3-24-2016

Evaluating Storm Sewer Pipe Condition Using Autonomous Drone Technology

Maria T. Meeks

EVALUATING STORM SEWER PIPE CONDITION USING AUTONOMOUS DRONE TECHNOLOGY

THESIS

Ms. Maria T. Meeks, P.E., USAF

AFIT-ENV-MS-16-M-167

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENV-MS-16-M-167

EVALUATING STORM SEWER PIPE CONDITION USING AUTONOMOUS DRONE TECHNOLOGY

THESIS

Presented to the Faculty

Department of Systems and Engineering Management

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Engineering Management

Maria T. Meeks, P.E., BS

GS-13, USAF

March 2016

AFIT-ENV-MS-16-M-167

EVALUATING STORM SEWER PIPE CONDITION USING AUTONOMOUS

DRONE TECHNOLOGY

Maria T. Meeks, P.E., BS

GS-13, USAF

Committee Membership:

Maj V. Valencia, PhD
Chair

D. Jacques, PhD
Member

E. White, PhD
Member

Maj S. Pierce, PhD
Member

AFIT-ENV-MS-16-M-167

Abstract

The United States Air Force (USAF) owns a total of 30.9 million linear feet (LF) of storm sewer pipes valued at approximately $2.3B in its vast portfolio of built infrastructure. Current inventory records reveal that 78% of the inventory (24.1 million LF) is over 50 years old and will soon exceed its estimated service life. Additionally, the USAF depends on contract support while its business processes undervalue in-service evaluations from long-term funding plans. Ultimately, this disconnect negatively impacts infrastructure performance and overall strategic success. Without a sustainable method of providing accurate, repeatable, and verifiable condition data for underground storm sewer pipes, the USAF civil engineering community risks making uninformed decisions in a fiscally constrained environment.

This research presents a proof of concept effort to automate storm sewer evaluations for the USAF using unmanned ground vehicles and computer vision technology for autonomous defect detection. The results conceptually show that a low-cost autonomous system can be developed using commercial off the shelf (COTS) hardware and open-source software to quantify the condition of underground storm sewer pipes with an efficiency of 36%, determined by the maximum F-measure possible at a single intensity threshold setting. Additionally, this research shows that 3D printing can be leveraged to exploit multi-sensor inputs during asset management (AM) data collection. While the results show that the prototype developed for this research is not sufficient for operational use, it does demonstrate that the USAF can leverage COTS systems in future AM strategies to improve asset visibility at a significantly lower cost.

This thesis is dedicated to my colleagues, friends and family that have provided mentorship, counsel, camaraderie and support throughout my career. Most especially:

To Mrs. Mims – for teaching me the fundamental writing skills that have taken me through my career, including my first introduction to the thesis statement, and for being an inspiration for my professional work ethic and determination to overcome any obstacle.

To my parents and brother – for being a constant source of loving support and encouragement, without which I would be a very different person.

To TJ, Alex, Luke and Levi – for your love and patience while I labored on this thesis. You are the very best of me, and without you I would be lost.

Acknowledgments

This research was only possible due to the generosity of others. I would like to sincerely thank Maj Vhance Valencia for his enthusiasm, unwavering support, and expert writing skills. Thank you for helping leverage my passion for asset management into a solid thesis, and for giving me the skills to communicate more effectively to others. Thanks also to Dr. David Jacques, who generously gave up his time, expertise, and lab equipment so that a simple civil engineer could cross over into the UAS world.

My undying gratitude goes to the ANT lab faculty, staff, and researchers – Rick Patton, Jeremy Gray, Tim Machin, and James Wheeler – for their patience and willingness to help a "newb" on a continual basis. I would still be stuck on the very first Python script and component integration had it not been for your willing kindness. Thanks also to Mr. Bill Porcaro and the AVT Technical Services team for the quick-turn emails and phone calls regarding third-party drivers and exposure settings – both were critical last-minute saves that vastly improved my research results.

My sincere thanks go to Daniel Worden and Derek Terrellion from the 88th AW Civil Engineering Squadron for generously giving up their Friday afternoons to find an adequate field test site and to ground truth the data. And many thanks to Brian Allen for helping grind through the gory image processing. Lastly, a huge thanks goes out to Capt Patrick Grandsaert for the inspiration that started me on this research, and also for the reachback technical expertise on the inner workings of the algorithm…not many would sacrifice off-duty hours from a stressful job in Korea to help a desperate researcher!

Maria T. Meeks

Table of Contents

List of Figures

List of Tables

List of Equations

EVALUATING STORM SEWER PIPE CONDITION USING AUTONOMOUS DRONE TECHNOLOGY

I. Introduction

"The real voyage of discovery consists not in seeking new landscapes, but in having new eyes." – Marcel Proust

Chapter Overview

The main purpose of this chapter is to introduce the focus of this research effort and relevant background information. This chapter defines the problem statement and establishes research objectives. In the background section, the chapter introduces the history of asset management within the United States Air Force (USAF) and continues with significance of condition assessments within the asset management framework. The chapter concludes with a summation of the report structure for subsequent chapters.

Background

The USAF operates and maintains a massive infrastructure portfolio valued at $276B located across 183 active, guard, and reserve installations positioned around the world (DUSD(I&E), 2015). The infrastructure portfolio is comprised of complex systems including facilities, pavements, utilities, waste management, and natural infrastructure systems. This research collectively refers to these complex systems and their components as built infrastructure. Each unique system and component of the built infrastructure age at different rates, and consequently deteriorate differently over time.

Thus, recurring sustainment investments are required to ensure satisfactory performance and USAF mission reliability.

Improved asset visibility and performance modeling through asset management enables objective financial decisions for sustainment investment strategies. In recent years, the Federal government instituted policies promulgating asset management to all agencies. In February 2004, President George W. Bush signed Executive Order 13327, Federal Real Property Asset Management, which recognizes the need for a structured real property management framework on a government-wide scale. The executive directive seeks "to promote the efficient and economical use of America's real property assets and to assure management accountability for implementing Federal real property reforms" (*Executive Order 13327*, 2004). The Air Force issued Policy Directive 32-90 in August 2007 to enforce the Executive Order, which empowered the USAF civil engineering (CE) community to adopt asset management principles and processes at all Air Force installations.

Asset management (AM) methodology facilitates targeted, informed, and predictive decision-quality data. This data enables USAF engineers to optimize resources and investments towards aging infrastructure by creating a framework to answer the following questions (Vanier, 2001):

- What infrastructure does the USAF have?
- What is its worth?
- What is its condition?
- What is the remaining service life?

- What do you fix first?

Asset condition inspections, proactively executed on a routine basis and yielding accurate condition data, are critical for fiscally responsible decision making. In-service inspections of performance are a large component of the asset management methodology, and align with Goal #3 of the *USAF Civil Engineering Strategic Plan* (USAF, 2011):

> Asset visibility and performance data will allow Civil Engineers to leverage strategic sourcing for requirements needed across our portfolio… Total asset visibility will be implemented across all functional areas to account for every piece of the Air Force Civil Engineering enterprise.

The USAF Civil Engineering Strategic Plan Goal #3 establishes the significance of in-service asset condition inspections within the asset management framework – it is through these inspections that the USAF has financial accountability of its budget. This research studies a conceptual methodology to generate accurate, repeatable, and verifiable condition data for underground storm sewer pipes in the USAF real property inventory.

Problem Statement

The USAF owns a total of 30.9 million linear feet (LF) of storm sewer pipes valued at approximately $2.3B. Within the Air Force asset management framework, these storm sewer pipes should have a service life of 40 to 70 years depending on the pipe material (AFCEC, 2014b). Current inventory records reveal that 78% of the USAF storm sewer pipe inventory (24.1 million LF) is over 50 years old. Furthermore, 53% (16.4 million LF) of the total storm sewer pipe inventory is at least 60 years of age (Figure 1). These ages are significant as they indicate that the USAF storm sewer pipe inventory will soon exceed their estimated services lives. In addition, the USAF does not know the condition of many of these pipes. Given the large volume of assets, the age of these assets, and their unknown conditions, the Air Force cannot conduct effective infrastructure asset management.

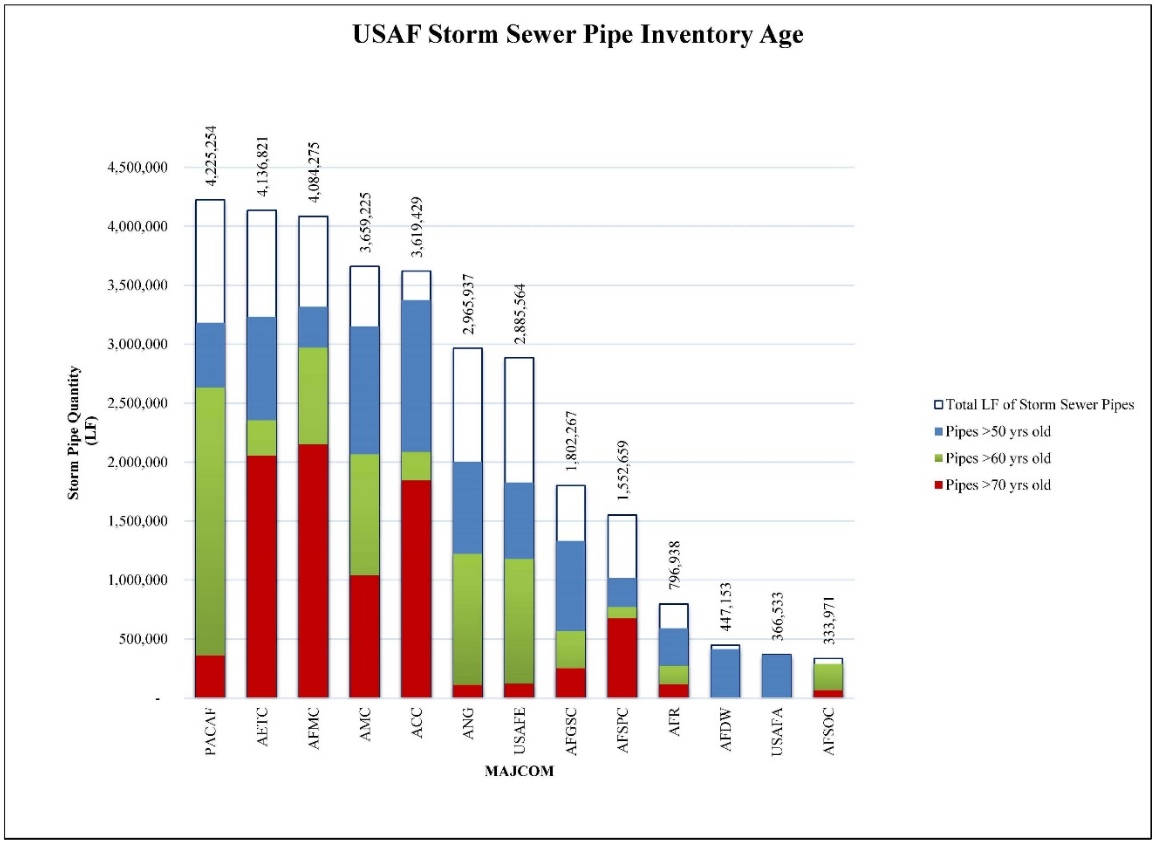


Figure 1. Total USAF Storm Sewer Pipe Inventory Age by MAJCOM (AFCEC, 2014a)

Storm sewer pipe location and condition data is critical to effective management of aging storm sewer assets. This data is difficult to obtain because storm sewer systems have the following characteristics: (1) pipes are often underground with limited access; (2) storm water collection system have an expansive footprint across installations, sometimes exceeding 150 miles per installation (AFCEC, 2014a); and (3) the pipe structure is a confined space in which to perform inspection. Collecting data inside of storm water pipes requires specialized equipment, most commonly closed-circuit television (CCTV) inspection systems. These CCTV systems require specialized skills and experience to operate effectively.

USAF civil engineering (CE) business practices do not support a sustainable method of performing comprehensive in-service evaluations using its own personnel or equipment. Although some CE organizations do have smaller CCTV inspection systems available for small-scale inspections and repairs, the USAF on a whole outsources that capability to private-sector firms. These firms have larger inventories of CCTV equipment and specially trained staff which often produce written reports on the condition of surveyed pipes. However, the reports generated by these private-sector firms are typically stand-alone products that are difficult to incorporate into the USAF software used to manage infrastructure. The USAF currently does not have a sustainable organic capability to provide accurate, repeatable, and verifiable condition data for underground storm sewer pipes.

Research Objectives and Investigative Questions

This research sets out to prove that a low-cost autonomous system can quantify the condition of underground storm sewer pipes as good as or better than a CCTV inspection. The operational goals of this system are to operate inside a storm sewer pipeline with minimal human operator activity and take measurements for the accurate location and current condition of the pipes. To limit the scope of the problem into achievable objectives, this research asks the following investigative questions:

1. How can a small autonomous unmanned ground vehicle (UGV) be configured to collect pipe condition information?
2. What field data and programming code is required to develop a data processing algorithm for pipeline fault detection?
3. How can the quality of pipeline fault detection data be quantified in order to inform decision-makers on pipe condition?

Assumptions and Limitations

This research presents a proof of concept effort to automate storm sewer evaluations for the USAF; therefore, a complete system design is not the focus of this study. The researcher considered only critical requirements and capabilities in the development of the data collection system prototype. Additionally, redundancy was not a priority for the critical capabilities in the system architecture.

According to the Pipeline Assessment and Certification Program (PACP), there are many possible defects within storm sewer pipes (NASSCO, n.d.). Figure 2 displays the four major categorizations of these various defects. Due to the complexity of the problem and breadth of available technologies, this research does not consider all possible defects in pipes and only focuses on crack detection. Further, it does not classify the severity or type of cracks detected. Rather, this work identified whether or not a crack or cracks existed in a section of the surveyed pipes. To validate the algorithm developed, subject matter experts were employed to "ground truth" images. A team of individuals interpreted the images and identified the crack location.

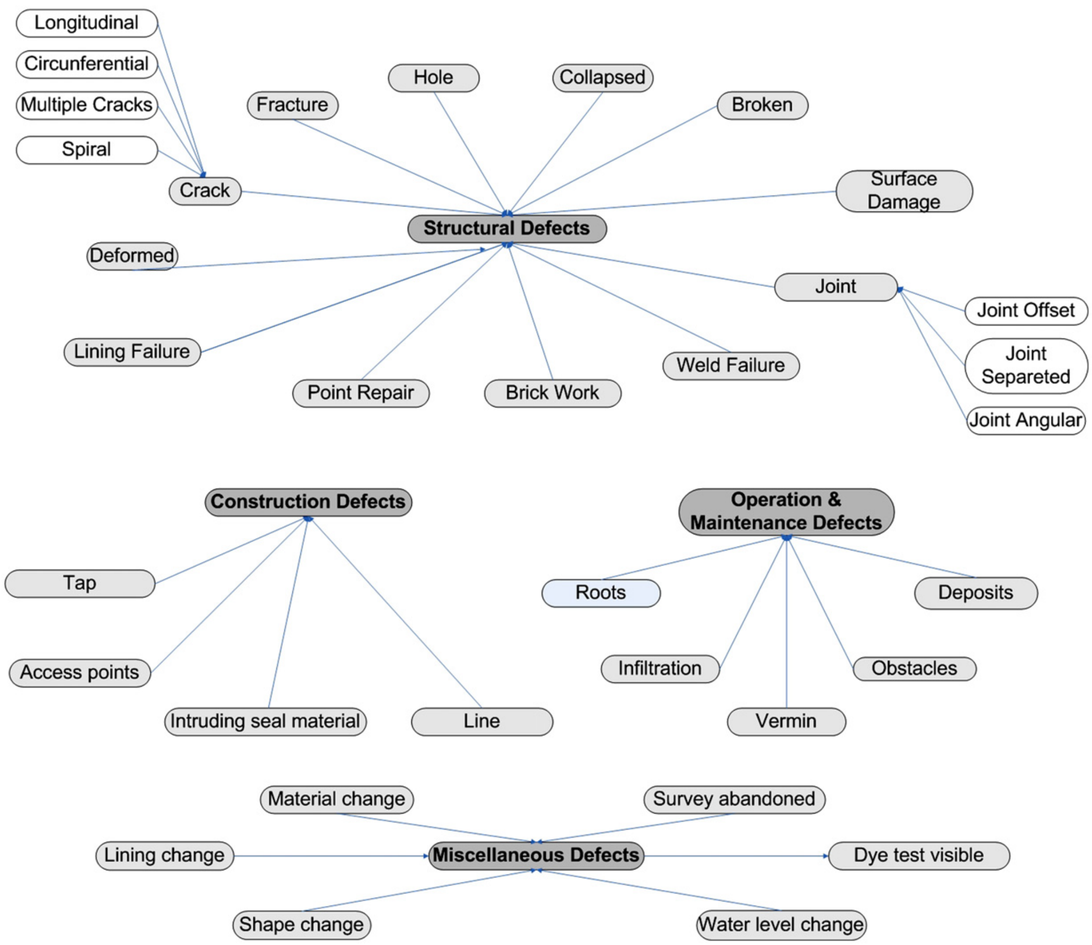


Figure 2. Defect classifications in sewer pipes according to PACP (NASSCO, n.d.)

With respect to location, the Global Positioning System (GPS) used for navigation could not operate within subterranean storm water pipes. While navigation in the absence of GPS is a current research focus area at the Air Force Institute of Technology (AFIT), it was not feasible to incorporate that work into this research due to time constraints. As a result, this research uses an open storm water drainage channel in place of an underground storm water pipeline for field testing.

Finally, this research is limited to the environmental conditions present during the field test performed. Due to time constraints and unpredictable weather events, this research performs only one field test, and the sample size was limited to the images collected at that time. Although the limitation of a single field test impacts sample size significantly, it is deemed sufficient for a proof of concept.

Overview

This thesis report follows the traditional five-chapter format. Chapter I provided the background and context for this thesis's research problem. Next, Chapter II consists of an extensive literature review of sewer evaluation technologies and robots used in civil engineering, and other viable technologies used in asset management for underground pipes. Chapter III presents the methodology employed in this study, which is field testing of the prototype system developed in this research effort. Chapter IV includes the analysis and results from the field tests. It includes the validated results from subject matter expert judgment through a "ground truthing" process. Finally, Chapter V presents a discussion of the results, conclusions that can be made from this research, implications for USAF infrastructure asset management and asset management practice in general, and suggestions for future research.

II. Literature Review

Chapter Overview

This chapter provides a foundation for understanding the central topics of this research based on existing literature. First, the importance of in-service evaluations to effective asset management is detailed. A discussion on in-service evaluations establishes the relevance of this research towards the effective asset management of storm water systems. Next, a summary of sewer evaluation technologies, and a comparison of advantages and disadvantages, is used to explain the selection of available sensors available. Robots used in infrastructure inspections, including both unmanned aerial vehicle (UAV) and unmanned ground vehicle (UGV) platforms, are introduced as models for this research. Sewer evaluations using robots integrated with sensors are described with two specific systems, KANTARO and PIRAT, highlighted. Finally, the concepts of computer vision techniques and mathematical modeling are introduced as a method of automating pipeline fault detection.

Asset Management

Any real property owner requires standardized processes and tools to effectively manage these assets. Using the analogy of a homeowner with a single home, it seems obvious that the owner must first know all systems (e.g. electrical, plumbing, exterior structure, interior structure) and components (e.g. fuse box, furnace, roof, painted drywall) that make up the house. The owner must then use systematic processes to inspect each system and component routinely in order to detect significant deterioration and anticipate failure. The homeowner must understand, on a holistic or macro-scale, what they have, what condition it is in, and remaining service life in order to make decisions about maintenance and repair investments.

The USAF is much like this homeowner, but owns multiple assets and varying types of built infrastructure (e.g. buildings, roads, airfields, buried utilities) which creates a challenging scenario for the service. Because of the wide variation in its asset portfolio, the USAF is at increased risk of poor investment decisions. In general terms, asset management (AM) is the processes, tools, and culture necessary to manage complex built infrastructure from "cradle to grave". Figure 3 shows a graphical model of AM for an organization (TMI Africa, n.d.), which is a similar process to that which the USAF follows.

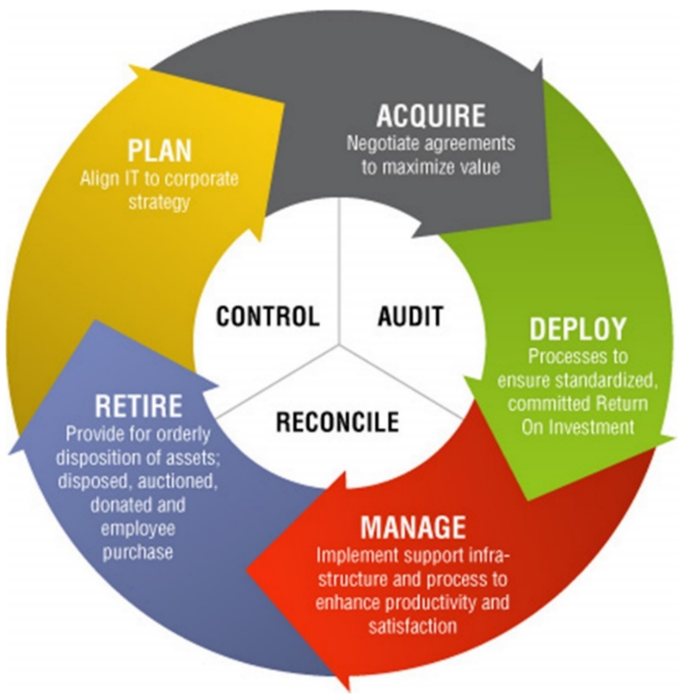


Figure 3. Example asset management framework. (TMI Africa, n.d.)

The USAF defines AM as “use of systematic and integrated processes to manage natural and built assets and their associated performance, risk, and expenditures over their lifecycles to support missions and organizational goals" (USAF, 2012: 196). The AM processes and systems play a critical role in controlling the lifecycle expenditures of an asset. By ensuring timely maintenance and rehabilitation requirements and ensuring asset performance meets strategic goals and needs, the USAF has a method to optimize its investments using limited funding. In short, it is identifying investments for the right expense at the right time (USAF, 2011: 2).

As part of a long-term investment strategy, in-service assessment is an essential part of effective infrastructure AM and controlling lifecycle expenditures. In-service evaluations involve monitoring the use and physical condition of an asset on a recurring basis. In-service evaluations can focus on either functional, structural, or environmental aspects of the system (Uddin, Hudson, & Haas, 2013). Functional evaluations focus on how effectively the asset performs its intended functions. Storm sewer system functional evaluations, for example, would measure the number of breaks and/or leaks per year, average volumetric flow, and quantity and type of repairs on the system. By comparison, structural evaluations assess the structural integrity of the asset by performing material tests. A storm sewer structural evaluation would analyze the type of pipe break in order to determine the remaining life of the pipe material. Finally, environmental evaluations focus on health and safety and require subjective data such as user satisfaction and hazard assessments.

Asset managers analyze the information collected during in-service evaluations to support decisions on performance and investment strategies. Uddin, Hudson, and Haas

(2013) detail that asset managers use data for analysis of failure calculations, establishing maintenance and repair schedules, validation of predicted component and asset performance data, and as a basis for evaluating construction and maintenance techniques. In-service evaluations of storm sewer systems and the data it produces play a pivotal role in establishing valid long-term investment strategies within AM.

Currently, the USAF CE community completes very limited in-service assessments for storm sewer networks, and typically assessments are not completed on a regularly-scheduled basis. The USAF routinely depends on recurring contracts to private-sector companies specializing in this service on an as-needed basis rather than use in-house personnel and equipment. The cost of the contracted surveys depends greatly on the quantity or length of pipes surveyed, complexity of data required, and market availability of specialized contractors. However, they can range from approximately $30K for a limited survey to $350K for a more comprehensive inspection (Isaacs, 2015). As a result of the substantial costs and the process to prioritize and allocate investments across the USAF, comprehensive surveys of an installation are typically completed only every 5-10 years (Isaacs, 2015).

Compounding this issue, storm water system assessments are not highly prioritized investments in the USAF integrated priority list (IPL). Assessments typically only score high enough to warrant funding if there is supporting evidence of catastrophic failure or a well-crafted justification statement illustrating significant cost avoidance through the investment (AFCEC, 2014b). In one instance, an inflow/infiltration (I/I) study valued at $300K to identify the inappropriate connection of surface drainage into sanitary sewer systems at Mountain Home AFB, ID was not funded. This project received a score of 88.4 out of a possible 210 points through the IPL process. This score placed it in the lower third of other ranked, unfunded projects and guaranteed that it would not receive funds for several years (Isaacs, 2015). This I/I study was funded in FY15 only after the base justified that $615K in unnecessary wastewater treatment would be avoided over a period of three years by completing an I/I study and targeting repairs to the storm water system (Isaacs, 2015). Based on current business rules, the USAF IPL consistently undervalues in-service evaluations from the long-term funding plans.

Routinely failing to fund needed in-service evaluations has a direct impact to the USAF's infrastructure AM. AM activities are interdependent and collaborative, therefore omitting evaluations will negatively impact decision making (El-Akruti, Dwight, & Zhang, 2013). Specifically, unreliable or sporadic condition assessments impact the accuracy of performance analysis and evaluation, which in turn impacts decisions regarding the maintenance and rehabilitation of assets. The lack of reliable and timely assessments negatively impact infrastructure performance and overall strategic success (El-Akruti et al., 2013). The USAF needs organic capabilities to support reliable and timely in-service assessments of its storm sewer infrastructure.

Sewer Pipe Condition Assessment Technologies

Various technologies exist for performing storm sewer condition assessments. In general, the alternative used in sewer evaluation is based on relevant characteristics of the pipe including sewer pipe geometry, the type of pipe material, and the nature of failures of the pipe network (Duran, Althoefer, & Seneviratne, 2002). Additionally, whether the evaluation is environmental, functional, or structural will have a bearing on the technology selected (Uddin et al., 2013). For example, a structural condition assessment may require a comprehensive internal inspection that extends to the soil beyond the inner surface of the pipe wall to detect corrosion through the pipe wall, soil settling, and root invasion damage to the pipe by vegetation. A functional condition assessment, by contrast, would require internal investigation of the state of pipe compared to normal operating parameters (e.g. leakage and capacity ratings).

Sewer pipe monitoring and evaluation alternatives are listed below and discussed in more detail in the next several subsections of this report. A comparison of advantages, disadvantages, and detection limitations for each technology are summarized below, and further displayed in Table 1 on page 20 (Costello, Chapman, Rogers, & Metje, 2007; Duran et al., 2002; Koo & Ariaratnam, 2006). Listed below are the main categories of these inspection techniques, the remainder of this section is organized according to these five categories:

1. Optical inspection (CCTV most typical)
2. Sewer Scanning and Evaluation Technology (SSET)
3. Acoustic and ultrasonic testing
4. Infrared (IR) thermography
5. Ground penetrating radar (GPR)

Closed Circuit Television (CCTV)

Closed-circuit television (CCTV) surveys are currently the most common method for assessing the condition of storm sewer networks (Duran et al., 2002). CCTV inspection systems basically consist of a camera and lighting source mounted on a remote-controlled vehicle similar to that shown in Figure 4. The CCTV camera records massive amounts of digital images of the pipe interior, and transmits the footage to a display within the support vehicle.

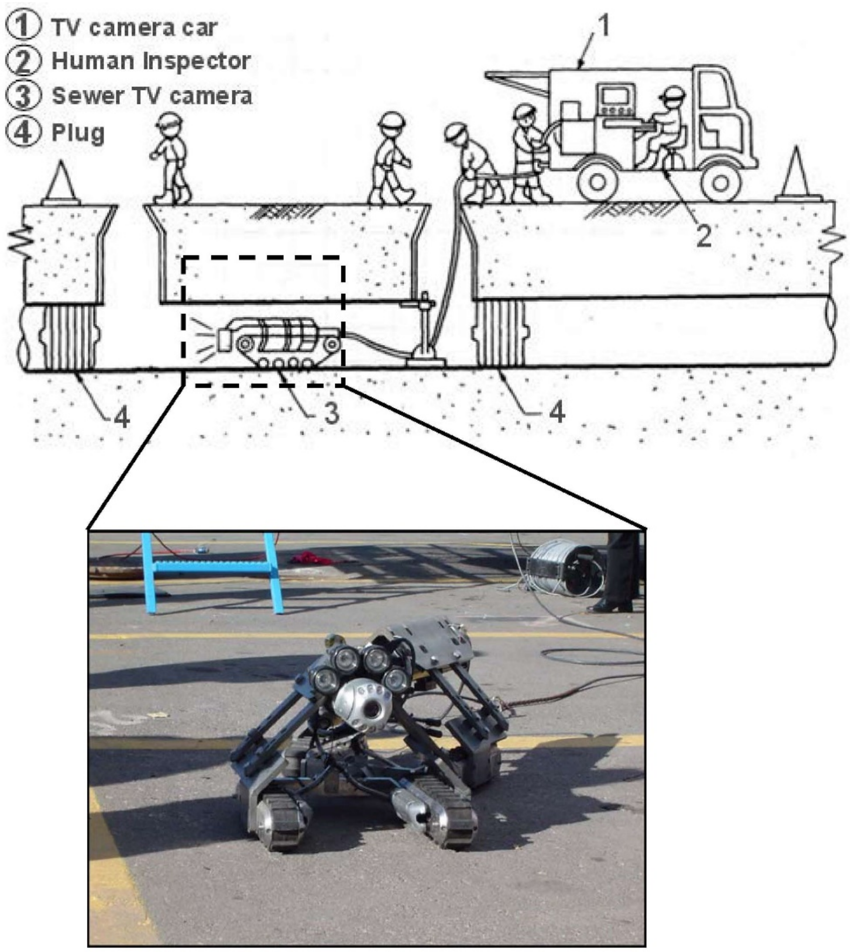


Figure 4. CCTV pipeline inspection system. Created using images from (Koo & Ariaratnam, 2006; Nassiraei, Kawamura, Ahrary, Mikuriya, & Ishii, 2007)

Condition assessment using CCTV systems is a time-intensive procedure. The placement, operation, and recovery of CCTV inspection systems from the pipe demand a significant amount of time in the field simply for data collection. Operators perform a minor amount of defect investigation and classification in the field; however, technicians typically review the footage after-the-fact and classify the defects off-site. Therefore, CCTV systems require a two-fold investment of human labor: operation of the systems in the field (i.e. data collection), and post-processing for condition rating (i.e. footage evaluation by technicians) (Romanova, Horoshenkov, Tait, & Ertl, 2012) .

