned helicopters to perform various field support missions. RedDog aims to simplify the medium through which Soldiers can request air support from UASs, thereby eliminating the need for specialized hardware, special training, and flight experience. RedDog uses sketch recognition and geolocation data to allow users of any proficiency to quickly and intuitively devise a UAS operation on a mobile platform. These applications can provide a dynamic, intuitive, and portable solution to the military's UAS interface needs.

8. Acknowledgments

The authors would like to thank the Navy and Army ROTC personnel who contributed their subject matter expertise to this project, as well as the members of the Sketch Recognition Lab, including Francisco Vides, George Lucchese, Jimmy Ho, Brandon Paulson, and Stephanie Valentine. This research is funded in part by a grant from the National Science Foundation and from the Office of Naval Research.

References

[BBN09] BBN TECHNOLOGIES: Openmap, Mar 2009. http://openmap.bbn.com. 2, 3

[BSE00] BLASER A., SESTER M., EGENHOFER M.: Visualization in an early stage of the problem-solving process in gis. *Computors & Geosciences 26*, 1 (2000), pp. 57–66. 2

[FUC03] FORBUS K., USHER J., CHAPMAN V.: Sketching for military courses of action diagrams. In *Proceedings of the 8th international conference on Intelligent user interfaces* (2003), IUI '03. 2

[GO10] GAL-OR B.: The new era of stealth, tailless, vectored aircraft. a 2010 review on current & future designs and applications of manned and unmanned, super-agile and safest military v. civil jets. *International Journal of Turbo and Jet Engines 27*, 1 (2010), pp. 1–10. 1

[HLP*10] HAMMOND T., LOGSDON D., PAULSON B., JOHNSTON J., PESCHEL J., WOLIN A., TAELE P.: A sketch recognition system for recognizing free-hand course of action diagrams. In *Proceedings of the 22nd Conference on Innovative Applications of Artificial Intelligence* (2010), IAAI '10. 2, 3

[Par12] PARROT: Parrot ar drone, Jan 2012. http://ardrone.parrot.com/parrot-ar-drone/usa/. 7

[PH08] PAULSON B., HAMMOND T.: Paleosketch: accurate primitive sketch recognition and beautification. In *Proceedings of the 13th international conference on Intelligent user interfaces* (New York, NY, USA, 2008), IUI '08, ACM, pp. 1–10. URL: http://doi.acm.org/10.1145/1378773.1378775, doi:http://doi.acm.org/10.1145/1378773.1378775. 5

[QGB04] QUIGLEY M., GOODRICH M., BEARD R.: Semi-autonomous human-uav interfaces for fixed-wing mini-uavs. In *Proceedings of the Conference on Intelligent Robot Systems* (2004), IROS '04. 2

[The98] THE DEPARTMENT OF THE NAVY: Close air support mcwp 3-23.1 - dept of the navy, pcn 143 000055 00, Jul 1998. http://ofp.umbr.net/Other/milpubs/Close03-23.1).pdf. 4

[The11] THE OFFICE OF NAVAL RESEARCH: Office of naval research: Autonomous aerial cargo utility systems, Oct 2011. http://www.onr.navy.mil/en/Media-Center/Fact-Sheets/Autonomous-Aerial-Cargo-Utility-Systems.aspx. 1

[VFSH11] VALENTINE S., FIELD M., SMITH A., HAMMOND T.: A shape comparison technique for use in sketch-based tutoring systems. In *Proceedings of the 2011 Intelligent User Interfaces Workshop on Sketch Recognition* (Palo Alto, CA, USA, 2011), IUI '11. 5

[web] Soldier - and that's with a capital s. http://www.military.com/NewContent/0,13190,122303_Soldier,00.html. 1

[WFH11] WOLIN A., FIELD M., HAMMOND T.: Combining corners from multiple segmenters. In *Proceedings of the 8th Eurographics Symposium on Sketch-Based Interfaces and Modeling* (2011), SBIM '11. 5

[ZPM*09] ZOMEREN M., PESCHEL J., MANN T., KNEZEK G., DOEBBLER J., DAVIS J., HAMMOND T., OOMES A., MURPHY R.: Human-robot interaction observations from a proto-study using suavs for structural inspection. In *Proceedings of the 4th ACM/IEEE international conference on Human robot interaction* (2009), HRI '09. 2