The cost of CCTV assessment systems can be expensive. An example of one commercially available pipe inspection robot is the FiberScope.net® Pipe Crawler STORMER S3000™, where a single system is currently priced at $35,900 (FiberScope.net, 2015). This research did not find a statistical study of typical CCTV inspection rates and costs in available literature. However, one study cited an inspection rate of 300 meters per day at approximate cost of $16 USD/meter (Nassiraei et al., 2007). It is difficult to estimate the average field inspection rate with CCTV systems. The time it takes to inspect a pipe is highly dependent on the number of defects, the degree of deterioration, and the proficiency of the operators (Wirahadikusumah, Abraham, Iseley, & Prasanth, 1998). However, in general researchers and industry experts subjectively estimate that the inspection rate using CCTV ranges from 300 to 1800 meters/day (Cancilla, 2016; Nassiraei et al., 2007) .

In addition to the high cost and slow inspection rate, there are several limitations with respect to the quality of CCTV assessments. Some of major drawbacks are the lack of visibility, potential for obstructions, and the non-uniform shape inside sewer pipes (Duran et al., 2002; Romanova et al., 2012). As a result of these limitations, engineers have pursued multi-sensor technologies to either replace or enhance the CCTV for optimal sewer inspections.

Sewer Scanning and Evaluation Technology (SSET)

Sewer Scanning and Evaluation Technology (SSET) complements conventional CCTV cameras with a 360° digital laser scanner and additional lighting systems. This increases the accuracy and resolution of the recorded images by creating a surface profile of the pipe interior (Costello et al., 2007; Duran et al., 2002; Koo & Ariaratnam, 2006). There are several profiling techniques possible with SSET, however in each case the operator must take precautions. For example, the operator must keep the angle between the camera and the illumination source on the same virtual optical axis in order to avoid complex geometric analyses required to reduce measurement error (Duran et al., 2002).

The main drawbacks to laser-based systems are a lack of calibration with respect to measuring changes in shape, and the need for specialized camera and lighting systems when working in water. Also, they cannot inspect beyond the inner pipe wall like other advanced sensing methods (Costello et al., 2007; Duran et al., 2002).

Acoustic and Ultrasonic Testing

Ultrasonic-based sensors use high-frequency sound waves to detect material thickness, lamination, and planar defects on surfaces that reflect acoustic energy back to a transducer. Ultrasonic is a very versatile and commercially available technology; however some components, such as improved air operational transducers and electromagnetic acoustic transducers, have not been successfully used in sewer assessment (Duran et al., 2002).

A major drawback of these systems is the inability of ultrasonic sensors to measure flooded and dry areas simultaneously, due to the optimal operating frequencies in those two mediums (Costello et al., 2007). As well, technicians using sonar technology require a high level of experience to successfully interpret results (Koo & Ariaratnam, 2006). Overall, the presence of water, non-uniform pipe materials, and rough surfaces inherent with a sewer system create difficulties with using ultrasonic technology for sewer inspections.

Infrared (IR) Thermography

Infrared (IR) thermography measures temperature differences across an object resulting from IR radiation distributing heat in a closed environment (Duran et al., 2002). The measured heat distribution is then converted into a visible image, where areas of differing temperatures are distinguished by different colors.

Duran et al. (2002) detail two different processes applicable in IR thermography: active (i.e. where an artificial heating source is required) and passive (i.e. no heating source is used). According to Wirahadikusumah, Abraham, Iseley, & Prasanth (1998), subsurface defects in sewer pipes previously not visible in conventional CCTV surveys were successfully identified using passive IR thermography. These defects included deteriorated pipeline insulation, leaks, and voids. Overall, the authors conclude that IR thermography is intrinsically safe, allows for measurement of large areas in shorter assessment periods, and is a viable method for performing sewer inspections.

Ground Penetrating Radar (GPR)

Ground Penetrating Radar (GPR) emits short pulses of electromagnetic energy to provide information about the pipe and the surrounding soils. It is possible to use GPR either inside or outside the pipe. According to Duran et al. (2002), sewer networks in France were successfully inspected using CCTV systems augmented with the GPR. The researcher was not able to confirm this work using the original source, however.

The main advantages of GPR are that: 1) the antenna does not have to be in contact with the surface of the pipe, 2) it can penetrate depths beyond the pipe wall to collect surrounding soil information, and 3) the inspection speed is much faster when compared to other methods (Duran et al., 2002).

Table 1 summarizes a general description of each technology and compares the advantages, disadvantages, and detection limitations for each. The information consolidated in Table 1 came from the research of Costello et al.(2007), Duran et al. (2002), and Koo & Ariaratnam (2006). The detection limitations listed are restrictions specifically called out in research source that influence the use of this technology in sewer condition assessments.

Table 1. Inspection and Data Collection Technique Comparison (Costello et al., 2007; Duran et al., 2002; Koo & Ariaratnam, 2006)

Method	Description	Advantages	Disadvantages	Limitation
Closed Circuit Television (CCTV)	A skilled technician controls a vehicle/platform fitted with a color, high-resolution video camera and lighting system. Camera acquires images of the inner surface of the pipe, and operator examines footage to classify and rate severity of pipe defects.	• Most conventional method of assessment, increases equipment availability and decreases costs	• Quality of assessment highly dependent on quality of acquired images and operator training • Size of data generated per assessment is exorbitant and major hindrance for conducting large-scale surveys (e.g. 30 hrs. of video per 10 km of pipe assessed) • Assessment quality impacted by changes in pipe shape, obstructions, and lack of visibility	• Subsurface inspection past inner wall of pipe is not possible • Visibility of operator limited by system lighting
Sewer Scanning & Evaluation Technology (SSET)	Conventional CCTV camera equipment, plus: - structured light sources (e.g. laser diodes) - fiber optic gyroscope - fish eye digital scanner Additional equipment increases the accuracy and resolution of the recorded images by creating a surface profile of the pipe interior and provides added coverage/mobility of camera.	• Better resolution than conventional CCTV images • Continuous 360° image • Increased accuracy of wall defect detection and improves assessment productivity	• Configuration of detector and objective lens is limited with respect to optical axis	• Subsurface inspection past inner wall of pipe is not possible • Limited to data collection above water line

Acoustics and Ultrasonic testing (Sonar)	High frequency sound waves used to measure geometrical changes in sewer inner wall (e.g. material thickness, lamination, and planar defects on surfaces).	• Very accurate results • Can detect corrosion pits, voids, and perpendicular cracks on pipes inner wall • Versatile technology, different commercial probes/ measurement modes available	• Requires high level of experience and training to interpret results • Large amounts of data are usually generated • Non-uniform pipe materials affect measurements • Rough surfaces of pipes can create coupling problems • Difficult to create guided waves and mode conversion (e.g. longitudinal waves transformed to transverse)	• Subsurface inspection past inner wall of pipe is not possible • Cannot measure flooded and dry areas of pipe simultaneously
Infrared (IR) Thermography	IR radiation (heat) is used to generate temperature differences across an object in a closed environment, then measured. The measured heat distribution is then converted into a visible image, where areas of differing temperatures are distinguished by different colors.	• Subsurface inspection beyond the pipe wall (e.g. soil condition) is possible • Inspection speed is high relative to other methods • Not affected by type of material to be tested	• High sensitivity to illumination	• More than one subsurface defect at same position cannot be detected
Ground Penetrating Radar (GPR)	Equipment emits short pulses of electromagnetic energy to provide information about the pipe and the surrounding soils.	• GPR can be used either inside or outside the pipe • Subsurface inspection beyond the pipe wall (e.g. soil condition) is possible	• Requires high level of experience and training to interpret results	unknown

Robots Used in Civil Engineering

Underground infrastructure inspections in civil engineering are typically performed by robots that can more easily travel inside of confined spaces and over longer distances without fatigue. The previous section discussed different sensing technologies; robots are the systems that integrate these sensors with a transport platform, data storage, and sometimes processing. Robots can be semi-autonomous or autonomous depending on the degree to which they leverage autonomous navigation or computer algorithms to interpret the sensor data (Nassiraei et al., 2007; Wirahadikusumah et al., 1998). A semi-autonomous robot is typically tethered to a support vehicle, allowing a human operator to partially control the robot during navigation using remote-control equipment, or have algorithms that interpret only part of the data collected, requiring a human operator to interpret the remaining data based on subjective judgment (Nassiraei et al., 2007). Fully autonomous robots, by comparison, are self-contained and do not require human operator inputs for either navigation or data interpretation.

Robots can be fitted with one or more of the advanced sensor technologies discussed in the previous section. Koo and Ariaratnam (2006) and Guo, Soibelman, & Garrett Jr. (2009) both provide evidence that combining collaborative (i.e. multiple) sensor technologies increases accuracy and yields better evaluation results. Koo and Ariaratnam (2006) performed field and experimental testing of sanitary sewer pipe using a prototype GPR and SSET combined tractor systems and found that the multi-sensor approach "overcomes the limitations of each technology" (2006: 487). Additionally, Guo et al. (2009) performed a case study to explain how an autonomous multi-sensor robot platform coupled with an autonomous defect detection algorithm would make conventional sewer pipeline condition assessments significantly better. The researchers collaborated with RedZone Robotics® to test their proposed change detection approach using a dataset of 103 CCTV images, taken by RedZone Robotics® using multi-sensor robot systems. There are two sewer evaluation robots used as models for this research: the PIRAT semi-autonomous system (Kirkham, Kearney, & Rogers, 2000) and KANTARO fully autonomous pipe inspection robot (Nassiraei et al., 2007).

Pipe Inspection Real-time Assessment Technique (PIRAT)

The Pipe Inspection Real-time Assessment Technique (PIRAT) (Kirkham et al., 2000) was a semi-autonomous (i.e. remote-controlled) in-pipe vehicle developed in 1996 in Australia. The PIRAT is a customized CCTV system augmented with laser scanning and sonar sensors (Kirkham et al., 2000). The researchers custom-built a vehicle (Figure 5) using commercial CCTV systems as a model, but improved it for additional payload, smooth continuous motion via tracks in lieu of wheels, and operation in flooded sewers. The PIRAT in-pipe vehicle was designed to fit 24-inch diameter pipes, and be semi-manually operated and is tethered by an 820-ft umbilical cable to the support vehicle. The umbilical cable serves a dual purpose of both transferring information from the PIRAT robot to the support vehicle and a means of retrieving the robot should it malfunction.

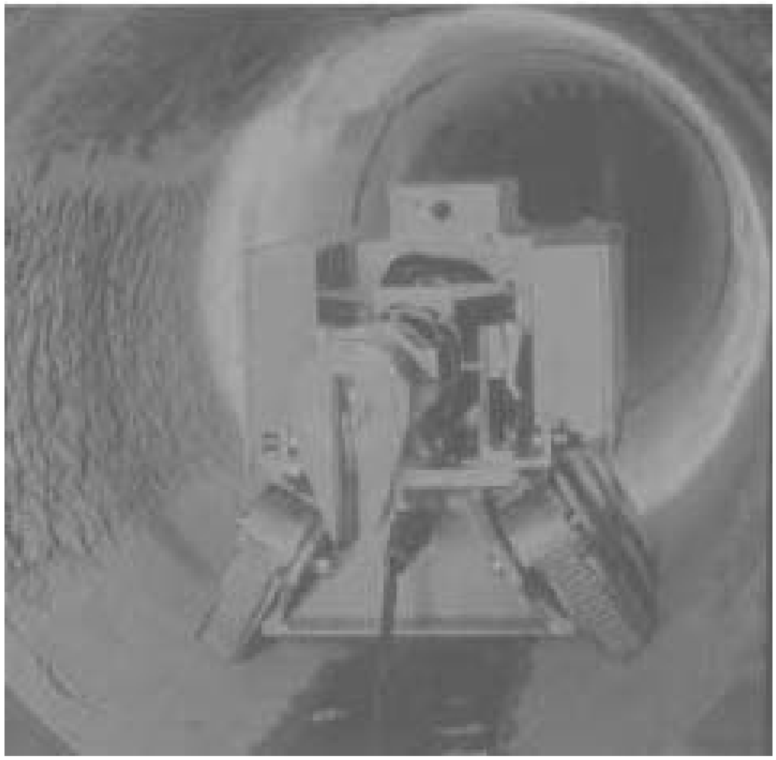


Figure 5. PIRAT in-pipe vehicle, support. (Kirkham et al., 2000)

The concept of PIRAT involves using the semi-autonomous vehicle to create a geometric model of the sewer pipe using measurements taken by laser and sonar scanners. PIRAT uses machine learning as a method of recognizing defects in sewer pipes based on data-driven predictions. It also uses neural network techniques, which use input factors or measurements to classify an output, to classify and rate various pipe defects (Kirkham et al., 2000: 1042). Kirkham et al.'s research focus was the collection and analysis of data rather than vehicle design and navigation. Therefore the PIRAT in-pipe vehicle is not overly maneuverable and relatively large compared to other sewer inspection robots.

The PIRAT prototype performed laboratory experiments in wet and dry concrete and vitrified clay pipes. The researchers performed subsequent field tests in wet and dry vitrified clay, concrete, and brick sewer pipes. Overall, the PIRAT results were superior to conventional CCTV in large diameter concrete and clay sewer pipes (Kirkham et al., 2000: 1052).

KANTARO Robot

The KANTARO sewer robot is a custom-built system developed in Japan in 2007. The KANTARO prototype (Figure 6) proved to be a superior design to the PIRAT, miniaturized to navigate pipes within a diameter range of 8 to 12 inches. KANTARO includes a patented moving mechanism that integrates artificial intelligence and highly sophisticated navigation techniques. The mechanism, which the researchers call “naSIR mechanism”, has the capability to maneuver through a wide variety of pipe bends and joints, traverse obstacles, and travel different size pipes without navigation controller intelligence or sensor inputs.

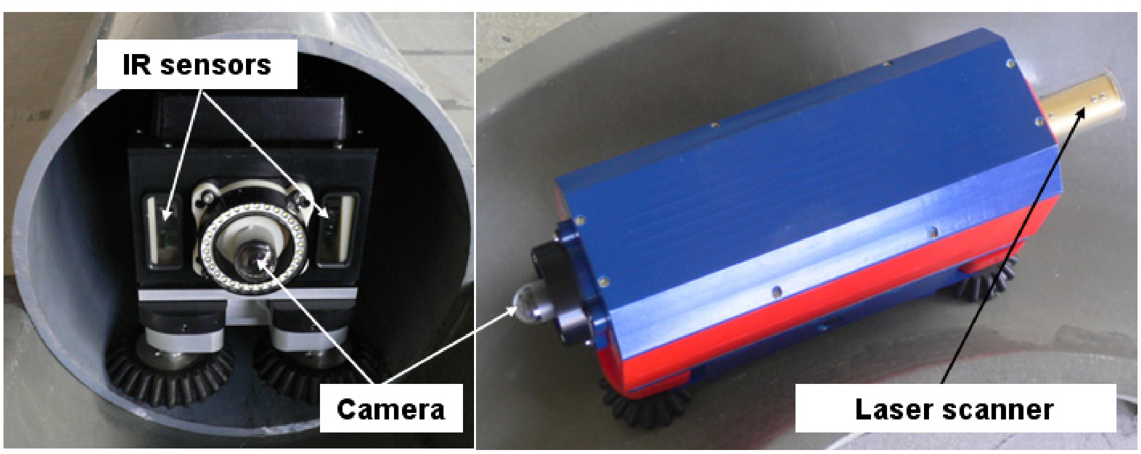


Figure 6. KANTARO system in field test pipe.(Nassiraei et al., 2007)

“KANTARO’s sensor system includes an intelligent laser scanner, one fish eye camera, two IR sensors, and an inclination sensor” (A. Nassiraei et al., 2007: 141). With this sensor selection and configuration, KANTARO has the capability to inspect pipes for fault-detection, correct sensor measurements for robot tilt and rotation, and avoid obstacles. Additionally, it uses a robust microprocessor that uses navigational landmarks such as manholes and pipe joints for navigation, called “fault-navigation” or “localization” (A. Nassiraei et al., 2007: 137). Three separate programs automate the analysis of sensor data by: (1) distinguishing landmarks from sewer features, (2) classifying sewer features into one of nine distinct sewer distresses via a patented fault detection algorithm, and (3) determining the location of the defect within the pipe network.

KANTARO has not been commercially marketed since its development in 2007. Nassiraei, Honda, and Ishii (2010) have continued to develop the autonomous localization of the KANTARO concept by adding passive arms mounted to the naSIR platform. The main reason this technology is not readily available within the sewer inspection industry is that “these complexities in mechanism and data processing make [it difficult] to realize reliable commercial products, especially for [small diameter] pipes” (A. Nassiraei et al., 2007: 137).

Automated Crack Detection

There are two main concepts that are relevant in automated crack detection: computer vision techniques and mathematical modeling. Computer vision generally refers to using computers to process two-dimensional camera imagery into real-world information (Klette, 2014). Mathematical modeling involves using mathematical relationships between inputs and outputs to predict a response, typically identifying an equation or using software to build a model. Using one or both of these concepts with respect to sewer pipe condition is not a new area of research—there are numerous journal articles detailing research on innovative approaches using both CCTV footage and multi-sensor UGVs. The following paragraphs discuss a sample of the work in the computer vision and math modeling areas as they relate to this research.

Computer Vision Techniques

McKim and Sinha (1999) apply computer vision techniques to automatically assess the structural condition of underground sewer pipes using SSET imagery. The researchers used an image enhancement method to eliminate non-uniform background noise (e.g. pipe joints, landmarks, and changes in lighting) and increase the probability of successful detection and processing. They then used image segmentation to partition an input image into its constituent parts or raw pixel data, and ran a line detection algorithm for statistical differences in the mean and standard deviation of pixels between images. Although the authors do not present actual results, the study does show that "elimination of non-uniform background [noise] without assuming any particular statistical distribution for the source image gray-levels" is feasible (McKim & Sinha, 1999: 36).

Guo et al. (2009) studied sewer pipe defects using a combination of pattern-recognition technologies and change detection through "frame differencing". Frame differencing involves a pixel-by-pixel comparison of an image to a pre-selected reference image (Guo et al., 2009) . The comparison results in the presence or absence of a defect in an image by means of statistical analysis. A simplified summary of the change detection method is illustrated in Figure 7.

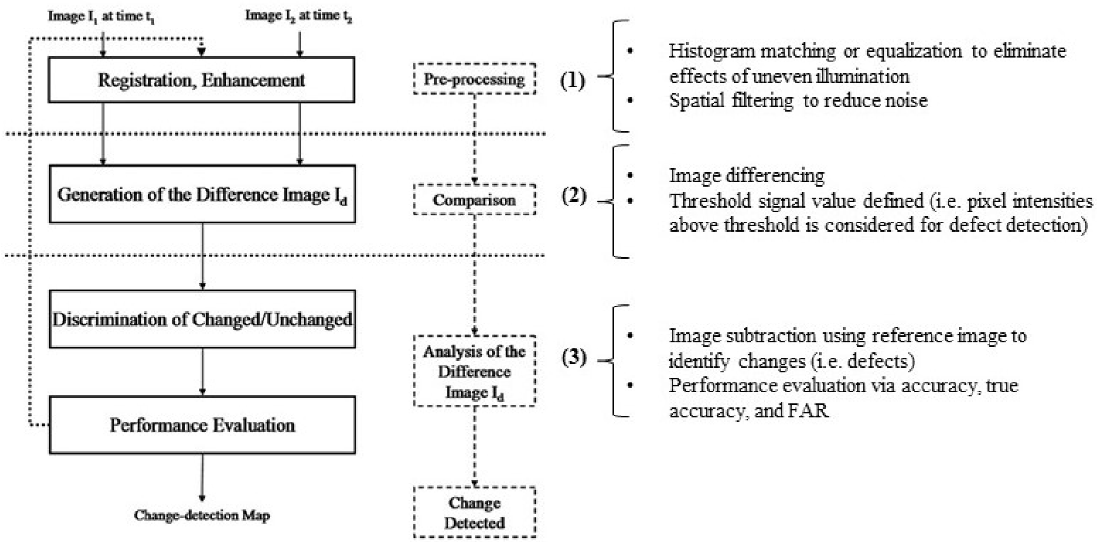


Figure 7. Change detection approach, modified from (Guo et al., 2009)

Guo et al. considered several sewer pipe defects in their research including cracks, surface corrosion, and landmarks (e.g. joints). The researchers collaborated with RedZone Robotics® on a case study of wastewater utility systems around Pittsburgh, PA, where existing CCTV inspections were available. Guo et al. tested the proposed change detection approach using a dataset of 103 CCTV images, taken by RedZone Robotics®, of a 60-meter length of storm water pipe in Pittsburgh. The researchers used the certified CCTV inspector results as a “ground truth” for comparison to the change detection experimental results. The study measured performance using three metrics (Guo et al., 2009):

1. Accuracy: defined as "the percentage of correct detections, including both correctly detected defects and non-defects, among all the images"

2. True Accuracy: defined as the "the percentage of all the predicted images excluding the missed defective images among the entire actual images under analysis"

3. "False alarm rate" (FAR): defined as "the false positive rate…the probability of false detection"

The overall results of the experiment found the change detection method yielded 84% accuracy, 95% true accuracy, and 21% FAR. Based on these results, the researchers found that the change detection method was useful for preliminary defect detection only but could not fully replace human evaluation of results. The change detection method facilitated faster CCTV condition assessments by reducing the workload of certified inspectors, who could focus on regions a high quantity of positive detections to distinguish the false alarms from true positives.

A similar approach was used by Zou, Cao, Li, Mao, & Wang (2012) for crack detection in pavement images. Although Zou et al. focus solely on one defect (i.e. cracks) and in a different infrastructure system (i.e. pavements), the overall methodology is the same. The researchers detection method, called CrackTree, breaks down into three basic steps (Zou et al., 2012: 227):

1. Image enhancement to remove background noise: the researchers used a geodesic shadow-removal algorithm

2. Crack fragment connection using tensor voting: this produces a crack probability map, which the researchers used to enhance the connection of crack fragments with good proximity and curve continuity

3. Minimum spanning trees and tree-edge pruning: researchers used these methods to identify desirable cracks and further reduce noise and potential false positives

Zou et al. tested CrackTree using 206 images of various cracks in pavements. The researchers use Precision, Recall, and F-measure as performance evaluation metrics. Both Guotte & Gaussier (2005) and Ting (2011: 781) define these metrics as:

- Precision: the ratio of true positives assigned by the algorithm to total positives assigned by the algorithm (i.e. true positives + false positives)
- Recall: the ratio of true positives assigned by the algorithm to the actual true positives possible identified in the ground truth (i.e. true positives + false negatives)
- F-measure: a single measure of algorithm performance; also the weighted harmonic mean of Precision and Recall

The confusion matrix in Table 2 is used to classify every output from the algorithm for use in the measures of Precision and Recall (Ting, 2011). True positives (TP) are when the algorithm finds a crack that the ground truth also identified. True negatives (TN) are where the algorithm does not detect a crack, and neither does the ground truth. While false positives (FP) are when the algorithm finds a crack, but the ground truth confirms it does not exist. Conversely, false negatives (FN) are where the algorithm does not detect a crack that ground truth identifies is present.

Table 2. Confusion matrix used to define Precision and Recall, modified from (Ting, 2011)

		Algorithm Prediction	
		Positive	**Negative**
Ground Truth	**Positive**	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

In practice, the Precision, Recall, and F-measure metrics are very similar to Guo et al. use of accuracy, true accuracy, and FAR. The final results of the crack detection in pavement images using CrackTree was 79% Precision, 92% Recall, and 85% F-measure.

Mathematical Models

Several studies have used the logistic regression models to predict condition in sewer pipes, however this research only highlights two. Koo and Ariaratnam (2006) successfully validated research where pipe age, cumulative volumetric flow, and slope were significant inputs to determining if a pipe has failed or not. However, this binary regression model relegated pipe condition into two states, failed and not failed, and does not account for any intermediate states.

Tran, Perera, and Ng (2009) compared using Probabilistic Neural Network (PNN) and Multiple Logistic Regression Models (MLRM) to predict the structural deterioration in storm water pipes. The researchers used influential factors of pipe size, depth, and age in the MLRM. “The results showed that the PNN model was more suited for modeling the structural deterioration of storm water pipes than the MLRM” (Tran et al., 2009: 553).

AFIT Research in Autonomous Drones

In 2015, an AFIT study explored using UAVs and computer vision algorithms as a viable way of performing autonomous pavement assessments of asphalt roads. The research used a fixed-wing Telemaster unmanned aerial vehicle (UAV) to collect pavement images at Camp Atterbury, Indiana using a Prosilica GE1660 camera travelling at an altitude of approximately 100-200 feet and speed of 25 mph (Grandsaert, 2015).

Images taken with this UAV system were 2 megapixel grayscale format. A total of 30 images were used as a sample data set for analysis.

The researcher successfully developed a crack detection algorithm to process the pavement imagery based on CrackTree (Zou et al., 2012). The algorithm uses pixel thresholding to determine the surrounding intensity level of each pixel and determines a thresholding value as the maximum intensity-difference in the image. Next, the algorithm performs a logical connection query that plots a graph of potential edge pixels and connects points that are within 40 pixels of each other. To reduce the runtime for this connection query, the researcher used a KD-tree method of indexing multi-dimensional search trees. Finally, minimum spanning trees using Kruskal's algorithm was used to detect cracks refine edges. The research implemented the algorithm in Python computer language using a an Intel® Core™ i5 (1.8 GHz processor), 120 GB solid state hard drive, 8 GB of RAM, running a Linux Ubuntu version 14.1 operating system.

Grandsaert (2015) established a ground truth by hand marking the visible cracks in each pavement image in Microsoft Paint. The algorithm compared the ground truth images to the results in order to evaluate the effectiveness of the algorithm using Precision, Recall, and F-measure calculations. The algorithm did not successfully perform at the optimal thresholding value. However, after experimentation at varying threshold intensity shifts, results yielded a maximum F-measure of 40% in his field testing. Figure 8 shows the qualitative results of the research.

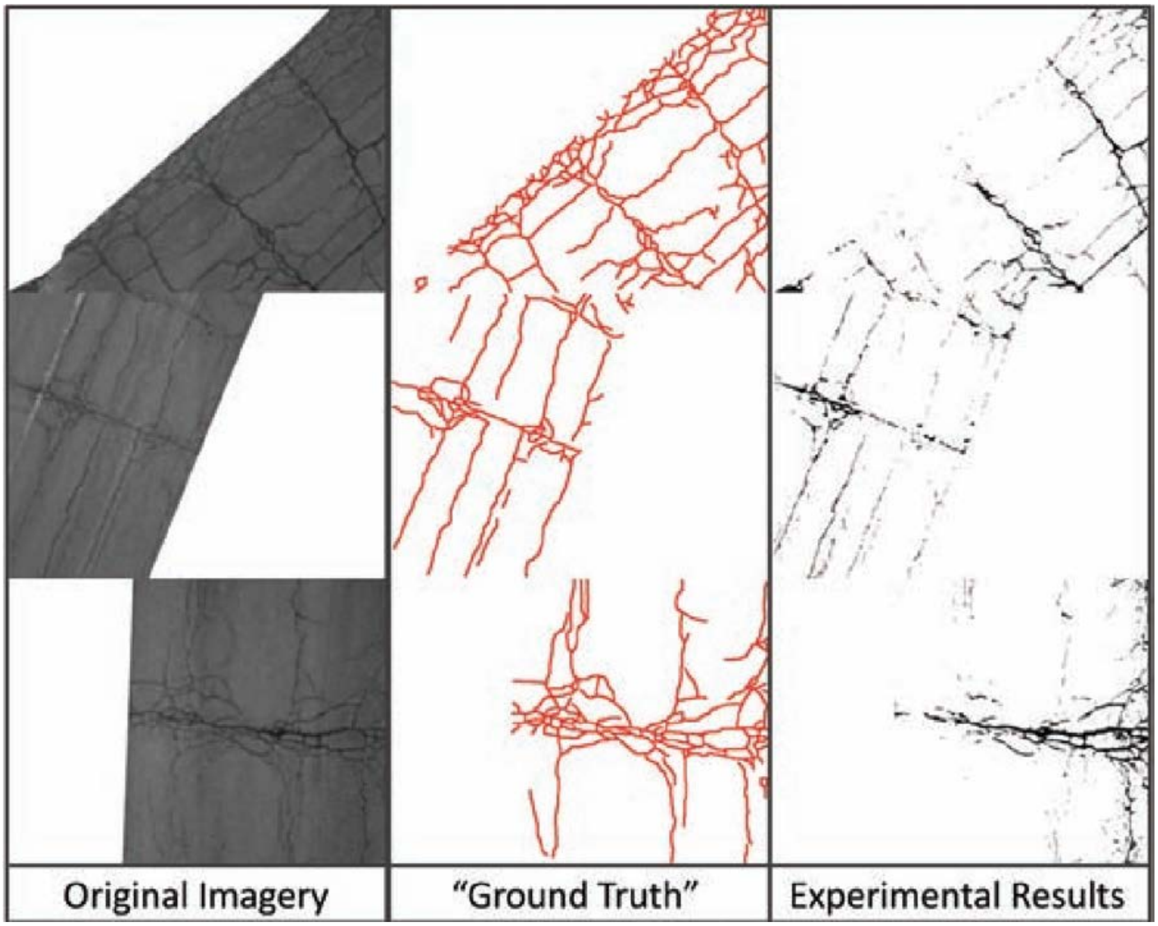


Figure 8. Experimental results of AFIT research (Valencia & Grandsaert, 2015)

Although this work focused on aerial imagery for roadways, it laid the foundation for the proposed research in this thesis. Both concepts involve the use of drones for infrastructure asset management surveys. The drones vary in that one collects data from the air and the other from the ground. However, both drones integrate vehicle, sensor, and algorithm technology for the singular purpose of collecting data. Crack detection, both in roadways and storm water pipes, is used in both studies as a litmus test for infrastructure AM assessment. The overall methodology behind the roadway crack detection algorithm is based on Zou et al.'s CrackTree, and is very similar to the change detection methodology used by Guo et al to find defects in sewer pipes. Therefore, the same algorithm is compatible with crack detection in storm sewer pipes in this thesis.

Summary

This chapter provided a summary of key concepts related to this research. Specifically, this chapter first addressed the relevance of performing in-service assessments in the AM process and system. It went on to discuss applications of UGV technology in civil engineering, specifically in underground pipe networks. This chapter also examined recent attempts to use computer vision techniques and mathematical modeling to autonomously quantify pipe condition as an alternative to humans visually recognizing and classifying pipe defects on CCTV footage. Finally, a summary of recent AFIT research regarding using autonomous drones to perform AM assessments was presented as a model for this research.

III. Methodology

Chapter Overview

This chapter discusses four aspects of the research method. First, a summary of the system architecture provides a context for the final conceptual design used for the data collection system fabrication. Second, the equipment used to collect the pipe condition data on the prototype is detailed. This chapter provides details regarding the sensor technology, unmanned ground vehicle (UGV), and automated navigation system used. Next, the chapter summarizes the automated evaluation of condition based on processing the imagery through computer vision algorithm. Finally, the field testing scenario used to collect data for analysis is generally described.

Systems Architecture

Systems architecture is useful for conceptualizing, designing, and building unique or complex systems. At its core, system architecture is a management tool that facilitates decisions for system development. Specifically, it outlines the structure of components, the relationships, and the principles and guidelines governing the design and evolution of a system (ISO, 2010).

This research culminates in an autonomous system, consisting of a drone, sensor, and algorithm technology, designed and fabricated for the express purpose of collecting storm sewer data. Systems architecture allowed the researcher a scalable structure for problem solving and planning for this system prototype. Additionally, the researcher used the system architecture detailed in this chapter and in Appendix A to select hardware and software for the prototype construction.

ASSETS Architecture

The system is collectively referred to as Automated Storm Sewer Evaluation Technical System (ASSETS). The Department of Defense Architecture Framework (DoDAF) Version 2.0 is the basis for ASSETS system architecture. DoDAF is comprised of different viewpoints, sets of architectural data organized around central concepts. The viewpoints used in ASSETS architecture are explained in Table 3.

Table 3. DoDAF Viewpoints used in ASSETS Architecture (DoD Deputy Chief Information Officer, 2015)

Viewpoint	Description
Capability Viewpoint (CV)	CV describes a vision for performing specified activities to achieve desired resource states under specified standards and conditions. It applies specified guidance and specified performers to those tasks.
Operational Viewpoint (OV)	OV describes organizations, activities they perform, and resources they exchange to fulfill DoD missions. This viewpoint includes the types of information exchanged, the frequency of such exchanges, the activities supported by information exchanges, and the nature of information exchanges.
Systems Viewpoint (SV)	SV describes system activities and resources that support operational activities.
Data and Information Viewpoint (DIV)	The Data and Information Viewpoint (DIV) describes information needs, data requirements, and the implementation of data elements within an architectural description. This viewpoint includes information associated with information exchanges in the architectural description, such as the attributes, characteristics, and inter-relationships of exchanged data.

Systems modeling language (SysML) using Sparx® Enterprise Architect™ Version 10 and Visio® software provided visual modeling capability for ASSETS. This research presents a proof of concept effort to automate storm sewer evaluations for the USAF. Therefore, a complete system design is not the focus of this study. Instead, this research targets only essential capabilities, functions, requirements, and data/resource flows.

Terminology

Even though the design for ASSETS is simplified to only essential requirements, the architecture has numerous terms and acronyms that require explanation for consistent interpretation. Table 4 contains a summary of key terms and definitions used for the system architecture and referenced in later sections. The system elements listed in Table 4 are either physical components of the system (i.e. entity item), external systems that provide data to ASSETS (i.e. actor), or a requirement achieved by ASSETS (i.e. capability). The element type, the second column in Table 4, clarifies this distinction.

Table 4. Summary of Key Terms for ASSETS Architecture

Element	Type	Definition
ASSETS	Entity Item	Automated Storm Sewer Evaluation Technical System - the system being architected.
ASSETS Component - Data Analysis System	Entity Item	ASSETS system component - contains a mathematical algorithm that ultimately quantifies the condition of the pipe.
ASSETS Component - Drone	Entity Item	The self-contained data collection system that would be capable of detecting the presence and location of damages inside of storm sewer pipes and collecting asset attribute data (location, diameter).
ASSETS Component - Relay Point	Entity Item	ASSETS system component - The system to transfer data from the Drone to the Data Analysis System.
ASSETS Component - User interface to Data Analysis System	Entity Item	ASSETS system component - the medium for engineer to manipulate/work with data analysis system.
ASSETS Component - User interface to Drone	Entity Item	ASSETS system component - the medium for utility craftsman to manipulate/work with drone.
Condition	Entity Item	A quantified measure of the physical and functional integrity of the pipeline compared to its initial state when constructed and installed.
Data	Entity Item	Measurements and statistics collected together for reference or analysis of the storm sewer pipe.

Mission plan	Entity Item	Existing information to be uploaded to the ASSETS prior to deployment. Tentatively will include: 1. Existing pipe attribute data to be verified 2. Pre-determined route that the ASSETS will survey
Pipe database	Actor	Storage system for pipe characteristics.
Pipe measurements	Capability	Relevant data about the storm sewer pipe that will be collected/recorded during the evaluation. Including (at this time): 1. Location in 3D space 2. Diameter 3. Surface features
Retrieval point	Entity Item	A location that can be used to either deploy or retrieve the ASSETS from the storm water network. The most typical example is a manhole.
Infrastructure Management Software System	Actor	Sustainment Management System (SMS), a software system used by Civil Engineering community to manage infrastructure assets. Examples: BUILDER, PAVER, GIS
Waypoint	Entity Item	The geographic coordinates or spatial reference of a specific location.

Assumptions and Constraints

The following assumptions and constraints were taken into account when defining requirements for ASSETS architecture:

1. This research presents a proof of concept effort to automate storm sewer evaluations for the USAF, therefore, a complete system design is not the focus of this study.
2. Drone will be deployed only when storm sewer pipes are mostly dry (less than 1 inch depth of water).
3. Drone navigation can occur without external inputs.
4. Drone shall have minimum slippage on pipe surface during transit.
5. Drone shall be operational in pipes having a diameter between 8 inches and 36 inches.
6. Mission Plan, generated from existing pipe geographic information system (GIS) database, shall include coordinate data for waypoint navigation.

Concept Definition

ASSETS is a system comprised of the following: (1) an autonomous drone integrated with sensors, hardware, controllers, and data storage; (2) a separate data analysis system with an algorithm to evaluate inputs and determine the condition of the pipe; and (3) a relay point between the drone and the data analysis system. There are also two distinct user interfaces for the drone and data analysis system.

The Operational Concept (OV-1) for ASSETS, Figure 9, serves as a graphical overview of the system capabilities, components, and relationships of stakeholders that interact with the system. Before deployment in the field, the autonomous drone receives mapping information from a pipe database. The human operator deploys the drone through a manhole. While the drone is in the storm sewer, it autonomously measures and detects different features with minimal input from human operators. After a mission, the human operator retrieves the drone. The pipe measurements from the drone are used in the algorithm to quantify a pipe condition. The algorithm updates the pipe database and sends the condition quantity to the infrastructure management software. Ultimately, the base civil engineer uses the infrastructure management software to make decisions regarding maintenance and repair investments.

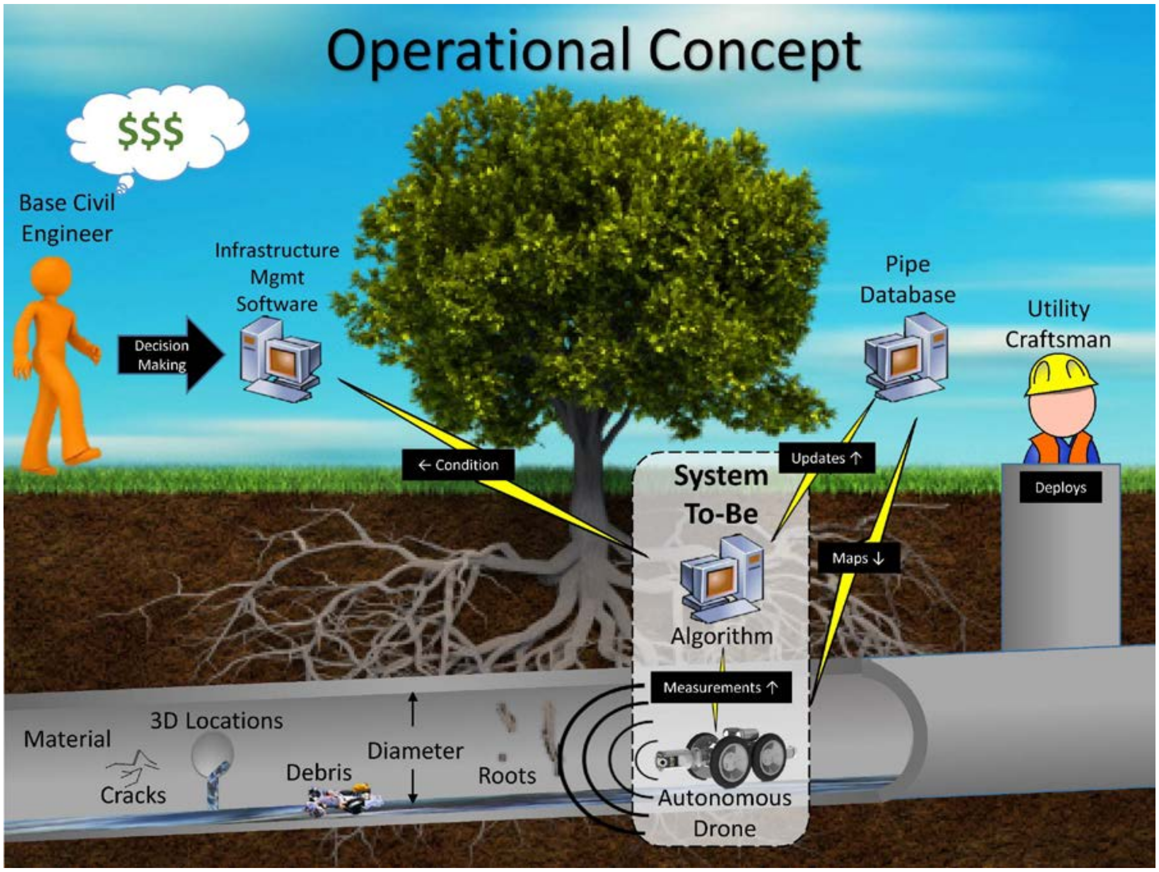


Figure 9: High Level Operational Concept (OV-1) - Created using images from (Alibaba.com, n.d.; Clipart, n.d.; Shel-Daat, n.d.)

The Operational Activity Decomposition Model (OV-5a), Figure 10, breaks down ASSETS field inspection capabilities into operational activities. This hierarchical structure defines the basic functionality of ASSETS and enables the researcher to identify adequate system components for each function.

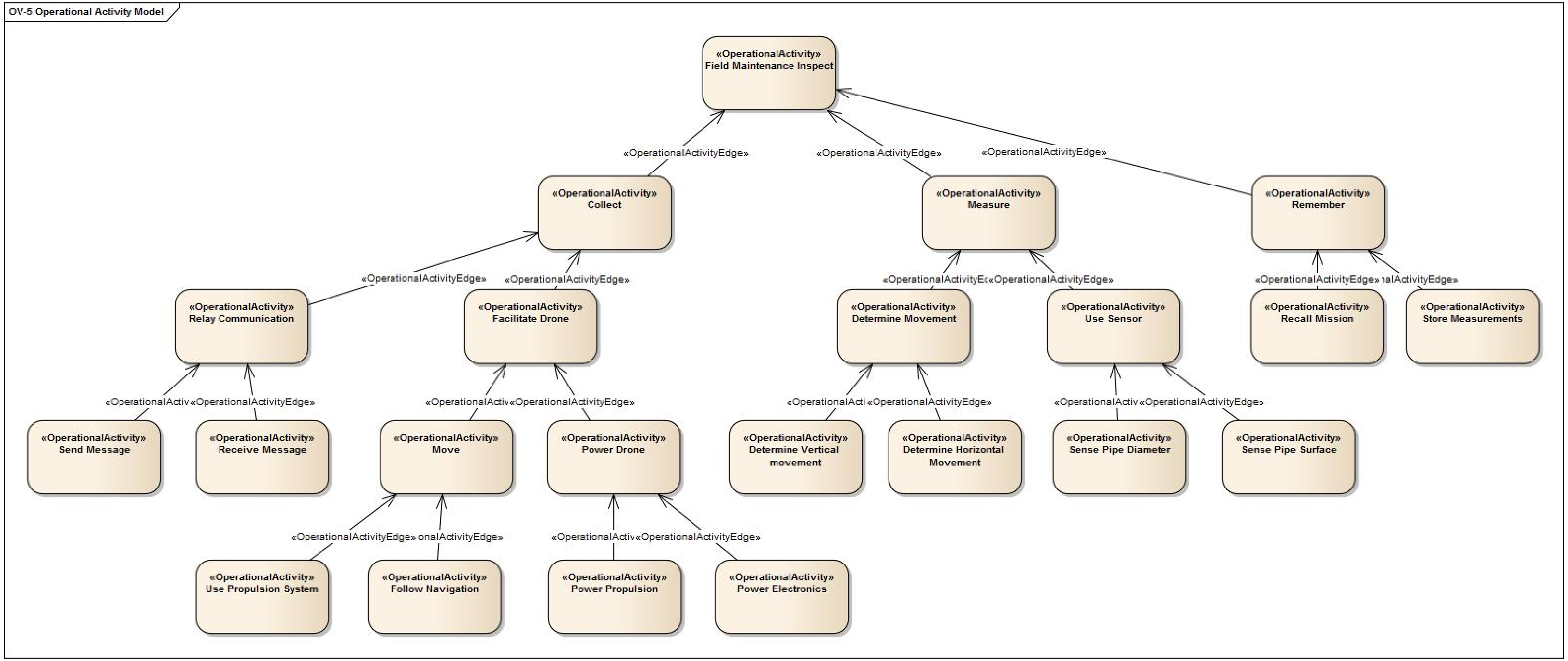


Figure 10. Operational Activity Decomposition Model (OV-5a)

The researcher used the OV-5a to create the modified SV-7 traceability matrix in Table 5, which identifies which component performs which function (i.e. OV-5a capability). This allowed the researcher to select equipment for each function using commercial off the shelf (COTS) hardware and open-source software. An analysis of this table indicates that the existing system design is not overly robust—only critical capabilities are mapped to system components. Additionally, it does not account for redundancies to key operational activities such as "Power Propulsion"—in other words, there is no backup if the drone vehicle propulsion should fail. The lack of redundancy is a system limitation. However, since the focus of this architecture is only critical activities, the fact that at least one component is assigned to each operational activity confirms a complete conceptual design.

Table 5. Operational Activities to Systems Components Traceability Matrix (modified SV-7)

	Drone: Data Storage	Drone: Battery #1 (propulsion, computing)	Drone: Battery #2 (sensors)	Drone: Auto Pilot	Drone: Comm Modem	Drone: GPS	Drone: LiDAR sensors	Drone: Receiver	Drone: Vehicle Propulsion System (rover)	Drone: Video Camera	Drone: Video Transmitter	Groundstation: Computer	Groundstation: Modem	Groundstation: Software Module	Groundstation: Video Receiver	SV-1::Drone Memory Storage	SV-4::Operate Propulsion Components	Utility Craftsman: R/C Radio
OV-5::Determine Horizontal Movement				X		X						X						X
OV-5::Determine Vertical movement				X		X						X						X
OV-5::Follow Navigation				X		X						X	X	X			X	X
OV-5::Power Electronics			X															
OV-5::Power Propulsion		X																
OV-5::Recall Mission														X		X		
OV-5::Receive Message					X			X					X		X			
OV-5::Send Message					X							X	X					
OV-5::Sense Pipe Diameter							X				X							
OV-5::Sense Pipe Surface							X			X	X							
OV-5::Store Measurements	X															X		
OV-5::Use Propulsion System									X								X	

Equipment

The researcher selected all system components using the complete ASSETS architecture (Appendix A). In the interest of time and cost constraints on this study, the researcher selected commercial-off-the-shelf (COTS) hardware and open-source software for all components. An itemized listing of all components and associated costs are in Table 6 on page 51. The most relevant equipment selection and rationale are detailed in the following paragraphs.

Sensor Technology

The researcher examined several sensor technologies, as detailed in Chapter 2, in selecting an appropriate sensor package for the prototype. Based on the literature review of infrared (IR) thermography, this technology provides superior performance in sewer pipe functional evaluations for crack detection. Unfortunately, a thermal camera was not readily available within the researcher's timeframe. Optical inspection using closed circuit television (CCTV) and sewer scanning evaluation technology (SSET) were available and selected as the most applicable technologies for crack detection.

SSET:

A combination of two laser-based SSET sensors is used for this research effort. The Hokuyo® URG-04X-UG01 scanning laser range finder was selected to scan the interior pipe surface to confirm pipe diameter and detect obstructions. The Hokuyo® was selected based on its 240 degree scan angle, accuracy (+/- 1.0 cm), low power demand (5V DC), and relatively small footprint. Although the hardware was integrated into the ASSETS prototype, time constraints impeded the researcher from developing the programming code necessary to use the Hokuyo® URG-04X-UG01 scanning laser in the field.

The Pulsed Light, Inc® LiDAR Lite™ unidirectional laser range finder (Figure 11) was selected to facilitate determining the location within the pipe by ranging the distance between the drone and a reflective board installed at the retrieval point. The LiDAR Lite™ was selected based on its low power demand (6V DC), reasonable accuracy (+/- 2.5 cm), and extremely small footprint.

Figure 11: LiDAR Lite™ range finder (RobotShop, n.d.)

Camera:

The Prosilica® GC1290C camera (Figure 12) is used to capture imagery for this research effort. The camera was selected based on its relatively fast exposure rate (32 frames/sec at 1.2 megapixels) and extremely small footprint (59 × 46 × 33 mm). The small footprint is possible because the camera does not have on-board data storage and rather it transfers imagery to a computer via a gigabit Ethernet port rated at 1,000 MB/sec. This interface allowed the researcher a higher storage capacity of images, but required more complex software control through a third-party computer software developed in Python (Appendix B. Programming Code). The Python code leveraged the free Vimba® software available through Allied Vision Technologies, Inc. and a Python wrapper, known as Pymba, to successfully capture images with the Prosilica® GC1290C camera. A 3-cell 11V lithium polymer (LiPo) battery powers the camera.

Figure 12. Prosilica GC1290C camera (AVT, n.d.)

On-board and Off-board Computers

An Intel® Next Unit Computing (NUC) is used to provide on-board data storage and processing capability to the ASSETS drone. The NUC computer has an Intel® Core™ i5 (1.6 GHz processor), 250 GB solid state hard drive, 8 GB of RAM, and runs a Microsoft® Windows 7™ operating system. A 6-cell 22.2V LiPo battery, connected via a voltage regulator, powers the NUC while operational in the field. The NUC and the 22.2V LiPo battery are relatively large in both size and weight compared to the other ASSETS components. However, the NUC's capability with respect to data processing and storage capability, even with the tradeoff, is superior to other alternatives such as the Raspberry Pii. The NUC contains all software necessary for data collection and analysis for this research, and is essentially the groundstation controller.

In order to make any adjustments in the field, an off-board groundstation laptop computer running TeamViewer™ Version 10 software is used to remote control the NUC. TeamViewer™ software streams the operating system on the NUC to the groundstation laptop display, and allows the researcher to control the NUC via this interface. The researcher used TeamViewer™ to adjust camera settings, run Python and Mission Planner® scripts, and verify images collected during the field test.

UGV

A hobbyist Traxxas® Stampede™ radio controlled car chassis is used for this research effort. Very little modification to the motor, suspension, frame and wheels was done. The aesthetic plastic shell was removed and a platform was attached to the frame, on which most of the other system components were attached.

The SSET sensors, camera, and two batteries were attached to the UGV platform and chassis using 3D printed brackets. The researcher collaborated with a fellow AFIT researcher to pinpoint the design constraints and objectives for 3D printing. The brackets were designed in SolidWorks® to print four separate brackets. The process to create these critical components, from preliminary design to second prototypes, was completed within one week. A full description of all four brackets is documented in Shields (2016). One bracket is detailed below to better illustrate how the 3D printed brackets influenced this research.

Prior to the introduction of the 3D printed brackets, the geometry of the platform limited the configuration of the camera and SSET sensors (Figure 13). The Prosilica® camera was previously mounted to the bottom of the platform, pointing directly forward. This orientation resulted in skewed images starting at approximately 1 meter in front of the UGV. The 3D printed bracket simultaneously improved the vantage point of the Prosilica® GC1290C camera by raising it higher and angling down to the area of interest, and secured the Hokuyo® URG-04X-UG01 scanning laser range finder (Figure 14). The overlap of the camera lens and Hokuyo® does not impede readings as currently tested.

Table 6. ASSETS itemized components and cost summarizes all ASSETS component selections, the desired specification from the systems architecture, and individual and total costs. The researcher set baselines, objectives, and targets based on the assumptions and constraints of the ASSETS architecture. Again, in order to reduce time and cost demands on this study, the researcher selected COTS hardware and open-source software for all components. The overall cost of ASSETS is $4,500, which is relatively inexpensive when compared to other CCTV systems detailed in Chapter II.

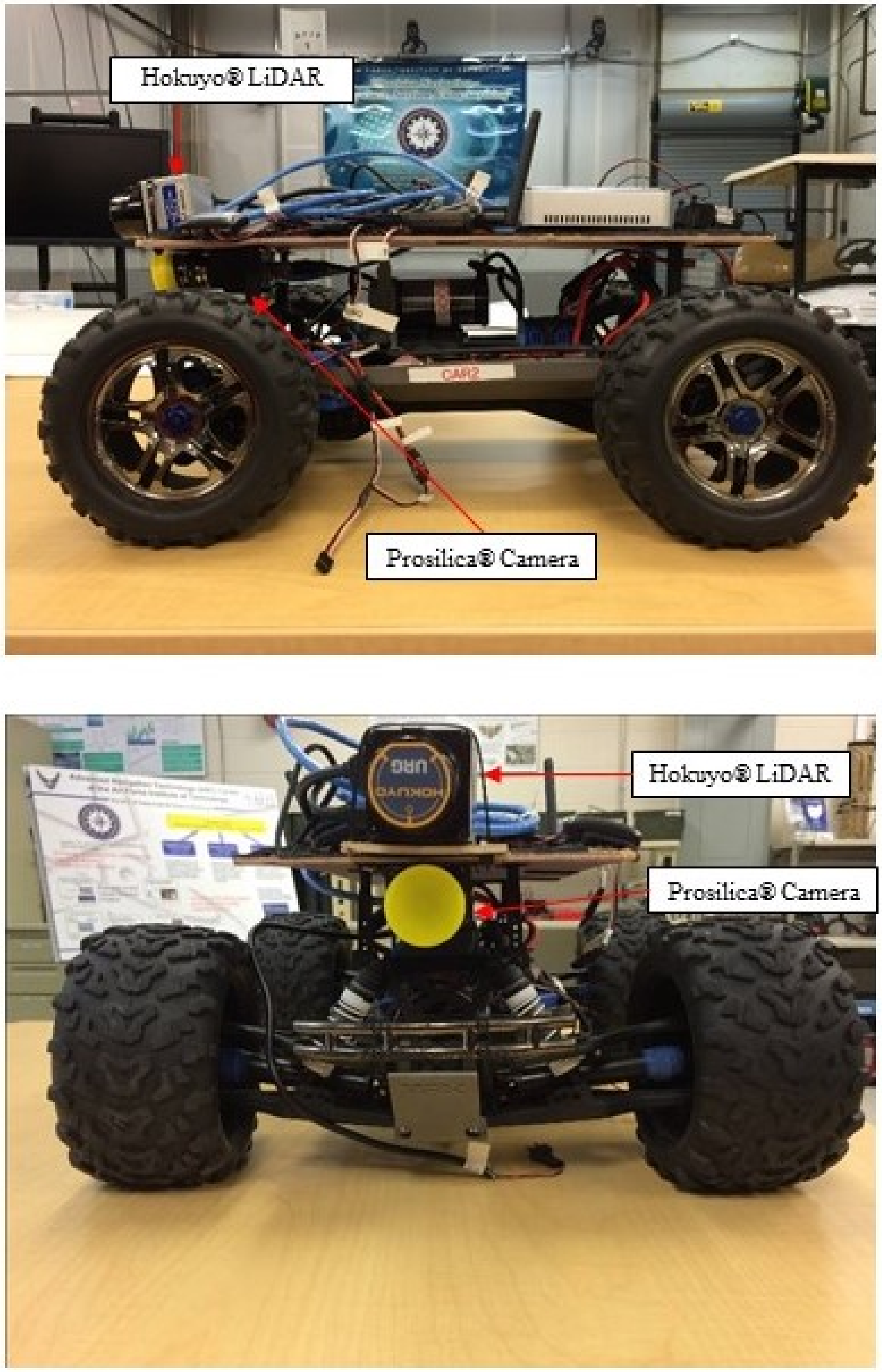


Figure 13. LiDAR and camera attached to platform (prior to 3D printed brackets).

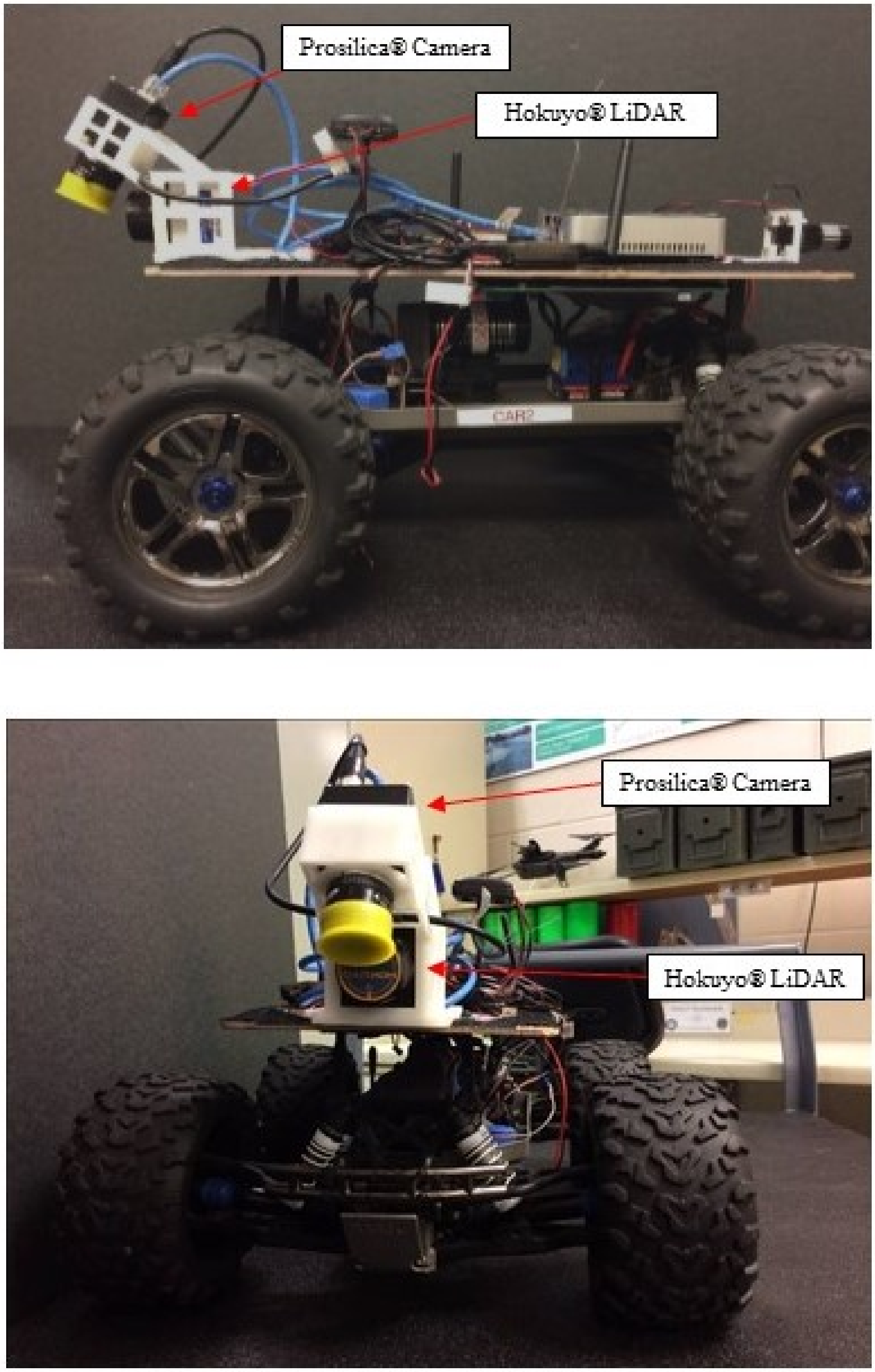


Figure 14. LiDAR and camera installed in 3D printed bracket.

Table 6. ASSETS itemized components and cost

Architectural Element	Type	Measure	Baseline	Objective	Target	Selected Hardware/ Software Component	Cost
Drone: Auto Pilot	Component	The autonomous navigation hardware and software required to navigate the pipe, ideally in the absence of GPS	I2C, PWM	I2C	Compatible with sensor interface requirements	3DR Pixhawk	$ 199.99
Drone: Comm Modem	Component	The frequency on which autopilot communicates with groundstation	915 MHz	915 MHz	The greatest frequency allowable within standards, compatible with autopilot navigation and groundstation software.	3DR Radio Set	$ 100.00
Drone: GPS	Component	Update rate for GPS data to provide for autonomous navigation by autopilot system	5 Hz	5 Hz	Maximum update rate available, compatible with autopilot software.	3DR uBlox GPS with Compass Kit	$ 89.99
Drone: LiDAR sensors	Component	Scan angle as close to 360 degrees as possible (for pipe diameter), minimum range of 60 meters.	20 m	60 m	Maximum coverage for scanning pipe diameter, and maximum range of unidirectional scan for location within the pipe	(1) Pulsed Light - LiDAR Lite (2) Hokuyo - URG-04X-UG01	$ 1,229.95

Drone: Vehicle Propulsion System (rover)	Component	Speed and maneuverability of vehicle	0.25 m/sec	2 m/sec	Minimum speed of 0.25 m/sec (i.e. faster than walking speed), maneuverability to allow for reasonable amount of maneuverability	Traxxas E-Maxx Truck	$ 750.00
Drone: Camera	Component	Fastest image capture with maximum resolution	1 Megapixel	2 Megapixels	Ability to capture images with minimal blur at target inspection speed of 2 m/sec	Prosilica (AVT) GC1290C	$ 1,125.00
Groundstation: Computer	Component					Lenovo Yoga 2.0	$ -
Groundstation: Modem	Component	The frequency on which autopilot communicates with groundstation	915 MHz	915 MHz	The greatest frequency allowable within standards, compatible with autopilot navigation and groundstation software.	3DR Radio Set	$ 100.00
Groundstation: Software Module	Component	Compatibility with autopilot hardware	n/a	n/a	Software system that is compatible with autopilot, and can process waypoints in 3D space.	3DR APM Mission Planner - Rover	$ -
SV-1::Drone Memory Storage	Component	The data storage capacity of the drone	500 GB	>1 TB	Data storage for 4 hrs of sensor measurements	Removable media (e.g. SD card @ 512 GB or USB 3.0)	$ 90.00
SV-4::Operate Propulsion Components	Function	The maximum horizontal velocity of the drone in the pipe	0.25 m/sec	2 m/sec	The greatest velocity allowable with accurate sensor measurements	Rover propulsion	$ -

SV-4::Provide Electricity to: camera	Function	The power/battery capacity available for camera	2,000 mAh	5,000 mAh	Power capacity for 4 hr field deployment	Battery #2	$ 35.00
SV-4::Provide Electricity to: (1) Propulsion & (2) AutoPilot	Function	The power/battery capacity available for vehicle propulsion, autopilot, and LiDAR sensors	2,000 mAh	5,000 mAh	Power capacity for 4 hr field deployment	Battery #1	$ -
SV-4::Provide Electricity to: Computing Component	Function	The power/battery capacity available for vehicle propulsion and computing components	2,000 mAh	5,000 mAh	Power capacity for 4 hr field deployment	Battery #3	$ 80.00
SV-4::Send Measurements to Memory Storage	Function	The speed at which sensor measurements can be converted to memory.	500 MHz	700 MHz	Processing capability to enable a speed of 30 ft/min	Processor (NUC)	$ 350.00
Utility Craftsman: R/C Radio	Component	The frequency on which autopilot commuincates with groundstation	2.4 GHz	2.4 GHz	The greatest frequency allowable, but R/C Radio should be fully compatible with autopilot hardware and software.	FRSKY Taranis PPM-Sum Compatible Transmitter	$ 295.00
						TOTAL COST =	**$ 4,444.93**

Computer Vision

This research uses the same crack detection algorithm developed in Grandsaert (2015) for detecting cracks in pavement images, which is based on the CrackTree concept (Zou et al., 2012). As previously mentioned in Chapter II, the overall methodology Zou et al. use for CrackTree is the same as the change detection methodology used by Guo et al to find defects in sewer pipes. Therefore, the algorithm employed in Grandsaert (2015) is compatible with crack detection in storm sewer pipes in this study. Future research should consider improving the algorithm with robust image enhancement similar to that tested by Guo et al to eliminate non-uniform background noise (e.g. pipe joints, landmarks, and changes in lighting).

Except for minor updates for file path and image size, this research did not adjust the algorithm from the final working code in Grandsaert (2015). The algorithm uses pixel thresholding to determine the surrounding intensity level of each pixel and determines a thresholding value as the maximum intensity-difference in the image. Next, the algorithm performs a logical connection query that plots a graph of potential edge pixels and connects points that are within 40 pixels of each other. A KD-tree reduces runtime for this connection query and finally Kruskal's algorithm is used to create minimum spanning tress and prune edges.

This research evaluates algorithm effectiveness using Precision, Recall, and F-measure metrics, as defined by Guotte & Gaussier (2005) and Ting (2011: 781):

- <u>Precision</u>: the ratio of true positives assigned by the algorithm to total positives assigned by the algorithm (i.e. how many of the cracks that the algorithm found were true cracks), calculated in Equation (1)

$$Precision = \frac{True\,Positives\,Found\,by\,Algorithm}{True\,Positives + False\,Positives} \quad (1)$$

- Recall: the ratio of true positives assigned by the algorithm to the actual true positives possible identified in the ground truth (i.e. Recall is how many of the true cracks the algorithm found), calculated in Equation (2)

$$Recall = \frac{True\,Positives\,Found\,by\,Algorithm}{True\,Positives + False\,Negatives} \quad (2)$$

- F-measure: a single measure of algorithm performance; also the weighted harmonic mean of Precision and Recall, calculated in Equation (3)

$$F - Measure = \frac{2 * Recall * Precision}{Recall + Precision} \quad (3)$$

Field Testing and Validation

Field Testing

Field testing was performed on a concrete-lined drainage channel on Wright-Patterson AFB, OH (Figure 15). The intent of using a concrete-lined drainage channel is to simulate storm pipe infrastructure, but avoid the challenges associated with not having a GPS signal underground. As this research is a proof of concept, successful field testing on a concrete channel shows that the system can autonomously collect images and process the data via computer vision algorithm. Unfortunately, autonomous navigation using the Pixhawk® autopilot was not possible for the field test—the ASSETS drone was operated manually using the Taranis® R/C controller. Preliminary trials at AFIT campus prior to field testing revealed a drift error when the autopilot navigated the drone in place of manual controls. The drift error was too great to compensate for with gain settings in the Mission Planner® software. This can be corrected in future research by additional calibration with the Mission Planner® software settings, or even working with the open source community for a proven solution.

Figure 15. Storm water drainage channel used as field test site, Wright-Patterson AFB OH. (Google, n.d.)

The 50-meter section of channel on the western edge, demarcated with a yellow line in Figure 15, was field tested on 4 December 2015.   The weather that day was partly cloudy and 32 degrees F. This section was tested by manually driving the ASSETS drone through the center of the channel and executing the image acquisition code to capture 50 images. This was completed four times, for a total of 200 images.

Subject Matter Expert (SME) Validation of Ground Truth

Two civil engineers from the 88th Civil Engineer Squadron visually inspected all 200 images collected during the field test to "ground truth" the defects in the drainage channel. As each image was reviewed, the SMEs discussed the image and reached consensus on the presence of a crack in the image. If a crack was detected, one of the engineers used a hardcopy of the image to draw the crack and define the edges. The ground truth information on the hardcopy was transferred to a digital image using Microsoft Picture Manager (to draw the crack) and Microsoft Word (to remove the background). The algorithm compared this digital ground truth to the results created by the computer algorithm. This process with SME inputs and digital file manipulation provided the data set against which the algorithm output could be validated.

Statistical Methods for Evaluating Crack Detection Effectiveness

This research used Precision, Recall, and F-measure to evaluate the effectiveness of the algorithm in detecting cracks. The algorithm calculated Precision, Recall, and F-measure by comparing the algorithm image output to its paired ground truth image and applying Equations (1) through (3), respectively. The fundamental analytical goal of this research was to explain if these factors were statistically different at the various intensity thresholds applied, but more importantly to identify under which scenario the algorithm performed best.

The researcher applied several statistical methods in JMP® v11 in order to explain the variance in F-measure, as this factor takes into account Precision and Recall. An analysis of variance (ANOVA) test was performed to determine what factors explained the variance observed in the F-measure results. The ANOVA was validated by testing its assumptions via the Shapiro-Wilk test for normality of the residuals, the Breusch-Pagan test for homoscedasticity (constant variance) of the residuals, and finally the Durbin-Watson test for residual independence. LS Means plots, Tukey HSD and Student's T tests were performed where applicable.

The researcher tested the associated null hypothesis of each statistical method. Rejection or failure of rejection of each null hypothesis was further evaluated with respect to its significance to the research goal. For this study, the null hypotheses (H_o) and alternate hypotheses (H_a) for each test are defined below:

Overall F-Test

- H_{o1}: None of the factors explains the observed variance in F-measure.
- H_{a1}: At least one of the factors explains the observed variance in F-measure.

Effect Tests

- H_{o2}: The F-measure means of the Images are the same.
- H_{a2}: At least one of the Images has a different F-measure mean.
- H_{o3}: The F-measure means of the Intensity Thresholds are the same.
- H_{a3}: At least one of the Intensity Thresholds has a different F-measure mean.

Shapiro-Wilk W Test

- H_{o4}: The population of the residuals is normally distributed.
- H_{a4}: The population of the residuals is not normally distributed.

Breusch-Pagan Test

- H_{o5}: The residuals display constant variance.
- H_{a5}: The residuals do not display constant variance.

Durbin-Watson Test

- H_{o6}: The residuals are independent of one another.
- H_{a6}: The residuals are dependent of one another.

Summary

This chapter presents an overview of the system architecture, equipment, and computer vision techniques used in this research effort. The ASSETS prototype for this research effort is field tested using a concrete drainage channel at Wright-Patterson AFB, OH. A crack detection algorithm in Python applies computer vision techniques to process the imagery collected by ASSETS. The cracks detected by the algorithm are then compared to a ground truth, established based on a consensus of two expert opinions, which represents the true cracks in the drainage channel. The algorithm calculates Precision, Recall, and F-measure results for each image. These quantitative results are analyzed using an ANOVA to determine what factor, if any, explains the variance observed in F-measure.

IV. Analysis and Results

Chapter Overview

This chapter contains the results of the field testing, SME validation, and algorithm processing. Each section in this chapter describes the relevant observations, processes, or techniques used for the data collection and analysis, and presents an overview of the results. Finally, this chapter concludes with presenting qualitative and quantitative results of the image processing performed by the algorithm. The analysis in this chapter lays the groundwork for Chapter V which interprets the results from the perspective of the research questions.

Results

Field Testing and SME Validation

Field testing occurred on 4 December 2015 from approximately 1330 – 1500 hours local time. The researcher used the ASSETS prototype described in Chapter III to collect a total of 200 images of the 50-meter section of storm water drainage channel (Figure 15). The researcher completed four different trial runs of the same route, referenced as Runs A, B, C, and D. Each trial run collected 50 images, for a total of 200 images.

The exposure settings on the Prosilica® GC1290C camera were adjusted at the field test site prior to Run A for optimal performance using a simple technique. Based on the expertise and guidance of the AVT Technical Services staff, the researcher manually adjusted the camera iris into the fully open position and decreased the absolute exposure time setting (i.e. ExposureTimeAbs) in Vimba® software to 1,513 microseconds. The Vimba® software settings are used when the Python image acquisition code script is executed. These adjustments ensured the fastest exposure time at the specific lighting conditions present at the field test site. This technique improved exposure rate from 0.86 to 1.39 frames per second (fps), and resulted in clearer pictures, as illustrated in Figure 16.

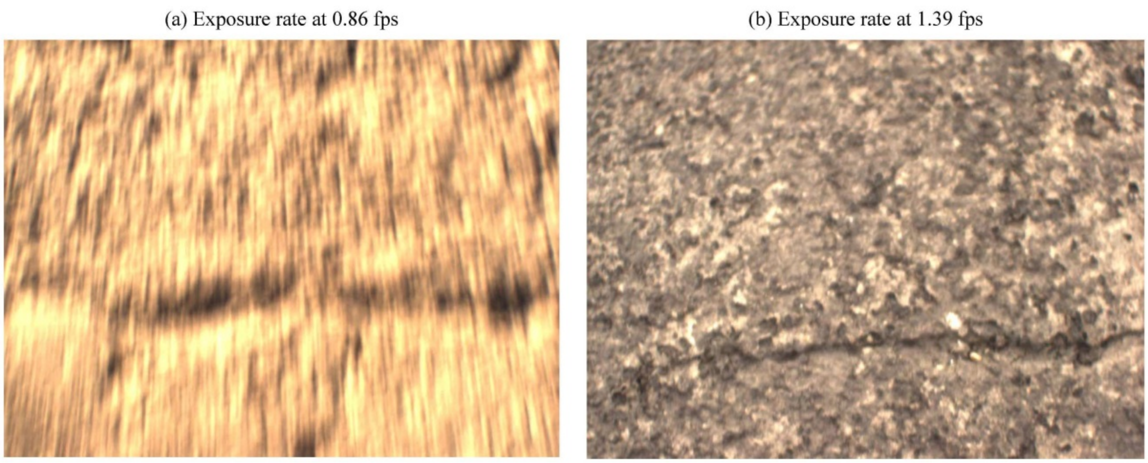


Figure 16. Comparison of exposure settings before (a) and after (b) using improved technique.

The images collected represented a wide range of crack scenarios, ranging from no cracks to many cracks plus debris. Lighting conditions were fairly uniform since the weather remained overcast throughout the field test. However, several images were darker and contained more debris (e.g. leaves) as the last 20 meters of the route was covered with trees.

The SMEs from the 88th CES confirmed cracks in 90 (45%) of the 200 images collected during the field test. If a crack was detected, one of the engineers used a hardcopy of the image to draw the crack and define the edges. The researcher later used the ground truth information on the hardcopy to create a digital image using Microsoft Picture Manager (to draw the crack in red) and Microsoft Word (to remove the background). Through this process, 90 pairs of images (i.e. one original image and one digital ground truth image) resulted from the field testing and

SME validation steps. The researcher used these pairs of images in the algorithm processing discussed in the Image Processing section.

Algorithm Control Test

A control test was performed in order to better understand algorithm performance and to evaluate the image processing results for inaccuracies. The researcher created a control image consisting of two black lines, both 1 pixel wide, going across the length of the image and crossing orthogonally exactly at their midpoints. Line A was 1 pixel wide by 960 pixel long, and Line B was 1 pixel wide by 1280 pixels long. A separate ground truth control image (Figure 17) was created by duplicating the control image exactly, but changing the color to red. This pair of images was used as the control test inputs for the algorithm processing.

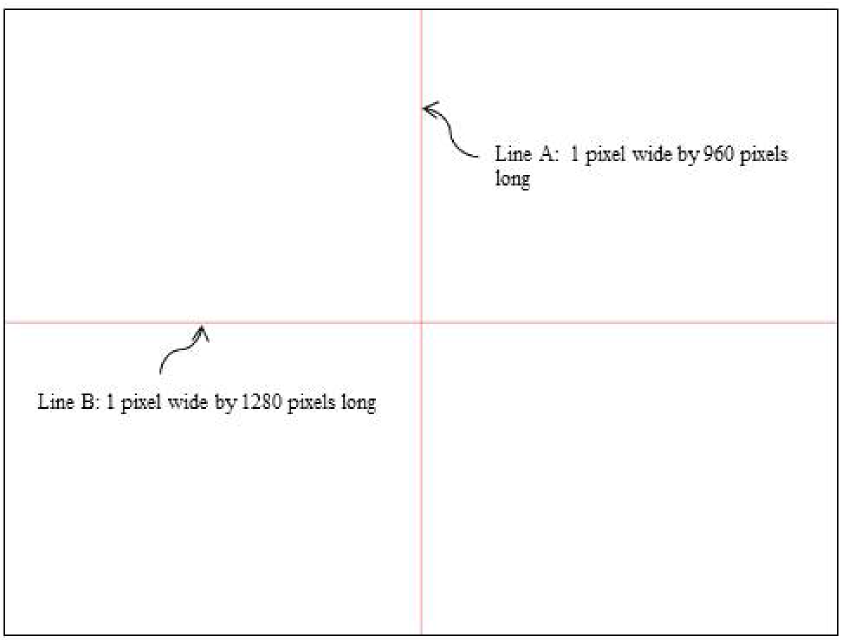


Figure 17. Ground truth control image.

The algorithm processing results are both qualitative (i.e. output images created by algorithm) and quantitative (i.e. calculations of Precision, Recall, and F-measure). The control test output image (Figure 18) visually confirms that the algorithm did a reasonably good job detecting the lines, but with some inaccuracies. First, lines detected by the algorithm were drawn wider than the true width represented in the control image. Both Line A and Line B were recreated 3 pixels wide instead of 1 pixel wide, causing a fairly large amount of false positives. Second, the algorithm missed 8 pixels on the far edge of each line, shown in red circles in Figure 18. These missed pixels resulted in a small amount of false negatives. Finally, the algorithm misrepresented the midpoint crossing of the two lines by omitting 5 pixels (i.e. false negatives) and mistakenly drawing approximately 20 pixels for the "diagonal" connections between the two lines (i.e. false positives).

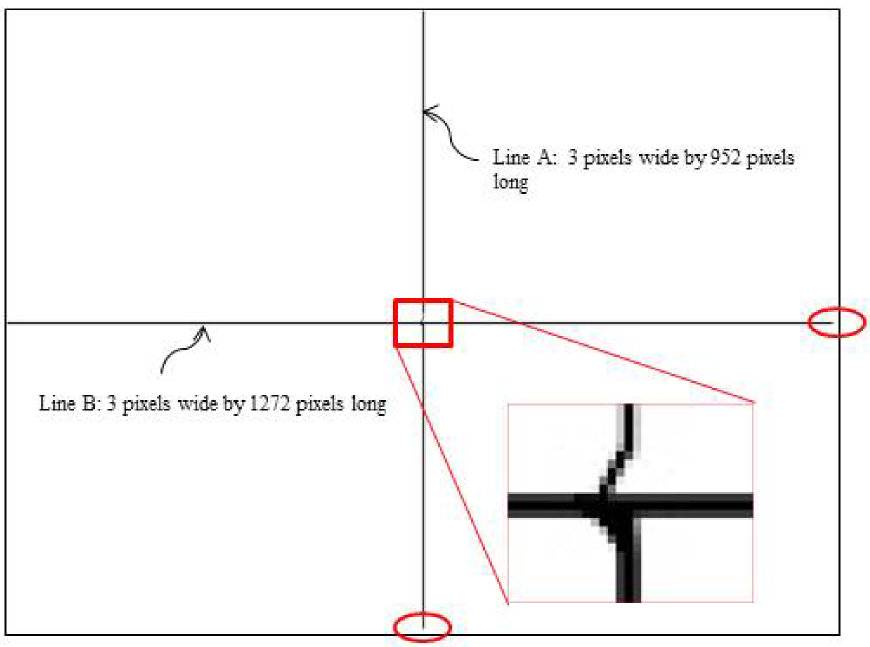


Figure 18. Algorithm output image from control tests, errors highlighted in red.

Based on the evaluation of the qualitative output image, the researcher hand-calculated expected values for Precision, Recall, and F-measure (Table 7) by using Equations (1) through

(3), respectively. The actual quantitative output results calculated by the algorithm are shown in Table 8. The algorithm calculated a Recall of 57.2%, which was surprising given the relatively low amount of false negatives in the output image. Upon further investigation, the researcher found that the equations for Precision and Recall were inverted in the programming code used in Grandsaert (2015). However, since the F-measure is the harmonic mean of Precision and Recall this inversion does not change the overall results presented by Capt Grandsaert. The researcher corrected the inverted labels in the algorithm used for this study. Lines 104-114 of the "Compare5xb" programming code script, found in Appendix B. Programming Code, were modified for the corrected equations for Precision and Recall.

Table 7. Expected quantitative results using control image evaluation.

Precision	33.5%
Recall	99.1%
F-measure	50.0%

Table 8. Actual quantitative results calculated for control test.

Precision	31.1%
Recall	57.2%
F-measure	40.3%

The inverted labels do not fully explain the discrepancies between Table 7 and Table 8. In other words, inverting these values in Table 8 uncovers that the algorithm is not calculating false positives and false negatives as anticipated by the researcher. There are several "tuning parameters" applied by the algorithm that can be adjusted to investigate performance changes. An example of one of these tuning parameters is a 15 pixel tolerance between a found crack pixel and the ground truth pixel (i.e. algorithm will positively count a crack pixel that is in the pixel location +/- 15 pixels in the ground truth image). The researcher did not tune these parameters to refine performance because the actual F-measure was fairly close to expected in the control test. However, this type of tuning is recommended for future research.

Although there were some inaccuracies in the algorithm results, overall the control test validated that the algorithm methodology successfully performed crack detection. The control test identified one error in the programming code from Grandsaert (2015), the inverse of Precision and Recall labels. The programming code was modified to correct this inverse, and was used for the field image processing detailed in the following section.

Image Processing

The algorithm could not process all 90 images at one time due to the limitations of the computer hardware used. A single batch process of all 90 images overwhelmed the NUC's processing memory, and resulted in unusable images starting at approximately image #33. The NUC's Intel® Core™ i5 (1.6 GHz processor) and 8 GB of RAM could not handle the massive amount of potential crack pixels within close proximity in the entire image data set. To avoid this problem, the images were processed in multiple batches. The researcher processed images in four different sets correlating to the four different trials performed during the field test. The largest set was 26 images, and took approximately 9 hours to process at an intensity threshold shift -40. At the intensity threshold -35, however, the system was again overwhelmed due to the increase of potential crack pixels at the lower threshold. For this setting, the researcher processed the images in eight smaller sets of approximately 10-12 images.

In total, the researcher processed the 90 pairs of images through the algorithm at three different intensity threshold shifts: -35, -40, and -45. The algorithm processing results are both qualitative (i.e. resulting images created by algorithm) and quantitative (i.e. calculations of Precision, Recall, and F-measure).

The qualitative results from three representative images at intensity threshold shift -40 are shown in Figure 19. A comparison of the qualitative results of the same three images, processed at the other intensity thresholds, are shown in Figure 20.

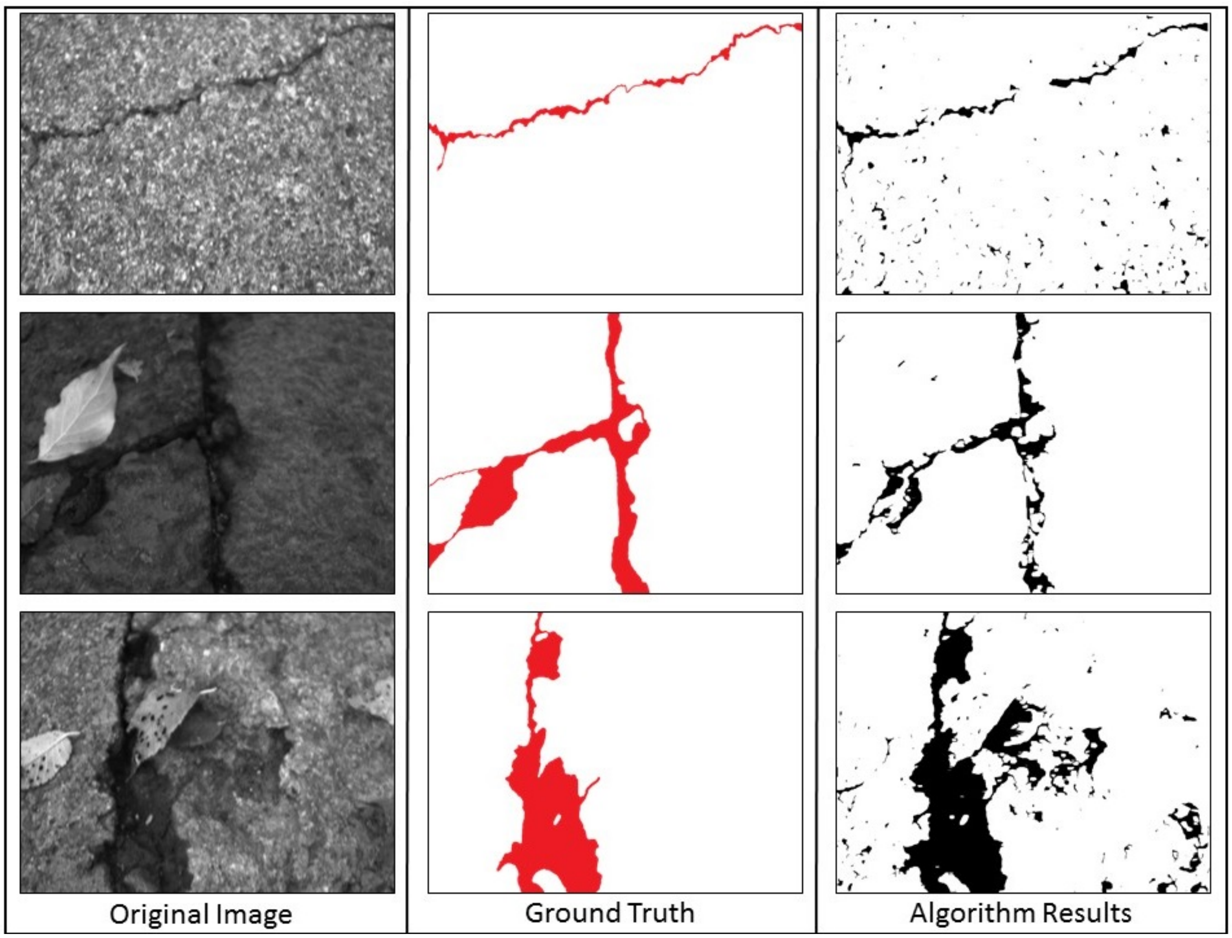


Figure 19. Crack detection on three representative images at intensity threshold shift -40.

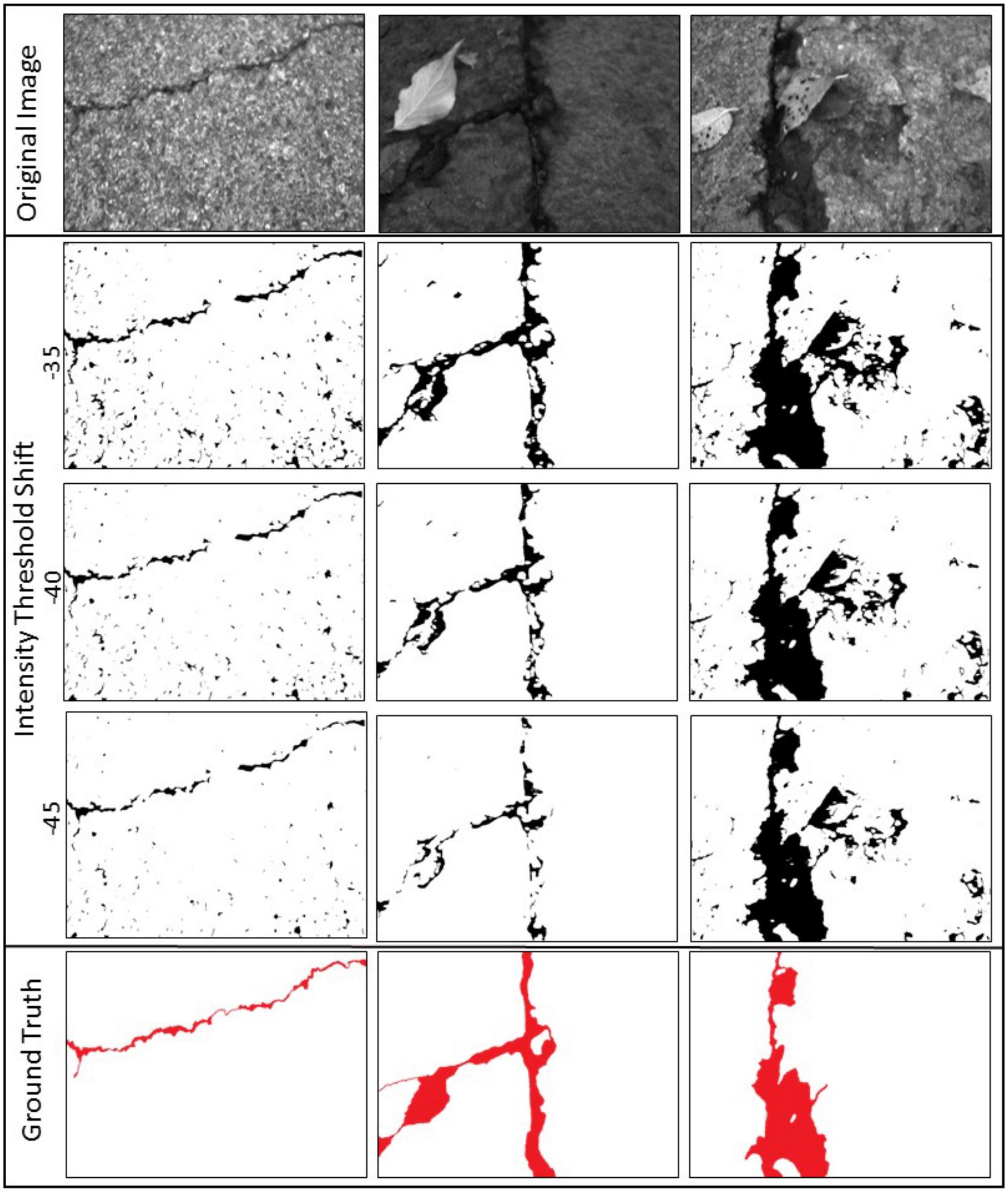


Figure 20. Comparison of algorithm processing results at intensity thresholds -35, -40, and -45.

Once the algorithm produced a resulting image, the program compared the output image to the digital ground truth and calculated a Precision, Recall, and F-measure for each image. The data was compiled in JMP® v11 and two quantitative analyses were performed: (1) interpretation of the summary statistics, and (2) an ANOVA test of the F-measure results (See Appendix C. Quantitative Data).

<u>Interpretation of Summary Statistics</u>

Figure 21 shows that the mean Recall achieved by the algorithm is 97.6% considering all thresholds. This mean includes several outliers, including four instances of an observed Recall of 0% in image #4, 60, 66, and 67 at threshold intensity -45. The researcher suspects that the extremely low Recall in those images is attributed to excessive debris in the images; however no further analysis was performed to confirm this suspicion. The reported mean is a conservative estimate of Recall, and the true Recall could possibly be higher if excluding these outliers was justified.

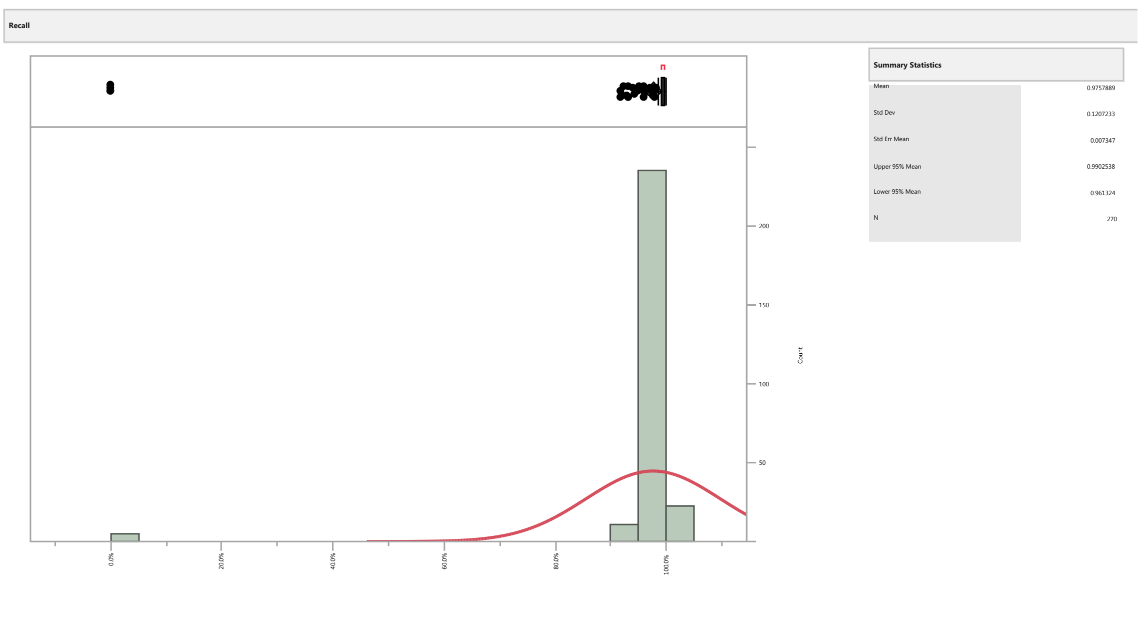
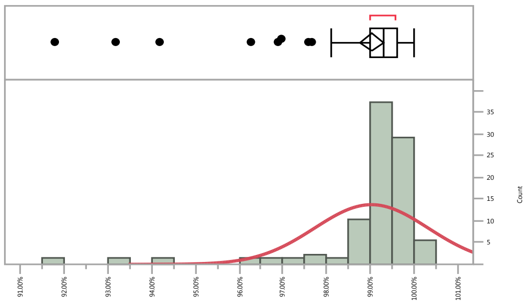


Figure 21. Overall Recall summary statistics (all thresholds)

A breakdown of the observed Recall at each intensity threshold shift using histograms is illustrated in Figure 22. To offer another perspective, a visual comparison of the group means using comparison circles for the All Pairs, Tukey HSD is illustrated in Figure 23. Intensity threshold -45 has the highest observed mean Recall at 99.04%, but intensity threshold -35 was almost equal with less variance.

Distributions Intensity Threshold=35

Recall

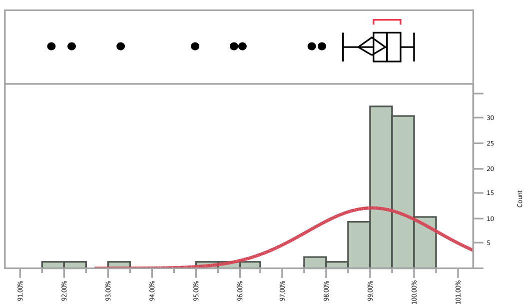


Summary Statistics	
Mean	0.9903333
Std Dev	0.0130505
Std Err Mean	0.0013756
Upper 95% Mean	0.9930667
Lower 95% Mean	0.9876
N	90

Normal(0.99033,0.01305)

Distributions Intensity Threshold=40

Recall

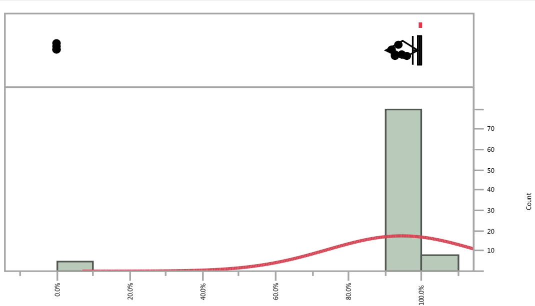


Summary Statistics	
Mean	0.9904667
Std Dev	0.0149074
Std Err Mean	0.0015714
Upper 95% Mean	0.993589
Lower 95% Mean	0.9873444
N	90

Normal(0.99047,0.01491)

Distributions Intensity Threshold=45

Recall



Summary Statistics	
Mean	0.9465667
Std Dev	0.2058203
Std Err Mean	0.0216954
Upper 95% Mean	0.9896749
Lower 95% Mean	0.9034584
N	90

Normal(0.94657,0.20582)

Figure 22. Recall quantitative summary statistics by intensity threshold

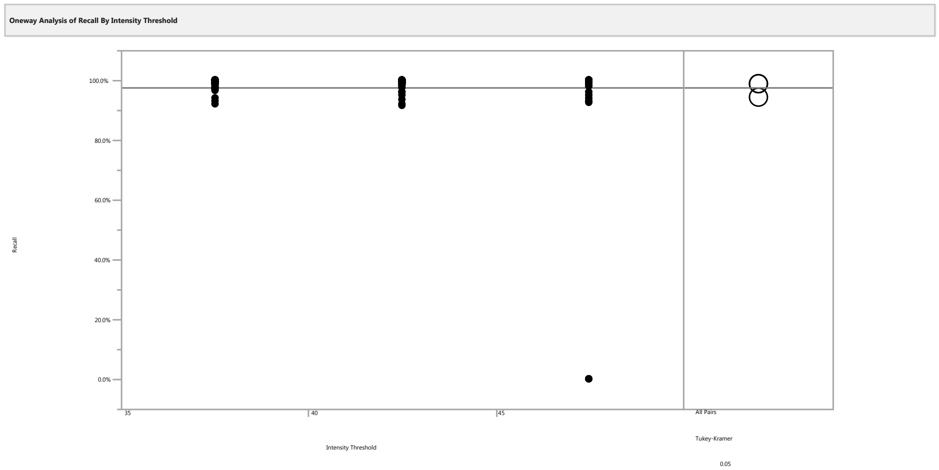


Figure 23. Comparison of Recall group means, with Tukey HSD comparison circles

Figure 24 shows that the mean Precision achieved by the algorithm is 17.2% considering all thresholds. This mean includes four possible outliers on the higher side of the range – image #33 at intensity threshold -40, and images #23, 33, and 40 at intensity threshold -35. In the case of image #33, the researcher again suspects that excessive debris may have affected the measure of Precision; however no further analysis was performed to confirm this suspicion. Additionally, there was no observed debris in images #23 and 80, so the high Precision observed is not fully explained by the presence or absence of debris.

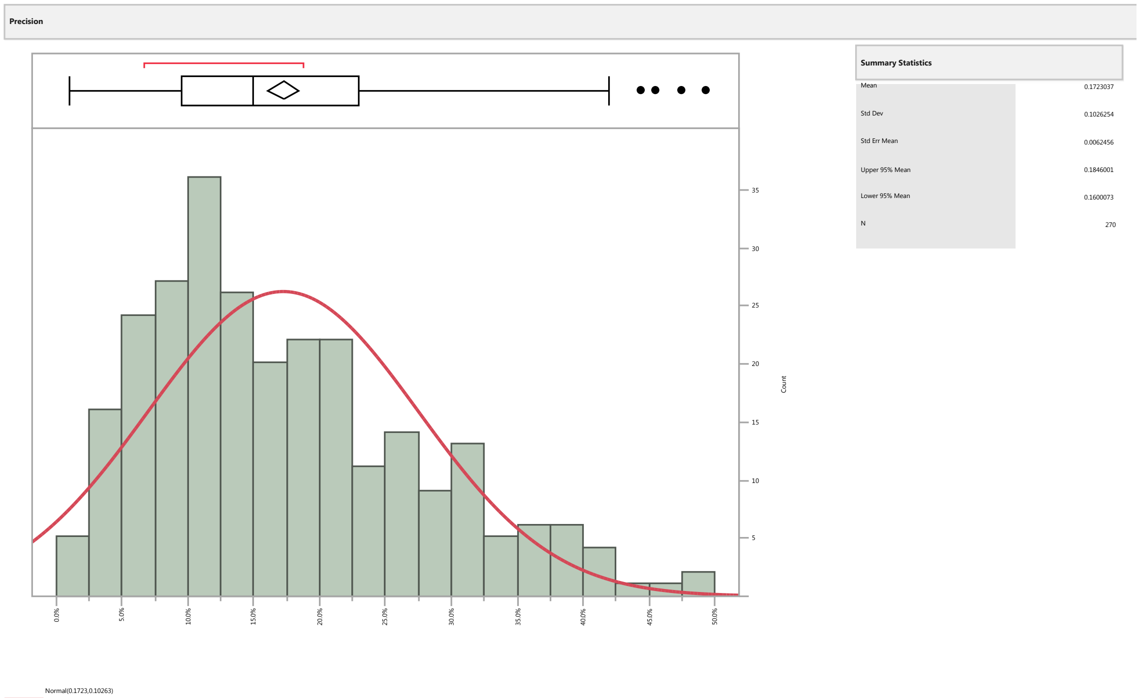


Figure 24. Overall Precision summary statistics (all thresholds)

A breakdown of the observed Precision at each intensity threshold shift is illustrated in Figure 25. To offer another perspective, a visual comparison of the group means using comparison circles for the All Pairs, Tukey HSD is illustrated in Figure 26. Intensity threshold -35 has the highest observed mean Precision at 23%.

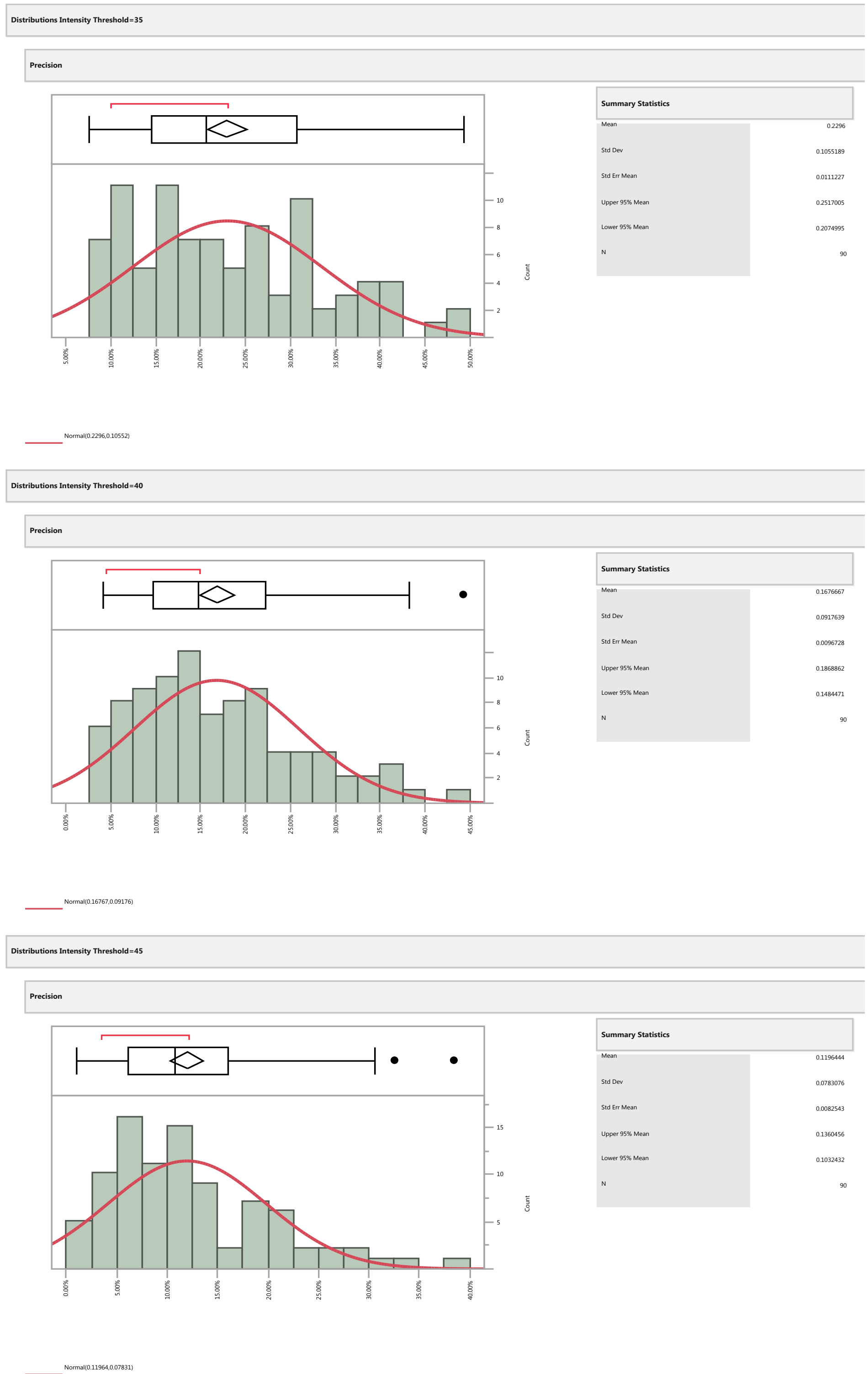


Figure 25. Precision quantitative summary statistics by intensity threshold

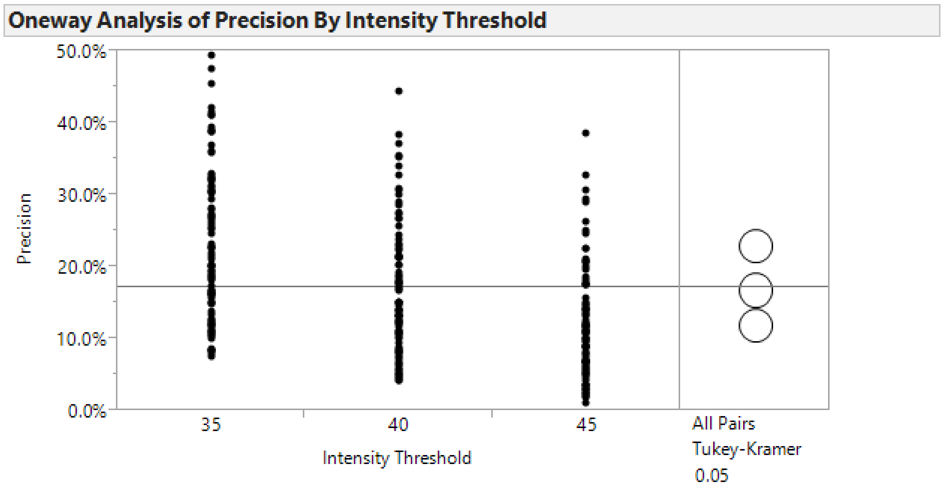


Figure 26. Comparison of Precision group means, with Tukey HSD comparison circles

Of the three parameters discussed, F-measure, as the harmonic mean of the other two factors, is the main value used for evaluating the effectiveness of the algorithm. Harmonic mean is a measurement of central tendency that is applicable when averaging rates ("Harmonic Mean Calculator, Formula & Calculation," n.d.). Figure 27 shows that the mean F-measure achieved by the algorithm is 28% considering all thresholds. This mean does not appear to include any outliers.

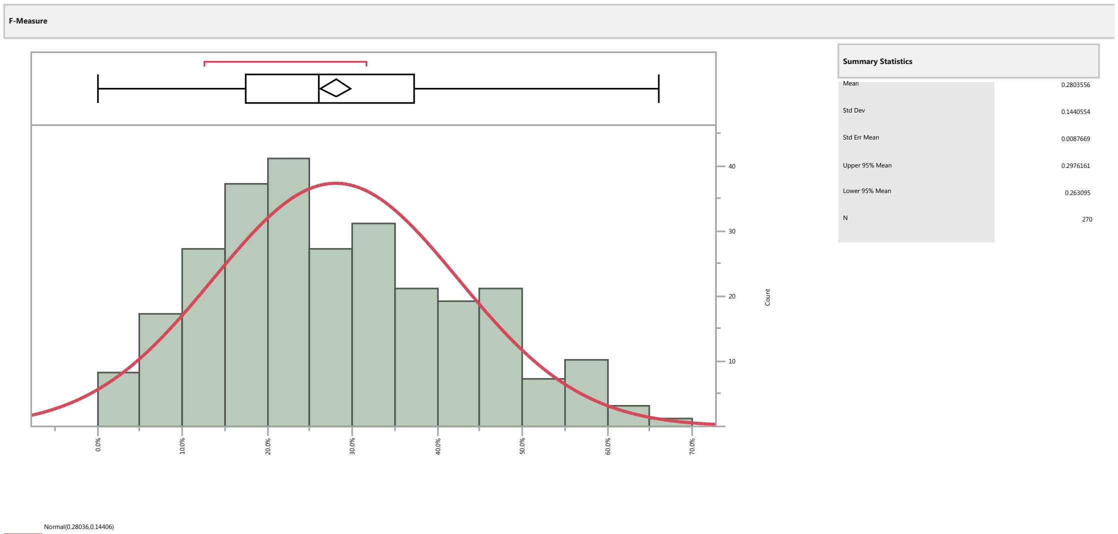


Figure 27. Overall F-measure summary statistics (all thresholds)

A breakdown of the observed F-measure at each intensity threshold shift is illustrated in Figure 28. To offer another perspective, a visual comparison of the group means using comparison circles for the All Pairs, Tukey HSD is illustrated in Figure 29. Intensity threshold -35 has the highest observed mean F-measure at 36.1%.

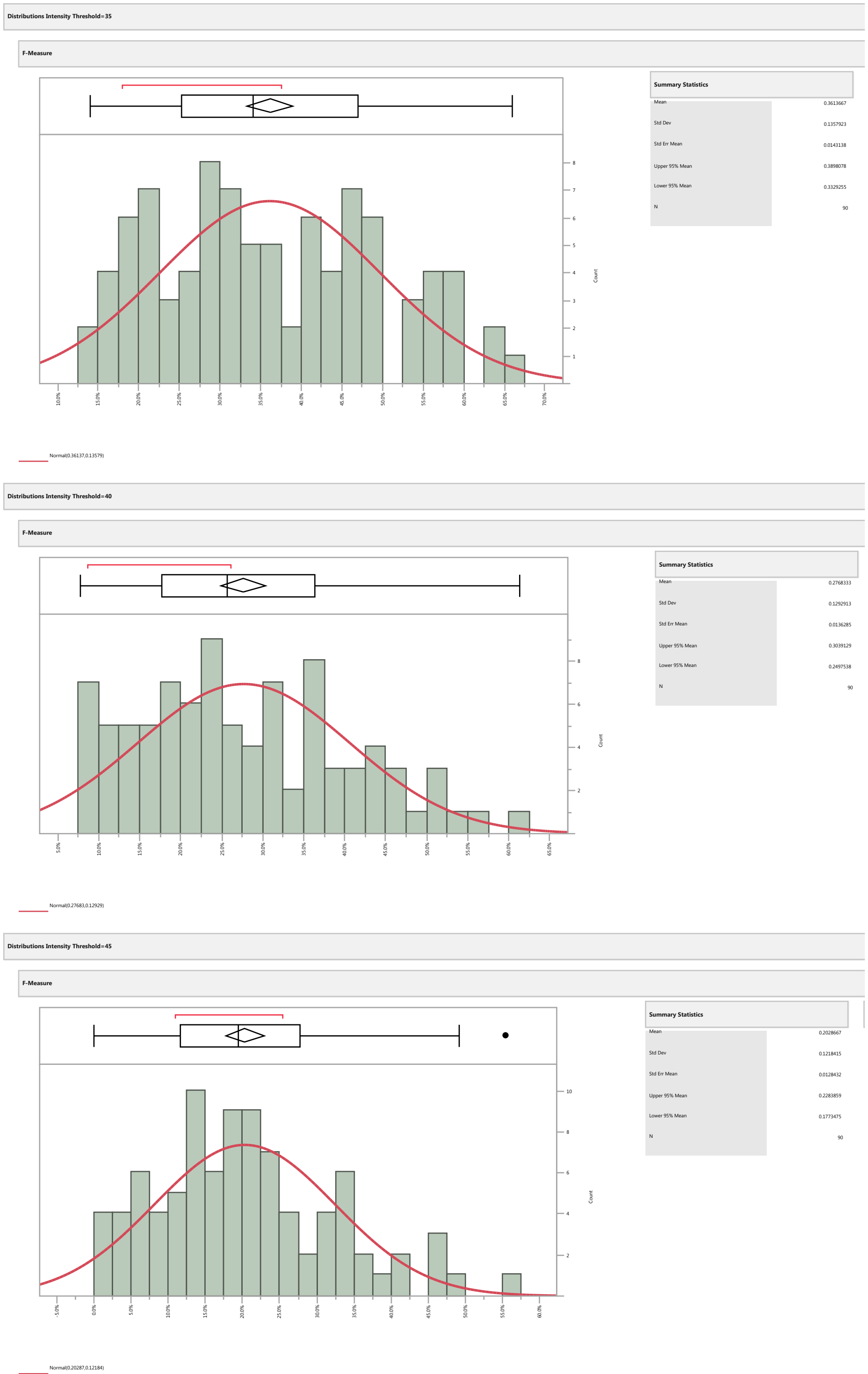


Figure 28. F-measure quantitative summary statistics by intensity threshold

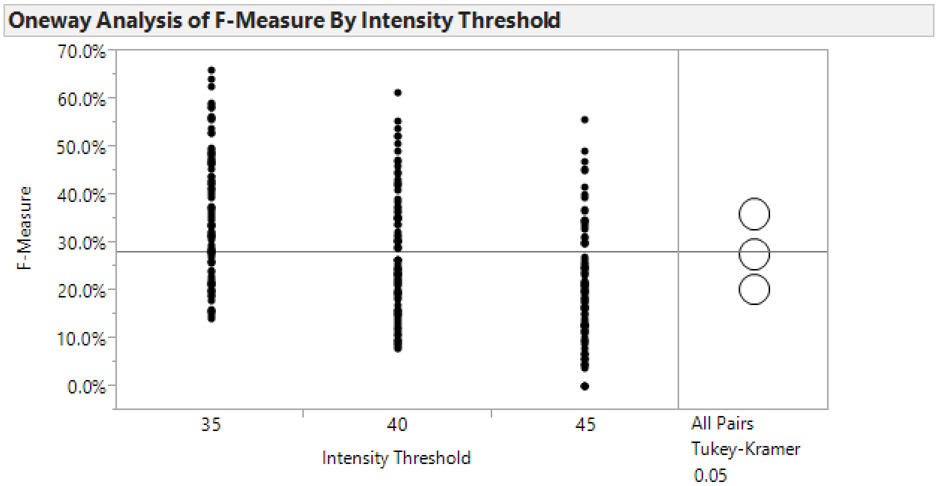


Figure 29 Comparison of F-measure group means, with Tukey HSD comparison circles

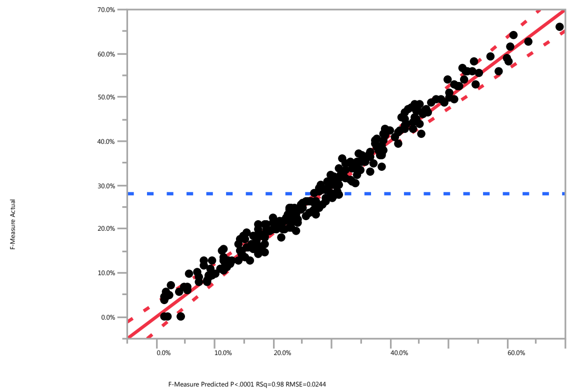
ANOVA test of F-measure

This research uses an analysis of variance (ANOVA) test to determine what factors explain the variance observed in the F-measure results. In this study, the only two factors under consideration are Image and Intensity Threshold. Image is the ordinal value for the image that distinguishes it from the rest of the images (i.e. File 1 through 90), where each image is non-identical and independent of the rest. Based on this fact, the F-measure was expected to vary greatly based on Image; however, accounting for this variance in the model was critical. Intensity Threshold is one of three threshold settings (i.e. -35, -40, or -45) used in the algorithm for image processing.

The results of the ANOVA test, found in Figure 30, show the overall F-test resulted in a p-value less than 0.0001. The null hypothesis, H_{o1}, is that none of the factors can explain the variance. Using an overall alpha value of 0.05, $\alpha_e = 0.05$, the ANOVA test successfully rejected the null hypothesis H_{o1}, showing that at least one of the factors can explain the variance in F-measure. Next, the effects tests of the two factors were analyzed using a comparison wise error rate, $\alpha_c = 0.025$. The effects tests analyze whether the difference of mean F-measure by each factor, Image or Intensity Threshold, are the same or statistically different. The null hypotheses H_{o2} and H_{o3} are that the means are the same in Image and Intensity Threshold, respectively. The effect tests of each factor, also found in Figure 30, show both p-values are less than 0.0001, far less than the α_c Therefore, both tests rejected H_{o2} and H_{o3}, showing that at least one of the Image or Intensity Threshold means are statistically different and affect F-measure variability.

Whole Model

Actual by Predicted Plot



Summary of Fit

RSquare	0.981088
RSquare Adj	0.971419
Root Mean Square Error	0.024354
Mean of Response	0.280356
Observations (or Sum Wgts)	270

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	91	5.4767053	0.060184	101.4702
Error	178	0.1055746	0.000593	Prob > F
C. Total	269	5.5822799		<.0001*

Effect Tests

Source	Nparm	DF	Sum of Squares	F Ratio	Prob > F
Image	89	89	4.3445292	82.3025	<.0001*
Intensity Threshold	2	2	1.1321761	954.4310	<.0001*

Residual by Predicted Plot

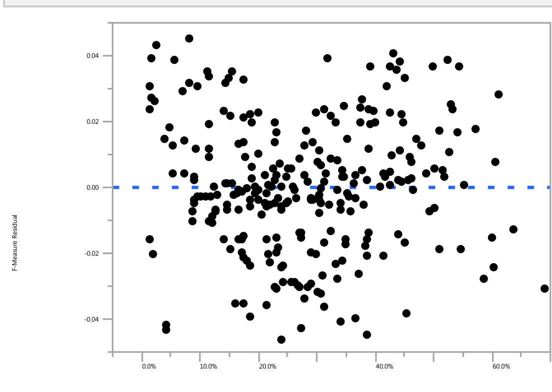


Figure 30. ANOVA results from JMP®

Figure 30 shows the R^2 value from the ANOVA. The R^2 value indicates that approximately 98% of the overall variance is explained by the factors of Image and Intensity Threshold. However, high R^2 values also indicate that the tests for assumptions of normality, independence, and constant variance will be more difficult to pass. The Durbin-Watson test, found in Table 9, resulted in a p-value of 0.6086 and successfully confirmed independence.

Table 9. Durbin-Watson independence test results

Durbin-Watson

Durbin-Watson	Number of Obs.	AutoCorrelation
0.7809958	270	0.6086

In Figure 31, the Shapiro-Wilk Goodness-of-Fit p-value is 0.0249, which is less than the α_c and therefore indicates the residuals are not normally distributed.

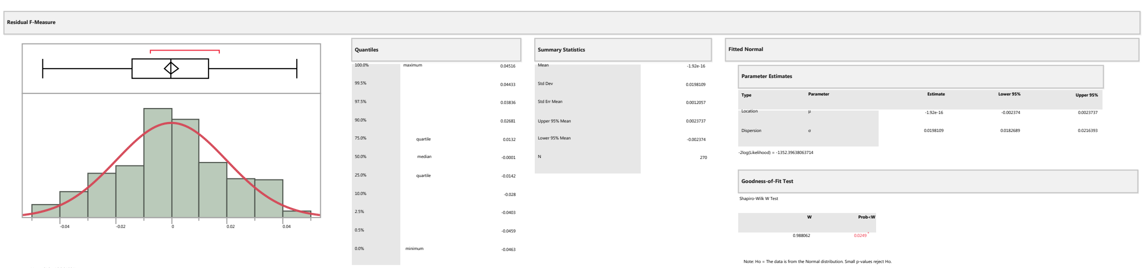


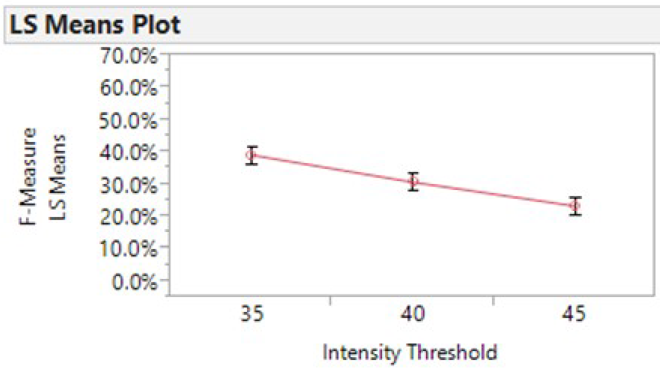
Figure 31. Histogram of residuals with Shapiro-Wilk W test results

Similarly, the Breusch-Pagan test of constant variance, found in Table 10, resulted in a p-value of 0.0000429 which is drastically below the α_c and confirms that the residuals do not have constant variance.

Table 10. Breusch-Pagan constant variance results

n	270	**observations**
df(exp)	91	**Model (ANOVA)**
SSE	0.1055746	**error Sum of Squares(ANOVA)**
SSR	0.00004704	**new Model (ANOVA from Res^2)**
TS	153.8317414	**(SSR/2)/(SSE/n)^2**
Pvalue =	4.2939E-05	**Chi dist**

Although the Shapiro-Wilk and Breusch-Pagan tests were unsuccessful, ANOVA is robust against deviations in normality and variance. Normality and constant variance is confirmed by the histogram of residuals (Figure 31) and residual by predicted plot (Figure 30), respectively. Therefore, this research concludes that the ANOVA is valid and that F-measure variability is attributed to the Image and Intensity Threshold factors. Upon further investigation of the effect test results, the LS Means Plot and Tukey HSD of Intensity Threshold, found in Figure 32, show that F-measures at each intensity threshold are statistically different. Additionally, Figure 32 shows that an intensity threshold shift of -35 is the most effective (i.e. highest F-measure) for this algorithm with a mean of 36%.



LSMeans Differences Tukey HSD

α= 0.050 Q= 2.36353

LSMean[i]	Mean[i]-Mean[j] / Std Err Dif / Lower CL Dif / Upper CL Dif	LSMean[j] 35	40	45
35	Mean[i]-Mean[j]	0	0.08453	0.1585
	Std Err Dif	0	0.00363	0.00363
	Lower CL Dif	0	0.07595	0.14992
	Upper CL Dif	0	0.09311	0.16708
40	Mean[i]-Mean[j]	-0.0845	0	0.07397
	Std Err Dif	0.00363	0	0.00363
	Lower CL Dif	-0.0931	0	0.06539
	Upper CL Dif	-0.076	0	0.08255
45	Mean[i]-Mean[j]	-0.1585	-0.074	0
	Std Err Dif	0.00363	0.00363	0
	Lower CL Dif	-0.1671	-0.0825	0
	Upper CL Dif	-0.1499	-0.0654	0

Level				Least Sq Mean
35	A			0.39201111
40		B		0.30747778
45			C	0.23351111

Levels not connected by same letter are significantly different

Figure 32. LS Means Plot and Tukey HSD results

Summary

This chapter presented the results of the field testing of the ASSETS prototype to collect imagery, and the subsequent SME validation of those images for to determine ground truth. Next, the qualitative results of the algorithm were summarized. A quantitative analysis of the Precision, Recall, and F-measure data was accomplished by interpreting the summary statistics and performing an ANOVA test to determine whether one of the factors (Image or Intensity Threshold) could explain the variance observed in the F-measure results. A complete dataset of Precision, Recall, and F-measure values is presented in Appendix C. Quantitative Data. The following chapter provides an interpretation of these results and a research conclusion.

V. Conclusions and Recommendations

Chapter Overview

This chapter interprets and discusses the results presented in this research. Specifically, this chapter draws conclusions from each analysis performed and highlights implications for USAF infrastructure asset management. This chapter also reviews and answers the research questions presented in Chapter I. Finally, suggestions for future research are detailed.

Conclusions of Research

This research has shown that it is conceptually possible to complete storm pipe condition assessments using a low-cost drone comprised of all COTS and open-source components. The photographic imagery collected was of sufficient quality and quantity that the algorithm could detect cracks autonomously with 36% efficiency. Efficiency, the F-measure, is the harmonic mean of Precision and Recall. Precision is the ratio of true positives assigned by the algorithm to total positives assigned by the algorithm (i.e. true positives + false positives), while Recall is the ratio of true positives assigned by the algorithm to the actual true positives possible identified in the ground truth. Evidence of an extremely high overall Recall (97.6%) and relatively low Precision (17.2%) indicates that the algorithm is detecting an excessive amount of false positives, instances where the algorithm detects a crack that the ground truth identifies is not present. Due to time constraints, no further analysis was possible regarding the low Precision. However, the images were taken very close to the pavement surface and it is very probable that normal surface features at this vantage point are mistaken for cracks in the algorithm as written. However, future research efforts to increase the Precision of the algorithm should consider more aggressive noise reduction and/or edge pruning techniques to reduce false positives resulting from enhanced surface features, uneven lighting, and debris present in the images.

The algorithm uses pixel thresholding to determine the surrounding intensity level of each pixel and determines an optimal thresholding value as the maximum intensity-difference in the image. As with the work performed in pavement crack detection, the algorithm did not perform successfully at the optimal thresholding value therefore intensity threshold shifting was required (Grandsaert, 2015). The ANOVA validates that intensity threshold shifts have a significant impact on F-measure response in this algorithm. For this system and application, intensity threshold -35 is the most effective threshold tested with this research. However, even with an observed mean of 36% (Figure 32), it is unlikely that the system as designed will be adopted for operational use. Infrastructure asset managers will likely desire increased F-Measure and Precision metrics. Still, this research does elucidate potential aspects of improving this technology to obtain more accurate crack detection data in the future.

With an R^2 of 98% the ANOVA accounts for almost all of the observed F-measure variance with only two independent variables – Image and F-measure. This research has shown that no additional input factors (e.g. asset age, construction material type) are required to control F-measure in the algorithm outputs. Because there is no need to collect other data, it is possible to detect cracks in photographic images using this algorithm at a minimal cost. However, other data inputs may be required if the research aperture were opened to include other types of defects. Also, the algorithm needs to be refined for better Precision (i.e. less false positives) to be operationally useful. This research draws from these conclusions to answer the investigative questions posed in Chapter I.

Investigative Questions Answered

This goal of this research was to prove that a low-cost autonomous system could quantify the condition of underground storm sewer pipes as good as or better than a CCTV inspection. The investigative questions posed in this study contribute to an effort focused on leveraging technology to autonomously detect condition defects in storm pipes. The results of this research and the answers to these questions can potentially aid USAF CE personnel in enterprise strategies for completing infrastructure AM condition assessments across the Air Force.

1. How can a small autonomous UGV be configured to collect pipe condition information?

A UGV used to collect pipe condition data is simply a means to an end, a tool for the specific purpose of collecting information. A system architecture can be used not only to create a drone by integrating a vehicle, applicable sensors, and algorithm technology towards data collection, but also to integrate the drone into the larger CE infrastructure asset management system. Literature shows that several sensor technologies can collect relevant sewer pipe condition information, including closed circuit television (CCTV) imagery, sewer scanning and evaluation technology (SSET), acoustic and sonar testing, infrared (IR) thermography, and ground penetrating radar (GPR). A UGV integrated with one or more of these sensors can collect data to quantify the condition of storm sewer pipes. This research demonstrates the use of an optical sensor for this application.

This research has proven that even a hobbyist radio controlled car can be used to collect pipe condition information by integrating it with a CCTV camera, LiDAR sensors, a computer processor, and a detection algorithm. Although this research was not able to leverage the autopilot hardware and software for autonomous navigation in the field, this proof of concept shows that a UGV can be manually controlled to collect pipe information in the field.

2. What field data and programming code is required to develop a data processing algorithm for pipeline fault detection?

Literature shows that there is a multitude of computer vision techniques and mathematical models tested to predict sewer condition. The field data required for pipeline fault detection is dependent on the type of evaluation performed. Additionally, Koo and Ariaratnam (2006) and Guo et al. (2009) both provide evidence that using multi-sensor platforms to collect the data increases detection accuracy and yields better evaluation results.

This research focuses on functional in-service evaluations, which speak to how effectively the asset performs its intended functions. This study uses photographic imagery collected in the field and a pavement crack detection algorithm developed in the Python programming language (see Appendix B. Programming Code) to detect cracks with 36% efficiency. This research shows that the algorithm, originally designed for crack detection in roadways, is also applicable for crack detection in storm sewer pipes.

3. How can the quality of pipeline fault detection data be quantified in order to inform decision-makers on pipe condition?

Once found, pipeline defects must be quantified into a logical representation of real-world pipe condition in order to be useful in AM decision-making. This can be accomplished in different ways, and depends on how the pipeline defects were found (i.e. mathematical modeling or computer vision techniques). This research used computer vision and metrics of Precision, Recall, and F-measure to evaluate the algorithm's success in detecting cracks. The algorithm calculated Precision, Recall, and F-measure by comparing the algorithm output image to its paired ground truth image and applying Equations (1) through (3), respectively. These factors were then statistically analyzed using analysis of variance (ANOVA) to show the significance of various intensity thresholds applied, but more importantly to identify under which scenario the algorithm performed best.

The results of this research show that with an F-measure of 36%, the algorithm is only partially successful in detecting cracks. The ANOVA successfully explained what factors affected the algorithm's effectiveness, but it did not quantify the condition into a logical value that would useful in AM decision-making. A more appropriate value would be a composite index, where the condition of a pipe section based on all relevant evaluation attributes (e.g. presence and severity of cracks, breaks, obstructions) is weighted and combined into a single value. This research is a proof-of-concept that the USAF could reengineer the AM inspection process to replace recurring contracts with government-owned and operated drones capable of classifying pipe defects into a quantifiable utility condition index (UCI).

Implications for USAF Infrastructure Asset Management

The existing USAF CE process architecture simply cannot provide the fundamental asset condition data at the speed necessary to effectively manage aging sewer assets across the world. By continuing to depend on contract support in a process architecture that undervalues in-service evaluations from the long-term funding plans, the USAF negatively impacts infrastructure performance and overall strategic success. Figure 33 is an example of how a condition decay curve could be used to develop preservation strategies based on an asset's remaining useful life and minimal acceptable performance level. It is critical to know where on the curve is the asset's current condition and what service life remains in order to make an informed decision. Because of sporadic funding support for contracted assessments, the USAF cannot accurately model the deterioration curve and has unknown risk in its storm sewer infrastructure.

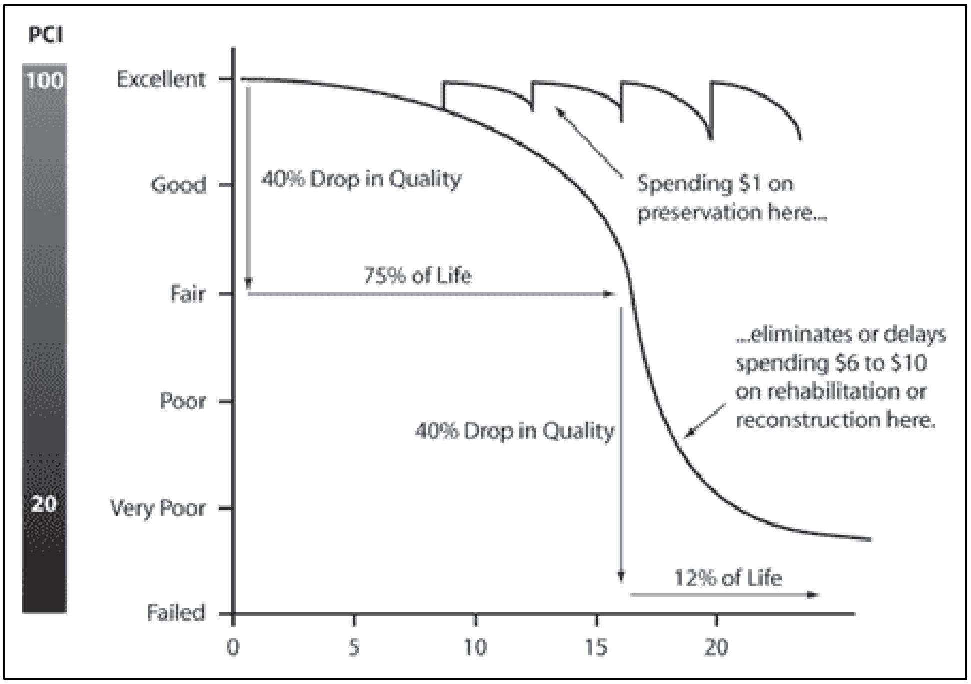


Figure 33. Example asset preservation strategy using condition modeling (Galehouse, Moulthrop, & Hicks, 2011)

The USAF needs organic capabilities to support reliable and timely in-service assessments of its storm sewer infrastructure. This research shows that a low-cost autonomous system can be developed using COTS hardware and open-source software to quantify the condition of underground storm sewer pipes. Additionally, it shows that 3D printing can be leveraged to exploit multi-sensor inputs during data collection. While the results show that the prototype developed for this research is not sufficient for operational use, it does demonstrate that the USAF can leverage COTS systems in future AM strategies. The significance in this concept is that the USAF could in essence reengineer the AM inspection process to replace recurring contracts with government-owned and operated drones capable of classifying pipe defects into a quantifiable utility condition index (UCI).

Recommendations for Future Research

This research has many outlets for future studies. Although this research could not achieve fully autonomous navigation, there is a good probability this can be resolved. The preliminary trials revealed that a drift error when the autopilot navigated the drone in place of manual controls. This drift error was too great to compensate for this research with gain settings in the Mission Planner® software. However, additional calibration with software settings or working with the open source community for a proven solution could correct the drift error. Once corrected, the ASSETS prototype could also be used to research other infrastructure systems (e.g. roads or airfield pavements where UAV flights may be impacted by real world operations).

Overcoming the challenge of autonomous navigation in the absence of GPS is a prerequisite for application of this technology in underground pipelines. Other researchers at AFIT and beyond have developed this capability; examples include Machin (2016) where UAVs navigate based on topographical landmarks, and Nassiraei et al. (2010) for sewer robot self-localization using passive arms and sensor inputs on the "naSIR mechanism" used in the KANTARO robot. Also, future research should consider improving the crack detection algorithm with robust image enhancement similar to that tested by Guo et al to eliminate non-uniform background noise (e.g. pipe joints, landmarks, and changes in lighting). Furthermore, additional sensors could be added to ASSETS for improved condition assessment. It is possible that using IR sensors and thermal post-processing analysis of the resulting imagery could provide the capability to detect subsurface anomalies such as impending root infiltration and soil loosening from leakage.

Another area of potential research is the development of a usable UCI based on an industry-recognized assessment standard such as the National Association of Sewer Service Companies (NASSCO) Pipeline Assessment Certification Program (PACP) condition rating methodology. This would also enable the development of an intelligent algorithm that could classify defects in pipes in accordance with the PACP ratings.

Finally, the effectiveness metrics used in this research could be improved by applying a probabilistic interpretation of Precision, Recall and F-score (Guotte & Gaussier, 2005). This probabilistic interpretation would potentially result in more accurate sample means or medians as well as better confidence estimates of Precision, Recall, and F-measure based on a probabilistic framework.

Recommendations for Action

With the recent accessibility and continued advancement of drone technology, there is a multitude of COTS options for ready-to-go systems that would drastically accelerate future research efforts in automating sewer pipe condition assessment. This research recommends that future research fund the purchase of a COTS ready-to-go system and target developing a detection algorithm using this system. For example, the RedZone Robotics® Solo™ pipe inspection robot (Figure 34) is an example of a fully autonomous CCTV system that can navigate and inspect 8-12” sewer pipes without a human operator. It does not, however, autonomously classify pipe defects and would be a good candidate for future research in that area as RedZone® is the developer of its own asset management software and may be willing to work with the USAF.

Figure 34. RedZone Robotics® Solo™ robot (RedZone-Robotics, n.d.)

A more comprehensive business case analysis should be performed prior to selecting a COTS system candidate for future research efforts. Systems such as the RedZone Robotics® Solo™ can inspect pipes at a much faster rate, approximately 190 meters per hour versus the CCTV system rate of 37 meters per hour (Nassiraei et al., 2007; RedZone-Robotics, n.d.). This estimate is based on the advantage that one operator can manage up to four RedZone Robotics® Solo™ robots simultaneously. The RedZone Robotics® Solo™ robot is commercially available for approximately $60,000 per unit, which is nearly twice the cost of the larger FiberScope.net® Pipe Crawler STORMER S3000 (FiberScope.net, 2015). However, this cost comparison does not account for the additional manpower required to operate the Pipe Crawler STORMER

S3000. By completing a thorough cost analysis of available COTS solutions, the USAF could target a viable system to use as a starting point for algorithm development. However, the goal of any future algorithm development should be to find an agile solution that would be compatible with other COTS data collection systems.

Summary

The fundamental objective of this research was to advance the USAF towards its goal of total asset visibility. Without a sustainable method of providing accurate, repeatable, and verifiable condition data for underground storm sewer pipes, the USAF CE community risks making uninformed decisions in a fiscally constrained environment.

This research conceptually shows that a low-cost autonomous system can be developed using COTS hardware and open-source software to quantify the condition of underground storm sewer pipes with an efficiency of 36%. Additionally, it shows that 3D printing can be leveraged to exploit multi-sensor inputs during AM data collection. While the results show that the prototype developed for this research may not be immediately adopted, it does demonstrate that the USAF can leverage COTS systems in future AM strategies to improve asset visibility at a significantly lower cost.

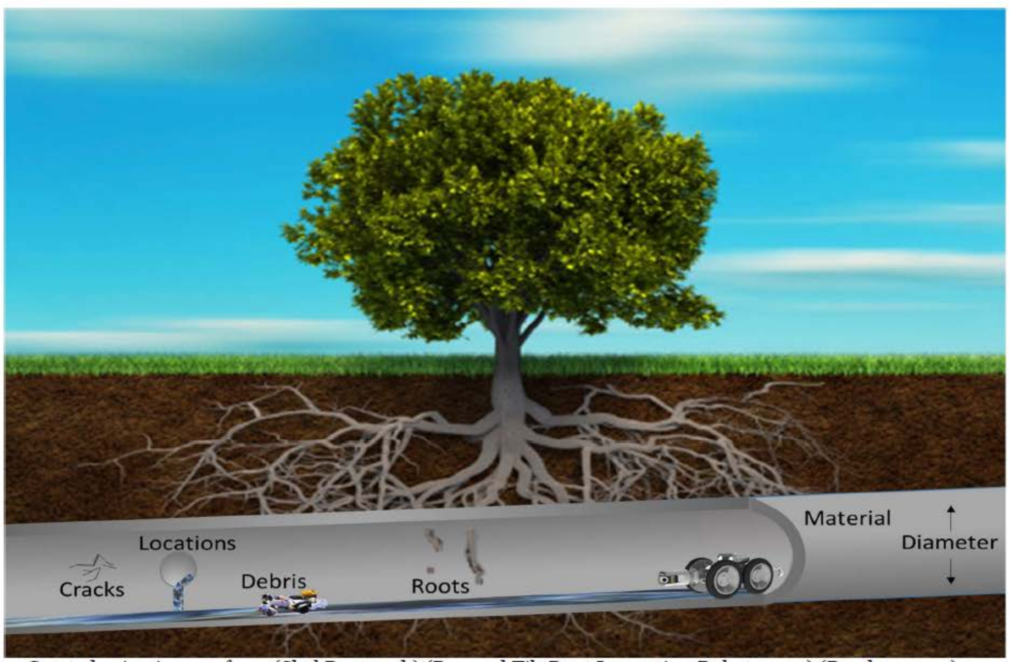
Appendix A. ASSETS System Architecture

Automated Storm Sewer Evaluation Technical System (ASSETS)

Full System Architecture

SENG 640, Spring 2015

<u>Group Members</u>:
Lt Erich Maxheimer
Lt Devin Menefee
Ms. Tracy Meeks



Created using images from (Shel Daat, n.d.) (Pan and Tilt Duct Inspection Robot, 2015) (Durden, 2014)

1

Contents

Project Proposal

1. **Project Title**: Automated Storm Sewer Evaluation Technical System (ASSETS)

2. **Problem Statement**: The United States Air Force (USAF) operates and maintains \$247B of infrastructure across 185 active, guard, and reserve installations globally (Deputy Under Secretary of Defense (Installations and Environment) (DUSD(I&E)), 2013). Part of this infrastructure includes an estimated 100,000 miles of storm sewer pipes (USAFE MAJCOM Asset Management Plan (MCAMP), 2013) which collect and drain excess surface runoff to avoid flooding and meet mission requirements. The accurate location and current condition of storm water pipes is critical information to effectively managing these assets. However, typically this data is very limited or even unknown because of the following storm water system characteristics: (1) pipes are often underground with limited access; (2) storm water collection systems have an expansive footprint across installations, sometimes exceeding 500 miles per installation; and (3) the pipe structure is a confined space in which to perform inspection. As a result of the lack of accurate information regarding storm water systems, expensive repairs to large sections of pipe are accomplished without ability to optimize repairs to only problem areas.

3. **Architectural Goal**: Inspection using ASSETS will be architected as a cost effective solution to assessing the condition of underground storm water pipes in the USAF inventory. The autonomous drone will be a self-contained system capable of detecting the presence and location of damages inside of storm sewer pipes, collecting asset information (location, diameter). The ASSETS determine current state and ultimately predict the future condition of the pipe. The ASSETS will be used by the Air Force Civil Engineering (USAF CE) community as a system-of-systems (SoS) within the asset management strategy adopted to effectively manage and sustain aging infrastructure assets across all AF bases.

4. **Scope**: An autonomous drone prototype will be developed by 2016 as part of a group member's AFIT thesis research, and implementation to the field is targeted, but not defined, to occur by 2020. This proposal will encompass the 5 year timeframe of prototype development and implementation of the ASSETS, and will make assumptions with respect to the integration into the USAF CE asset management SoS.

5. **Context**: The specific capability of the proposed architecture will be to accurately determine the state and predict the future condition of a storm water system and make more informed decisions for maintenance and repair investments. All base organizations, as invested stakeholders interested in functional infrastructure supporting mission requirements, are impacted by the potential capabilities of this architecture. However, Civil Engineering is the main AF mission area pertaining to the identified problem detailed in the Architectural Goal above and will be the primary organization involved. References relevant to this proposed architecture include:

    i. Executive Order 13327: Federal Real Property Asset Management (Feb 2004)
    ii. Executive Order 13423: Strengthening Federal Environmental, Energy, and Transportation Management (Jan 2007)

4

iii. Executive Order 13514: Federal Leadership in Environmental, Energy, and Economic Performance (Oct 2009)
iv. White House Memo: Disposing of Unneeded Federal Real Property (Jun 2010)
v. AFPD 32-10: Installations and Facilities (Mar 2010)
vi. AF/A7C Strategic Plan, Goal #3: Build Sustainable Installations

6. **Critical Questions**: As this is a complex and very specific problem with relatively original scope in the USAF CE context, there are many unknown elements which will likely affect the architectural design of the ASSETS. The critical questions identified at this time are listed below. As the research is preliminary, the priority order of critical questions is not known.

i. How much of the pipe infrastructure requires assessment to yield a condition of statistical significance?

ii. What type, quantity, and configuration of sensor technology is optimal for use in autonomous drone surveys of sanitary mains for the purpose of evaluating pipe condition?

iii. What field measurements, machine learning, and/or programming code is required to develop a data processing algorithm which will autonomously detect faults from background "noise"(e.g. pipe joints, manhole gaps) in sewer pipes?

iv. What low-cost vehicle option is optimal for use in an integrated autonomous drone?

v. How will the drone navigate existing pipe infrastructure? What pipe characteristics are required prior to deploying the drone for a successful inspection?

vi. What will be the user interface between the pipe measurements gathered from the drone and meaningful data to be used by USAF CEs?

7. **Team Experience**: Briefly describe each team member's experience highlighting factors that may be relevant to the problem and/or design solution.

i. Tracy Meeks: Graduate of Boise State University with a bachelor of science in Civil Engineering (BSCE). Approximately 10 years of USAF CE experience and has held positions at Mountain Home AFB, ID and Ramstein AB, Germany dealing with environmental engineering, pavement engineering, project management, asset optimization, and project programming. Her work as a civil engineer will lend experience in storm water systems, asset management principles and processes, and infrastructure maintenance and repair requirements.

ii. Lt Erich Maxheimer: Graduate of University of Illinois with a BSCE. Approximately 1 year active duty in the USAF as a developmental engineer (62E). He is a systems engineering major at AFIT, therefore has background in SE principles and processes that will be necessary in this architecture.

iii. Lt Devin Menefee: Graduate of The United States Air Force Academy with a bachelor of science in Mechanical Engineering (ME). Approximately 3 years of active duty in the USAF as a developmental engineer (62E) with positions in MQ-1/9 and KC-46, both at WPAFB, OH. Work as a combat engineer for the Air Forces largest "drone" program, he has experience with remotely piloted/ autonomous systems along with ground control stations/user interface.

Concept for Automated Storm Sewer Evaluation Technical System (ASSETS)

Executive Summary

Condition assessments for underground storm sewers typically are accomplished by closed circuit television (CCTV) systems, which consist of a transport vehicle deployable underground, camera system, human operator(s), and a supporting vehicle located aboveground. The CCTV assessment systems are expensive, ranging from $50K to $500K, and require a significant amount of human labor to both deploy the systems for data collection as well evaluating the data to determine condition. As a result, using these systems is not an organic capability within US Air Force (AF) civil engineering (CE) organizations.

The proposed ASSETS concept will fill this capability gap by the design of a cost-effective autonomous system to assess the condition of underground pipelines. The ASSETS will be able to operate inside a storm sewer pipeline with minimal human operator activity, take measurements for the accurate location and current condition of the pipes, and ultimately predict the condition of a pipe section with significance. A successful prototype is desired by 2016 with possible production and deployment by 2020.

Purpose

This architecture will enable the development of a cost-effective autonomous system to assess the condition of storm sewer pipes. The primary undertaking of the ASSETS is to accurately determine current state and predict the future condition of storm sewer pipes to facilitate decision-making for infrastructure investments, however a secondary capability of ASSETS will be to validate existing pipe characteristics in AF records (e.g. location, diameter).

Background

The United States Air Force (USAF) operates and maintains $247B of infrastructure across 185 active, guard, and reserve installations globally (Deputy Under Secretary of Defense (Installations and Environment) (DUSD(I&E)), 2013). Part of this infrastructure includes an estimated 100,000 miles of storm sewer pipes which collect and drain excess surface runoff to avoid flooding and meet mission requirements. The accurate location and current condition of storm water pipes is critical information to effectively managing these assets. However, typically this data is very limited or even unknown because of the following storm water system characteristics: (1) pipes are often underground with limited access; (2) storm water collection systems have an expansive footprint across installations, sometimes exceeding 500 miles per installation; and (3) the pipe structure is a confined space in which to perform inspection.

As a result of the lack of accurate information regarding storm water systems, expensive repairs to large sections of pipe are accomplished without ability to optimize repairs to only problem areas. As an example, a sanitary sewer repair project was executed at Mountain Home AFB, ID to replace a complete loop (approximately 5,000 linear feet) of pipe between 2007-2009. Because the pipe had the same typical age, no effort was taken to distinguish the condition of the pipe sections (smaller lengths of 25-50 linear feet) and the overall effort cost approximately $2M. If this concept were realized, the repairs could have been localized to the worst condition sections of sewer versus a complete replacement. Additionally, there were two additional phases of this project to other loops at Mountain Home AFB, ID with the same logic applied, and the total investment to this strategic sewer rehabilitation effort for Mountain Home AFB was closer to $7M.

Future Environment

The ASSETS will need to enable decision-makers at all levels of the AF to optimize application of resources and investments by providing targeted, informed, and predictive decision quality data – better known as asset management. This aligns to the *AF/A7C Strategic plan* (2011), Goal #3: Develop sustainable installations by implementing asset management principles for built and natural assets, which states:

"Asset visibility and performance data will allow Civil Engineers to leverage strategic sourcing for requirements needed across our portfolio (e.g., the annual chiller requirements we need across the Air Force can be strategically purchased, leveraging size for reduced costs, and delivery on demand). Total asset visibility will be implemented across all functional areas to account for every piece of the Air Force Civil Engineering enterprise."

Specifically, the following future environmental factors will have a role in this concept:

1. Changes in pipe network configurations and loads in response to future mission capabilities
2. The future implementation of standardized infrastructure assessment techniques and software (e.g. Sustainment Management Systems (SMS) and NextGen IT)
3. The continued deterioration of current infrastructure with time

Concept Time Frame/Scope

An autonomous drone prototype will be developed by 2016 as part of a group member's AFIT thesis research, and implementation to the field is targeted, but not defined, to occur by 2020.

Concept scope will encompass the 5 year timeframe of prototype development and implementation. Further, the ASSETS concept will be integrated into the USAF CE asset management timeline, which has not been formally identified.

Relevant references pertaining to scope and context of this architecture include:

- Unified Facilities Criteria (UFC) 3-201-01, Civil Engineering
- Civil Engineering enterprise architecture – NexGen IT or SMS
- Department of Defense Handbook (MIL-HDBK 1138), Wastewater Treatment System Operations and Maintenance Augmenting Handbook

Military Need Statement

As a result of the lack of accurate information regarding storm water systems, expensive repairs to larger sections of pipe are sometimes accomplished without consideration of optimizing repairs to only problem areas. Additionally, resources such as manpower to support preventative maintenance (e.g. snaking lines, clean outs, inspection) and corrective maintenance (e.g. spot repairs) are sometimes not optimized given the vast quantity of assets across an installation. The USAF CE community has a need for a low-cost, sustainable inspection system given the state of aging underground assets.

Central Idea

The autonomous drone will be a self-contained system capable of detecting the presence and location of damages inside of storm sewer pipes, collecting asset information (location, diameter), determining the current state and ultimately predicting the future condition of the pipe. It will require minimal human operators during deployment, and will work in conjunction with separate computing hardware and software for data analysis.

The autonomous drone could be a system comprised of a robotic vehicle integrated with sensors, hardware, controllers, and data storage and will be used to take measurements in the pipeline and record them. It will then transfer data to a separate computer with algorithms to evaluate inputs and determine the condition of the pipe.

Capabilities

The autonomous drone will take measurements inside of the pipe (e.g. presence/location of damages, configuration/location of pipe in 3D space, diameter of pipe) which will ultimately be used by an algorithm in a separate computer system to determine the current state and predict the future condition of the pipe segments. The predicted condition of the pipe segment will be quantified as a utility condition index (UCI), likely a range of 0 to 100. Once the algorithm is calculated, all data will be uploaded to an existing Sustainment Management System (SMS) for analysis and decision-making.

In general, the pipe measurements collected and the calculated UCI will be used to model the deterioration of the pipe network over time in the SMS. This information is critical and will be used to make decisions regarding maintenance and repair investments, operational performance measures, optimization of resources, and capacity calculations for new construction efforts.

Risks

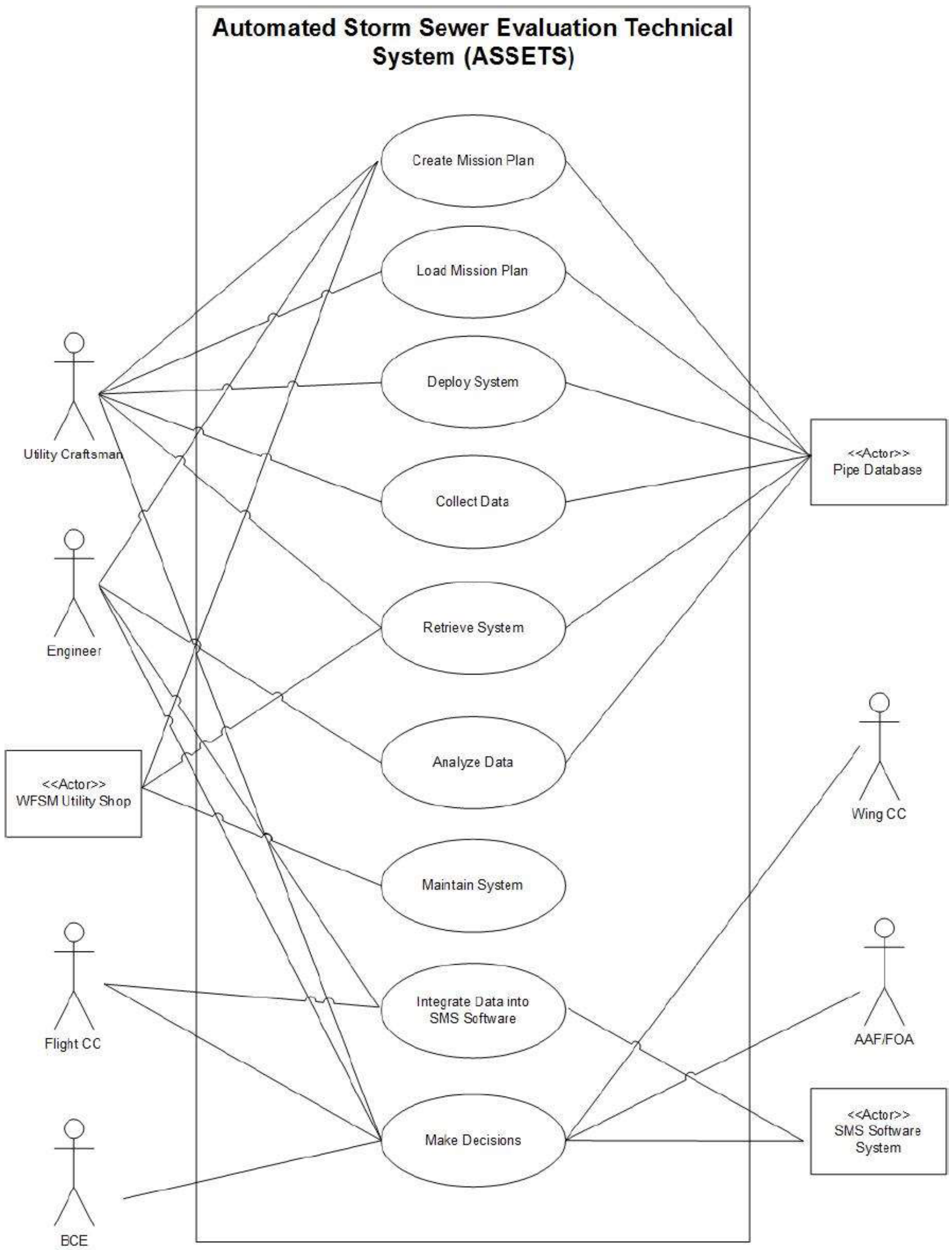
Risk to Mission: The major risk to the mission will be if the ASSETS were unable to be retrieved from the pipeline (e.g. due to malfunction) ultimately introducing an obstruction to storm water flow within the network. This scenario could cause an additional resource expense in order to retrieve it, but also a delay in effective storm water drainage in a localized area of the base. If this were to occur it could risk further damage to the pipe network. To mitigate this risk, the ASSETS should have a “backup retrieval” option if possible via either a tethered attachment to infrastructure or alternatively recovery by another ASSETS. There is always a risk of malfunction of any system, so this concept will allow for redundancy to critical subsystems where necessary.

Risk to Institution: Data storage requirements will be significant for the ASSETS, due mainly to the sheer size of the storm water pipe networks and potentially large file size depending on sensor type. The hardware, software, and human capital costs to support the data storage and management will be a risk to the force because it could drive additional secondary costs which are intangible in this concept architecture.

Summary

The USAF CE community has a need for a low-cost, sustainable inspection system given the state of aging underground assets. The ASSETS will be able to operate inside of a storm sewer pipeline with minimal human operator activity, take measurements for the accurate location and current condition of the pipes, and ultimately predict the condition of a pipe section with significance.

Use Cases



Automated Storm Sewer Evaluation Technical System (ASSETS)
Create Mission Plan
Load Mission Plan
Deploy System
Collect Data
Retrieve System
Analyze Data
Maintain System
Integrate Data into SMS Software
Make Decisions
Utility Craftsman
Engineer
<<Actor>> WFSM Utility Shop
Flight CC
BCE
<<Actor>> Pipe Database
Wing CC
AAF/FOA
<<Actor>> SMS Software System

11

Collect Data
Fully Dressed Use Case

1. Collect Data

1.1. Automated Storm Sewer Evaluation Technical System (ASSETS) Component – Drone (DRONE) travels through a storm sewer pipe and collects pipe measurements using sensors. This data is recorded to an onboard data storage device.

2. Actors Involved

2.1.1. Utility Craftsman (Primary)
2.1.2. Pipe Database (Supporting)

3. Flow of Events

3.1. BASIC PATH

3.1.1. Utility Craftsman initiates DRONE.
3.1.2. DRONE begins ongoing collection of pipe measurements through sensors.
3.1.3. DRONE moves at a preprogrammed rate towards next waypoint.
3.1.4. DRONE reaches waypoint.

If addition waypoints, go to 3.1.3

else continue to 3.1.5

3.1.5. DRONE waits at retrieval point.
3.1.6. DRONE sends "mission complete" message to Utility Craftsman.
3.1.7. DRONE goes into standby mode until retrieved.

3.2. ALTERNATE FLOWs

*a At any time, DRONE encounters significant obstruction in pipe.

1. DRONE sensors detect significant obstruction.
2. DRONE halts movement and goes into standby mode.
3. DRONE sends "obstruction" message to Utility Craftsman.
4. Utility Craftsman evaluates "obstruction" message.
5. Utility Craftsman manually maneuvers DRONE around obstruction.

   5a. Utility Craftsman commands DRONE to go to nearest safe retrieval point if obstruction is impassible.

6. Resume BASIC FLOW at 3.1.2

*b At any time, DRONE experiences subsystem malfunction.

1. DRONE detects subsystem malfunction.
2. DRONE halts movement and goes into standby mode.
3. DRONE sends "malfunction" message to Utility Craftsman.
4. Utility Craftsman evaluates "malfunction" message.
5. Utility Craftsman commands DRONE to continue mission if malfunction is fixable or minor.

5a. Utility Craftsman commands DRONE to go to nearest retrieval point if malfunction is significant and not fixable.

6. Resume BASIC FLOW at 3.1.2

3.3. EXCEPTION FLOWs

*a At any time, DRONE loses the capability to move.

1. DRONE sends "malfunction" message to Utility Craftsman if able.

Go to "Retrieve System" Use Case for an emergency retrieval sequence.

*b At any time, DRONE needs to be extracted before mission is complete.

1. Utility Craftsman overrides DRONE current mission.
2. Utility Craftsman commands DRONE to go to a nearby retrieval point.
3. Go to BASIC FLOW 3.1.3.

4. Special Requirements/Assumptions

4.1. DRONE shall only be deployed when storm sewer pipes are mostly dry (<1" depth of water).
4.2. Mission Plan shall include data for waypoints.
4.3. DRONE shall have minimal slippage on pipe surface during transit.
4.4. DRONE shall be operational in pipes having a diameter between 8" and 36".

5. Preconditions

5.1. DRONE is located in the pipe.
5.2. DRONE has mission plan loaded.
5.3. DRONE is turned on.
5.4. DRONE's subsystems are functioning and calibrated.
5.5. DRONE's is fully charged.
5.6. DRONE has full data storage capacity.

6. Postconditions

6.1. DRONE has recorded pipe measurements.
6.2. DRONE is waiting at retrieval point.
6.3. DRONE is in standby mode

7. Glossary of Terms

7.1. Underlined terms are defined in AV-2.

Overview and Summary Information (AV-1)

Architecture Project Identification	
Name	Automated Storm Sewer Evaluation Technical System (ASSETS)
Description	The primary undertaking of the ASSETS is to accurately determine current state and predict the future condition of storm sewer pipes to facilitate decision-making for infrastructure investments, however a secondary capability of ASSETS will be to validate existing pipe characteristics in AF records (e.g. location, diameter). ASSETS will be a system comprised of: (1) an autonomous drone integrated with sensors, hardware, controllers, and data storage; (2) a separate data analysis system with an algorithm to evaluate inputs and determine the condition of the pipe; and (3) a relay point between the drone and the data analysis system; and (4) and (5) the user interfaces, respectively.
Architects	Lt Erich Maxheimer, Lt Devin Menefee, Ms. Tracy Meeks
Organization	Air Force Institute of Technology
Assumptions and Constraints	The ASSETS architecture will be constrained by the following assumptions:
	- DRONE will only be deployed when storm sewer pipes are mostly dry (less than 1" depth of water).
	- DRONE navigation can occur without external input
	- DRONE shall have minimal slippage on pipe surface during transit.
	- DRONE shall be operational in pipes having a diameter between 8" and 36".
	- Mission Plan shall include data for waypoints.
	- Utility Craftsman and Engineer are trained in their respective user interfaces.
Approval Authority	LtCol Tom Ford
Date Completed	Final architecture to be released no later than 6 June 2015
Scope: Architecture View and Models Identification	
Views Developed	- CONOPS
	- Use Cases
	- Overview and Summary (AV-1)
	- High Level Operational Concept Graphic (OV-1)
	- Capability Taxonomy (CV-2)
	- Capability to Operational Activities Mapping (CV-6)
	- Operational Resource Flow Description (OV-2)
	- Operational Resource Flow Matrix (OV-3)

14

	- Organizational Activity Relationships Chart (OV-4)
	- Operational Activity Decomposition Tree (OV-5a)
	- Operational Activity Model (OV-5b)
	- Operational Rules Model (OV-6a)
	- State Transition Description (OV-6b)
	- Event Trace Description (OV-6c)
	- Logical Data Model (DIV-2)
	- Systems Interface Description (SV-1)
	- Systems Functionality Description (Sv-4)
	- Operational Activity to systems Function Traceability Matrix (Sv-5)
	- Integrated Dictionary (AV-2)
Capabilities	- Primary: 1. Accurately determine current state of storm sewer pipes. 2. Predict the future condition of storm sewer pipes 3. To facilitate decision-making for infrastructure investments
	- Secondary: Validate existing pipe characteristics in AF records (e.g. location, diameter)
Time Frames Addressed	A successful demonstration is desired by 2016 with possible implementation by 2020.
Organizations Involved	- Air Force Institute of Technology (system architect)
	- Air Force Civil Engineer Center (customer and SMS services)
	- Base Civil Engineering (system owner)
	Air Force Materiel Command (Acquisition and Sustainment)
Purpose and Viewpoint	
Purpose	Condition assessments for underground storm sewers typically are accomplished by closed circuit television (CCTV) systems, which consist of a transport vehicle deployable underground, camera system, human operator(s), and a supporting vehicle located aboveground. The CCTV assessment systems are expensive, ranging from $50K to $500K, and require a significant amount of human labor to both deploy the systems for data collection as well evaluating the data to determine condition. As a result, using these systems is not an organic capability within US Air Force (AF) civil engineering (CE) organizations. The proposed ASSETS concept will fill this capability gap by the design of a cost-effective autonomous system to assess the condition of underground pipelines. The ASSETS will be able to operate inside a storm sewer pipeline with minimal human

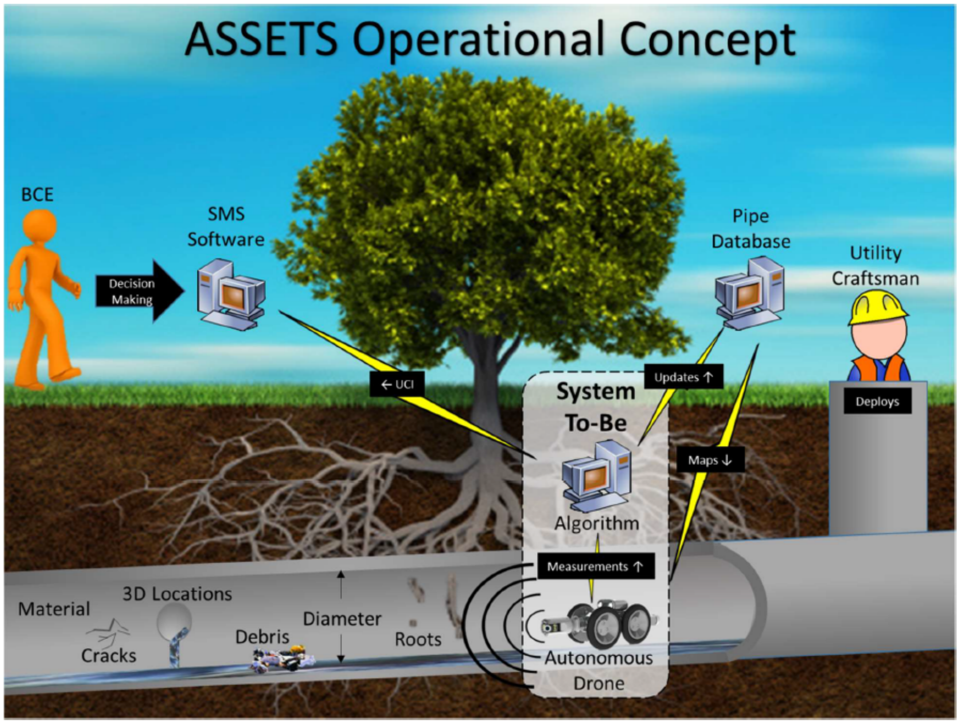
(Problems, Needs, Gaps)	operator activity, take measurements for the accurate location and current condition of the pipes, and ultimately predict the condition of a pipe section with significance. A successful prototype is desired by 2016 with possible production and deployment by 2020.
Questions to be Answered	1. How much of the pipe infrastructure requires assessment to yield a condition of statistical significance? 2. What type, quantity, and configuration of sensor technology is optimal for use in autonomous drone surveys of sanitary mains for the purpose of evaluating pipe condition? 3. What field measurements, machine learning, and/or programming code is required to develop a data processing algorithm which will autonomously detect faults from background "noise"(e.g. pipe joints, manhole gaps) in sewer pipes? 4. What low-cost vehicle option is optimal for use in an integrated autonomous drone? 5. How will the drone navigate existing pipe infrastructure? What pipe characteristics are required prior to deploying the drone for a successful inspection? 6. What will be the user interface between the pipe measurements taken from the drone and meaningful data to be used by USAF CEs?
Architecture Viewpoint	The ASSETS architecture will be developed from the perspective of USAF CE and its feasibility as a future infrastructure assessment tool. All viewpoints will have a focus intended to appeal to the USAF.
Context	
Mission	To accurately predict the condition of storm sewer pipe networks.
Doctrine, Goals, Vision	The ASSETS will need to enable decision-makers at all levels of the AF to optimize application of resources and investments by providing targeted, informed, and predictive decision quality data – better known as asset management. This aligns to the AF/A7C Strategic plan (2011), Goal #3: Develop sustainable installations by implementing asset management principles for built and natural assets.

<table>
<tr><td>Rules, Conventions, and Criteria</td><td>The ASSETS architectural data conforms to the DoD Architectural Framework (DoDAF) Version 2.0.

References relevant to this proposed architecture include:

i. Executive Order 13327: Federal Real Property Asset Management (Feb 2004)
ii. Executive Order 13423: Strengthening Federal Environmental, Energy, and Transportation Management (Jan 2007)
iii. Executive Order 13514: Federal Leadership in Environmental, Energy, and Economic Performance (Oct 2009)
iv. White House Memo: Disposing of Unneeded Federal Real Property (Jun 2010)
v. AFPD 32-10: Installations and Facilities (Mar 2010)
vi. AF/A7C Strategic Plan, Goal #3: Build Sustainable Installations</td></tr>
<tr><td>Linkages to Other Architectures</td><td>The ASSETS is linked to the SMS architectures.</td></tr>
<tr><td colspan="2">Tools and File Formats to be Used</td></tr>
<tr><td colspan="2">Sparx Enterprise Architect V10.0, Visio, Microsoft Word, Excel, & Powerpoint</td></tr>
<tr><td>Findings</td><td>See “Additional Insights” and “Critique” on pgs. 42 and 43.</td></tr>
</table>

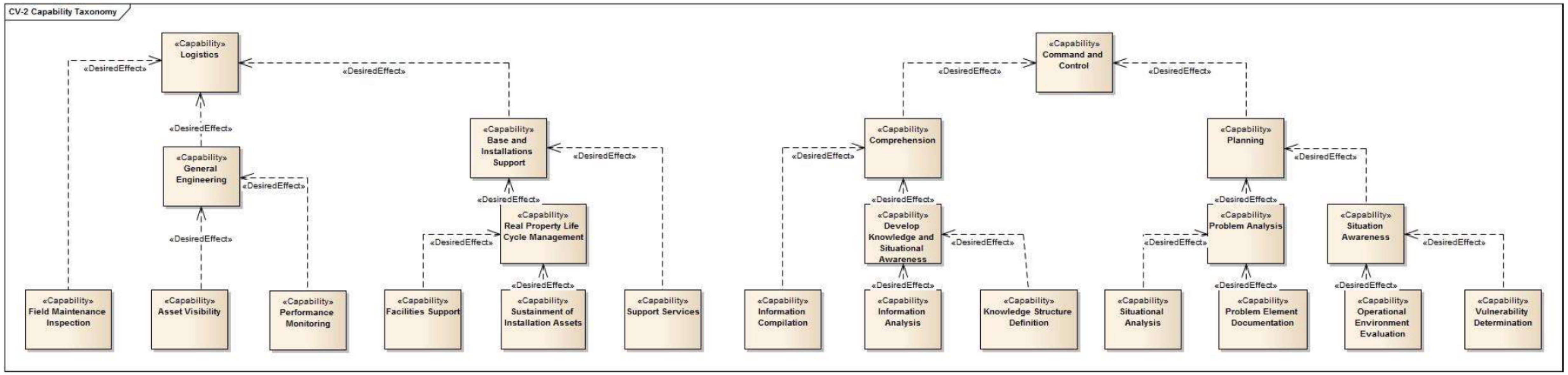
High Level Operational Concept Graphic (OV-1)



Created using images from (Shel Daat, n.d.) (Pan and Tilt Duct Inspection Robot, 2015) (Durden, 2014) (Clip Shrine, n.d.) (Gross, 2012) (Worker, 2015)

The image above is the Operational Concept for ASSETS. It is meant to serve as a graphical overview of the system and includes most of the components and relationships which interact with the system. To clarify any confusions, this paragraph will explain the image. Before being deployed, the ASSETS Autonomous Drone receives mapping information from a Pipe Database. The Utility Craftsman deploys the Drone through a manhole. While the Drone is in the storm sewer, it autonomously measures and detects different features. This inspection process is completely autonomous. The only time that the Drone needs human assistance is if it malfunctions, detects an obstruction which may be impassible, or needs to be removed from the storm sewer prematurely. In those situations, the Utility Craftsman would interact with the Drone to solve the problem. After a mission, the Utility Craftsman retrieves the Drone. The pipe measurements are transferred from the Drone to the Algorithm which is processed into a UCI. The Algorithm updates the Pipe Database and sends the UCI to the SMS Software. Ultimately, the BCE uses the SMS to make decisions.

Capability Taxonomy (CV-2)



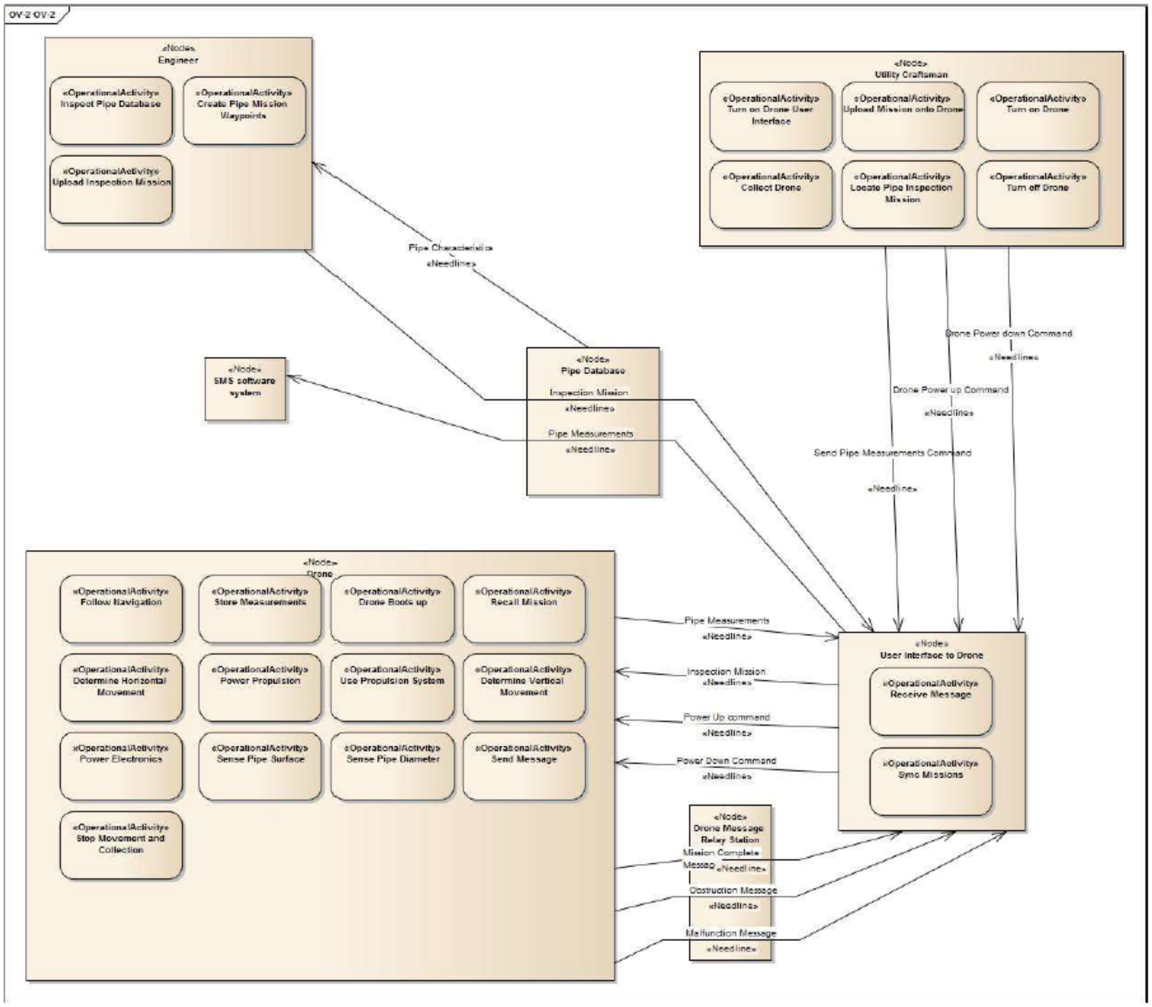
The CV-2 above depicts the high-level capabilities of the ASSETS and how they are broken down into more specific operational activities. The CV-2 is relatively balanced and shows approximately equal focus on both Command and Control and Logistics capabilities. We believe this is appropriate given that the complexity of ASSETS is matched equally by the abstract nature of the Logistics capabilities within civil engineering. In a fully dressed CV-2, there would be addition intermediates between some nodes. For example, a node for “Maintain” and “Field Maintenance” would be above “Field Maintenance Inspect.” “Engineering” would be above “General Engineering.” “Installation Services” would be above “Launch Services.” Finally, “Organize Information” would be above “Compile Information.” Each of these intermediates have many other sub-capabilities, besides ASSETS’s. These nodes were not included in this CV-2 because the goal was to only contain ASSETS capabilities. To do this some intermediates would only include one sub-capability of ASSETS. The CV-2 leafs (at the bottom of the diagram) are the operational activities that feed the Operational Activity Model (OV-5a).

Capability to Operational Activities Mapping (CV-6)

	Send Message	Receive Message	Use Propulsion System	Follow Navigation	Power Propulsion	Power Electronics	Determine Vertical Movement	Determine Horizontal Movement	Sense Pipe Diameter	Sense Pipe Surface	Recall Mission	Store Measurements
Field Maintenance Inspection	X	X	X	X	X	X	X	X	X	X	X	X
Asset Visibility							X	X	X	X		X
Performance Monitoring							X	X	X	X		X
Facilities Support							X	X				X
Sustainment of Installation Assets							X	X	X	X		X
Support Services												
Information Compilation												X
Information Analysis							X	X	X	X		X
Knowledge Structure Definition							X	X	X	X		X
Situational Analysis							X	X	X	X		X
Problem Element Documentation							X	X	X	X		X
Operation Environment Evaluation							X	X	X	X		X
Vulnerabilies Determination							X	X	X	X		X

The CV-6 identifies how operational activities in the OV-5a fulfill the capability elements in the CV-2. For this class, we have only performed an OV-5a on the capability element "Field Maintenance Inspect." For this reason, all of the operational activities map onto "Field Maintenance Inspect." "Determine Vertical Movement," "Determine Horizontal Movement," "Sense Pipe Diameter," "Sense Pipe Surface," and "Store Measurements" all map onto many capability elements. This indicates that they are critical activities to the success of the system. On the other hand, the other remaining operational activities do not map onto any capability elements besides "Field Maintenance Inspect." This proves that the specifics of how the system communicates and moves is not as important.

Operational Resource Flow Description (OV-2)



The diagram above is meant to show the "resource" flow, primarily of commands, within the ASSETS system during an inspection mission. The main nodes for this ASSETS OV-2 are the; Engineer, Utility Craftsman, Drone, and the User Interface to Drone. While the Drone itself contains the majority of the operational activities, the User Interface to Drone creates 4 needlines and receives 8, which is just one shy of the total. The amount of User Interface to the Drone needlines depicts how the ASSETS requires it for not only the communication between the main actors/nodes, but also for the medium in which the navigation, messages, and pipe measurements are exchanged. The Drone Message Relay Station ~~and the Base Pipe Network are~~ is used in this situation purely as a medium for the resource transition. The Pipe Database contains the information (Pipe Characteristics) the Engineer needs to create the Inspection Mission while the SMS software system receives the newly collected Pipe Measurements. SMS Software System and Pipe Database nodes currently do not contain any operation activities because we have not yet created an OV-5a for every leaf level capability in the CV-2. If we were to create an OV-5a for the "Information Analysis" capability, some of its activities would be included under the SMS Software System node. If we were to create an OV-5a for "Information Compilation" capability, some of its activities would be included under the Pipe Database node.

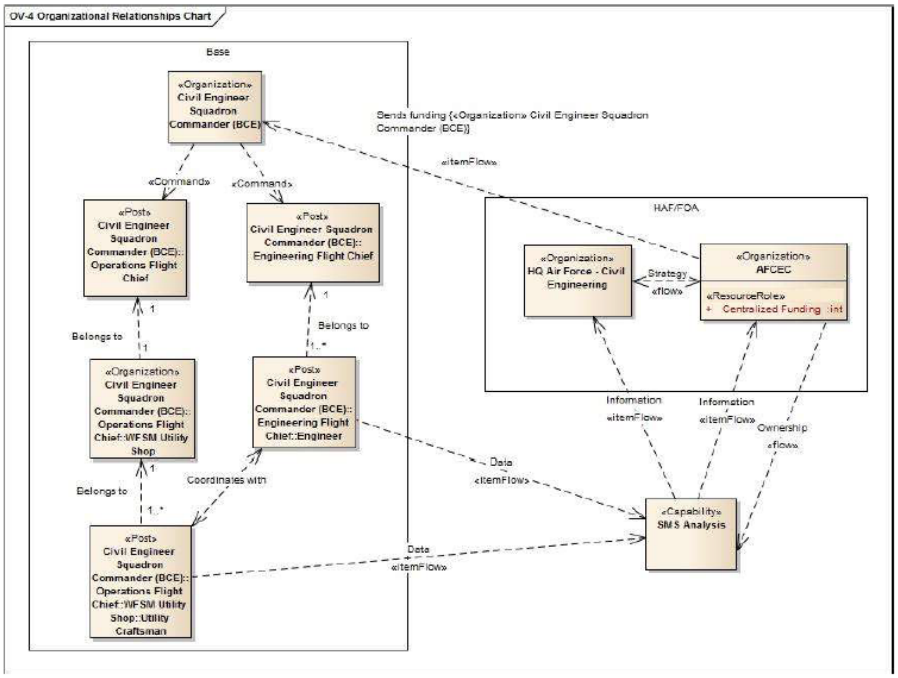
21

Operational Resource Flow Matrix (OV-3)

Connector_Name	Producer_Name	Consumer_Name
Pipe Measurements	Drone	User Interface to Drone
Mission Complete Message	Drone	User Interface to Drone
Obstruction Message	Drone	User Interface to Drone
Malfunction Message	Drone	User Interface to Drone
Inspection Mission	Engineer	User Interface to Drone
Pipe Characteristics	Pipe Database	Engineer
Inspection Mission	User Interface to Drone	Drone
Power Up Command	User Interface to Drone	Drone
Power Down Command	User Interface to Drone	Drone
Pipe Measurements	User Interface to Drone	SMS software system
Power Down Command	Utility Craftsman	User Interface to Drone
Power Up Command	Utility Craftsman	User Interface to Drone
Send Measurements Command	Utility Craftsman	User Interface to Drone

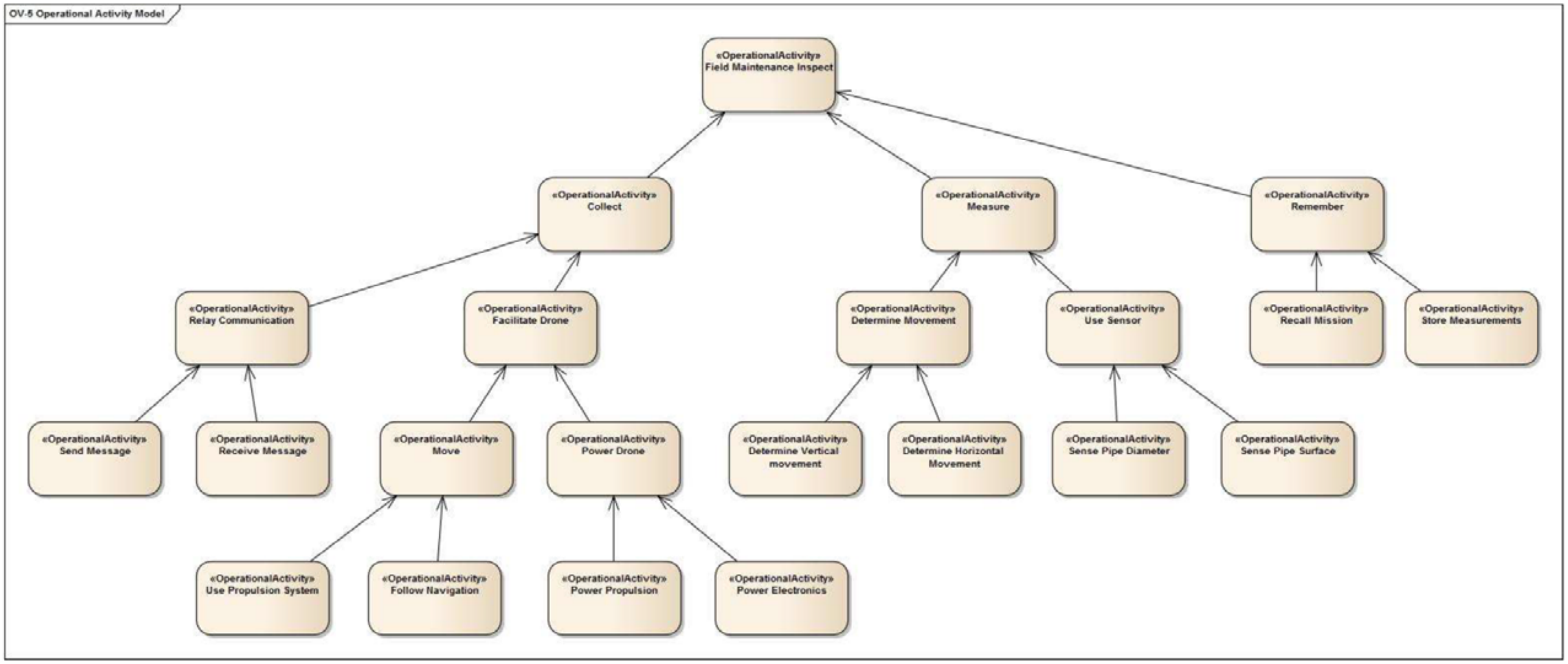
This chart is a simplified list of all the resources/products from the OV-2. Of note, 8 of the products are commands or messages which apply directly to the execution of the inspection mission. Also of note, the User Interface to Drone is either a Producer or a Consumer of all connectors except for the Pipe Characteristics.

Organizational Relationships Chart (OV-4)



The OV-4 above depicts the organizations having a part in the activities of using the ASSETS and implementing its capabilities. The OV-4 reinforces that the preponderance of the involvement is at the base level, which is the critical vantage point of effective asset management. There most knowledgeable people about the infrastructure are typically those base CE personnel that manage that infrastructure on a day-to-day basis. Additionally, the analysis of the data within SMS is critical for base asset management. However, funding for maintenance, repair, and construction of infrastructure has been centralized at the Air Force Civil Engineering Center (AFCEC), and policy and guidance continues to be issued from Headquarters Air Force (HAF). As a result, it is critical that information collected by ASSETS be transparent to the Headquarters Air Force (HAF) level for strategic policy implementation and AFCEC for investment decisions.

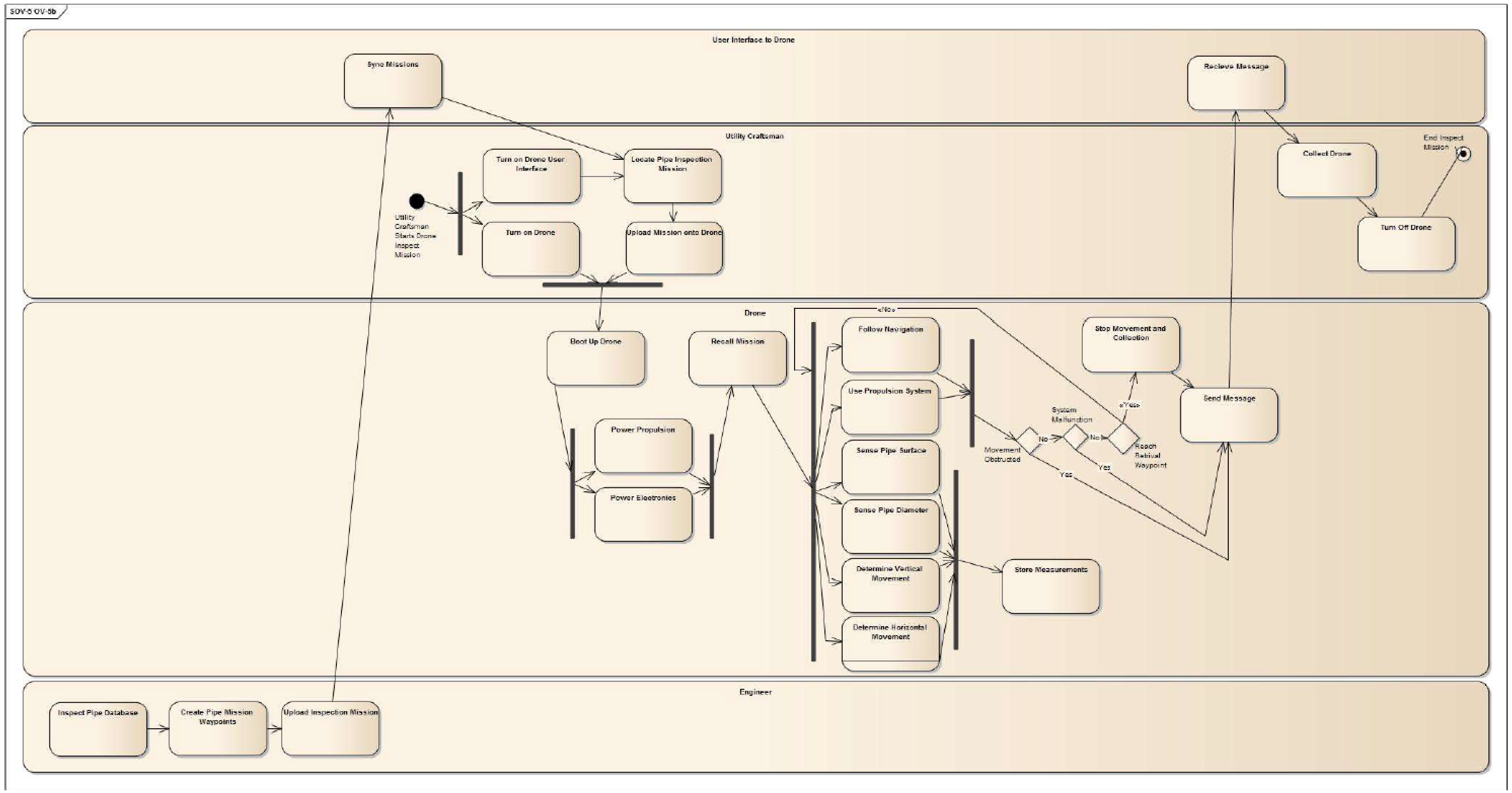
Operational Activity Decomposition Tree (OV-5a)



The OV-5a above depicts the "Field Maintenance Inspect" operational activity of the ASSETS. The OV-5a is a bit unbalanced with more definition (an additional level) under the "Facilitate Drone" activity. However, makes logical sense since this represents the critical thinking that has taken place with regards to the ASSETS Drone that is the focal part of a member's thesis at this time.

24

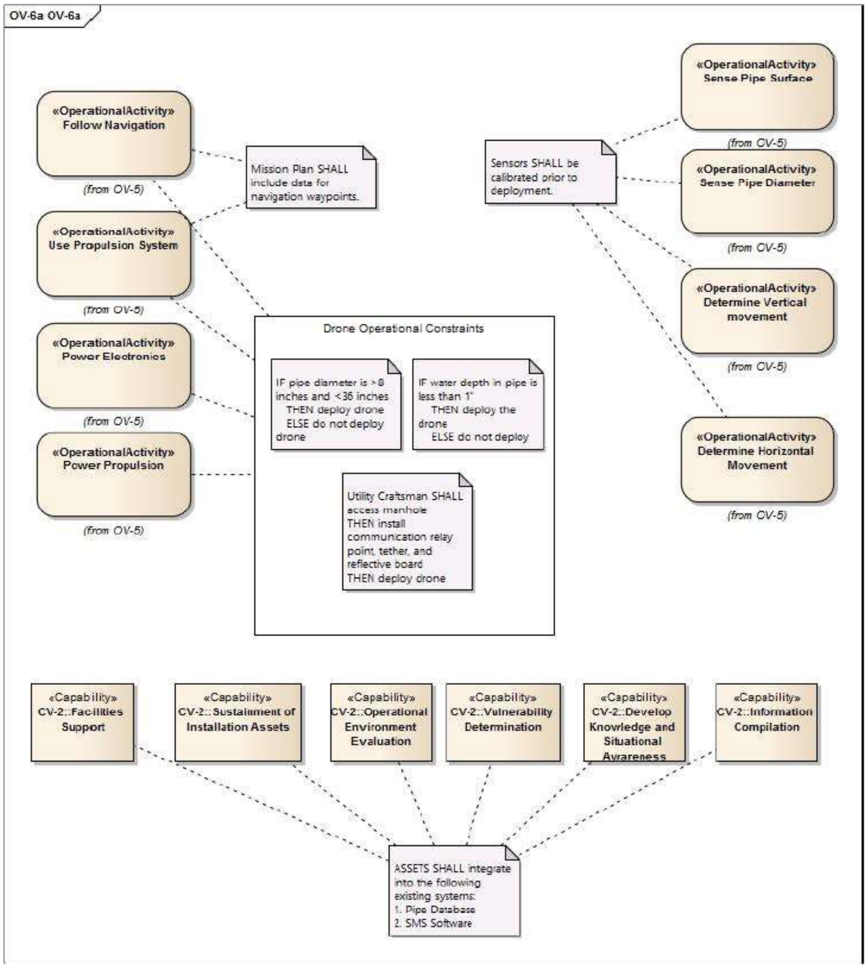
Operational Activity Model (OV-5b)



25

The activity diagram shown in the OV-5b is for the Field Maintenance Inspect activity. The Field Maintenance Inspection activity was decomposed in the OV-5a on page 23, however there is a lack of concordance as additional activities from the decomposition of other operational capabilities in the CV-2 were used besides those detailed on page 23. Before the activity even starts, the mission needs to be determined, to include the waypoint(s), by the engineer and uploaded and made available for the Utility Craftsman via the User Interface to Drone. From there, the Utility Craftsman can initialize the mission by; turning on the User Interface and drone, locating the mission specific for the pipes set to be inspected, and uploading them onto the drone. The drone autonomously runs the inspect mission by; recalling the mission, moving through the pipe, and collecting measurements. There will be continuous collection of measurements along with the continued check of if the drone has stopped movement for any reason or reached the retrieval waypoint. If either of these are true, the drone will send a corresponding message to the User Interface for the Drone. Then, the Utility Craftsman will collect and turn off the drone, thus ending the inspect mission.

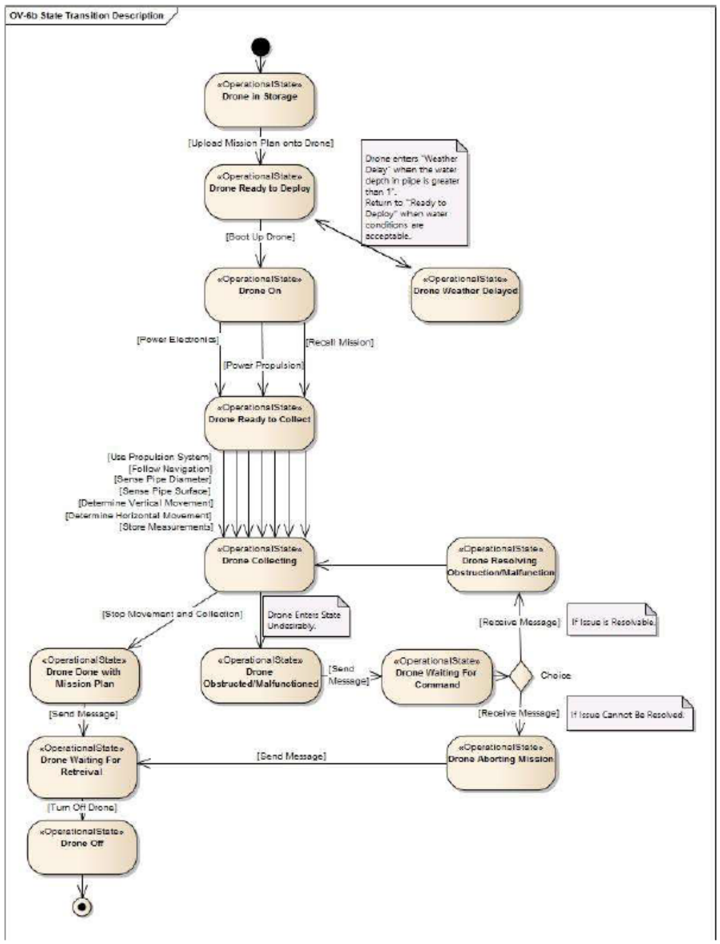
Operational Rules Model (OV-6a)



The OV-6a Rules Model above specifies operational constraints relating to what the ASSETS architecture will do. As has been evident in other views, the OV-6a is very focused on the ASSETS Drone since this represents the critical thinking that is the focal part of a member's thesis at this time. Most of the operational rules, five of six in the diagram, are mapped to operational activities from the OV-5a. However, the rule pertaining to integrating ASSETS into the existing systems was mapped to high-level capabilities from the CV-2. This was appropriate because the existing systems are currently being developed, and there are no specific details yet available on how ASSETS will tie into them; but the capability that integration is achieved is absolutely critical.

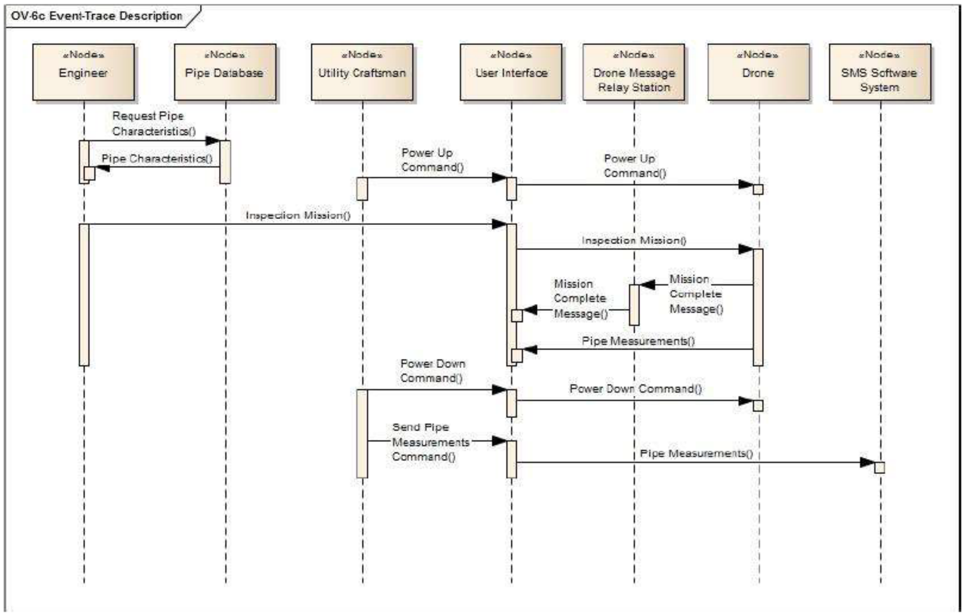
27

State Transition Description (OV-6b)



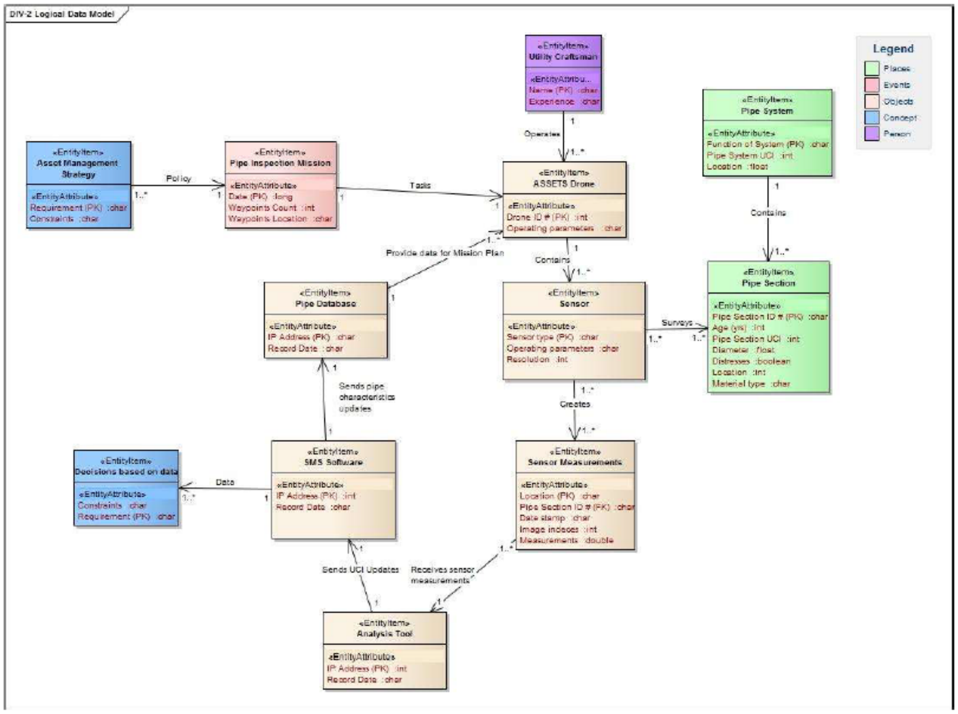
The above OV-6b is the State Transition Diagram for the ASSETS Drone. The initial state assumes that the Drone is in storage and the final state is when the Drone is turned off after a mission. In reality, there are other state which the Drone could be in, and other ASSETS comments which have their own states. For the scope of this class, we only evaluated the states of just the Drone during pipe inspection. The diagram uses activities from the OV-5a and OV-5b, or notes, as the events which trigger state transition. Note, the transition arrows in our model do not include solid triangles because of EA challenges. A choice node is used to show that there are two options when an issue occurs. The Utility Craftsman will use judgement to attempt to resolve the issue or abort the mission. This OV-6b represents a typical single use transition from storage to operation. In reality, the drone may not always start from storage.

Event Trace Description (OV-6c)



The above OV-6c is the Event-Trace Description for the overall, high level, mission scenario of assets. It uses nodes and messages from the OV-2 and is consistent with elements from the DIV-2. The diagram only represents the "happy case" and does not account for alternate scenarios. "Request Pipe Characteristics()" is included as the trigger for this diagram even though it is not include in other diagrams since it is assumed to be true.

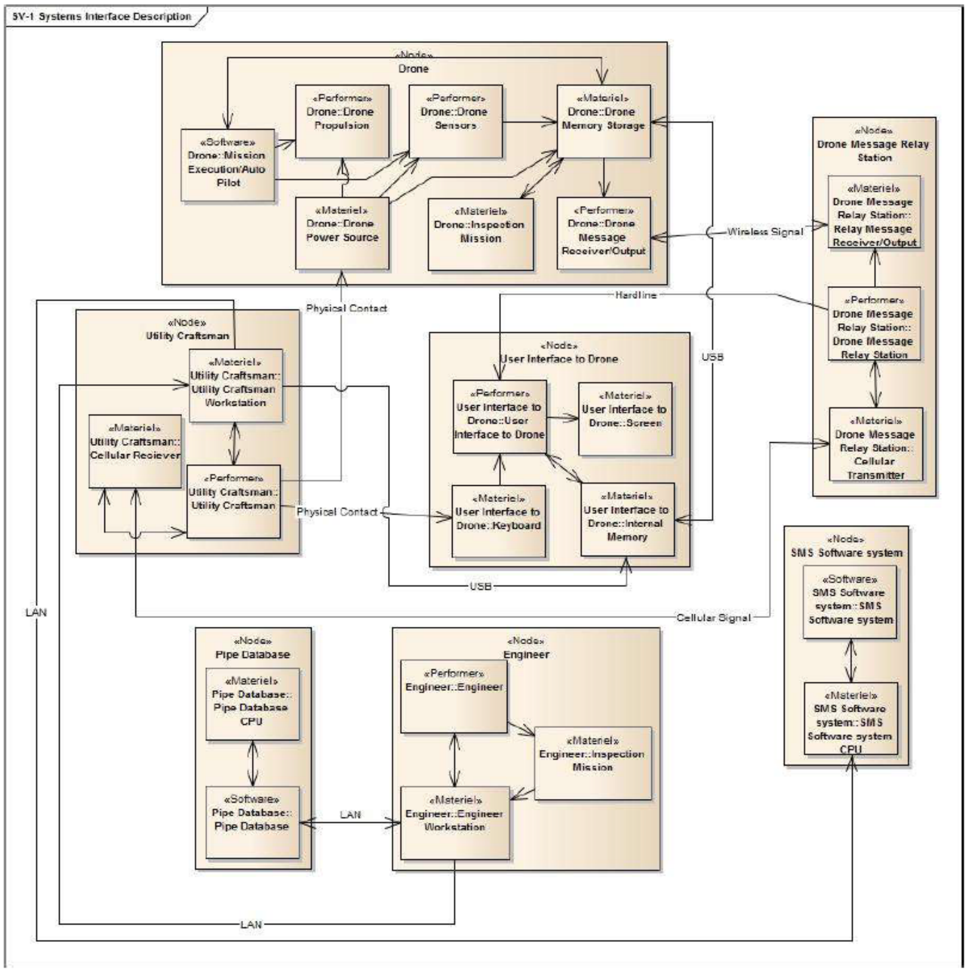
Logical Data Model (DIV-2)



The DIV-2 above is a visual representation of the data requirements and information transfers in ASSETS. This DIV-2 is not exhaustive to all data requirements (e.g. it does not include all person actors from the Use Case Diagram; interfaces are not detailed). However, it does include the most critical data requirements of the ASSETS operational concept. From the DIV-2 above it is clear that the "ASSETS Drone" is a key entity of the data model, as it has the most connections of any entity. It is the lynchpin for the operational capability of ASSETS, since the sensor and sensor measurements entities are contained within the Drone. There are very detailed attributes contained within the "Pipe Section" and "Sensor Measurements" because these data will be inputs into the algorithm within the "Analysis Tool" entity.

30

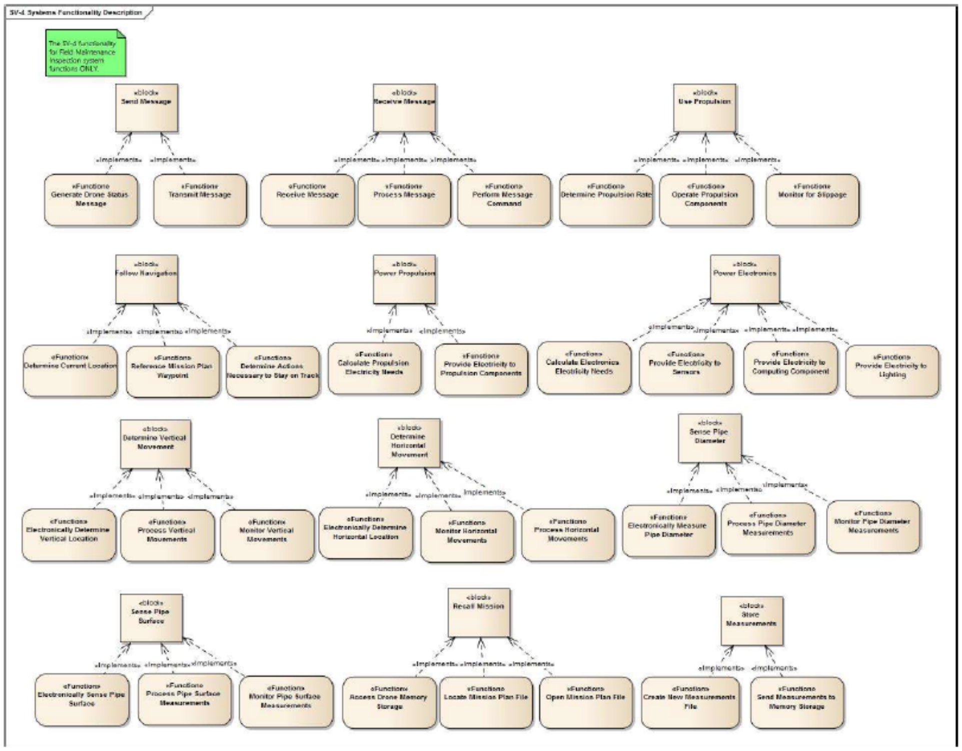
Systems Interface Description (SV-1)



The primary purpose of the above diagram is to show the methods and mediums of interaction between the ASSETS systems and actors. The process starts with the Engineer accessing the Pipe Database (which holds the known pipe characteristics) through his workstation and creating the inspection mission. From the Utility Craftsman's workstation, he can locate the Inspection Mission and download it onto the portable User Interface to Drone via USB. The Utility Craftsman can then plug in the Drone to the User Interface to Drone and upload an Inspection Mission prior to the mission; and after the mission download the collected Measurements using the same method. At the start of the mission, the Utility Craftsman can confirm everything is working correctly by connecting to a hardline run from the Drone Message Relay Station. During the mission, the Drone can communicate with the Utility Craftsman via a wireless signal sent through the Drone Message Relay Station and then sent from the Relay

Station to the Utility Craftsman through a cellular transmitter. After all inspection missions are completed, the Drone can send a mission complete message to the Utility Craftsman, notifying him to pick up the Drone and Relay Station. He then can upload the Pipe Measurements from the User Interface to Drone to the SMS Software system when he plugs the User Interface into his workstation.

Systems Functionality Description (SV-4)



The above SV-4 only provides the system functions under the “Field Maintenance Inspection” capability. Each of the leaf level activities from our OV-5a were included in the diagram as the higher level blocks. Since our system is not yet known, the diagram was created by starting with the OV-5a leaf level activities and determining system functions which would be needed for the activity to occur. Our OV-5a was very detailed so only one layer of system functions are included under each block.

Operational Activity to Systems Function Traceability Matrix (SV-5)

	SV-4::Access Drone Memory Storage	SV-4::Calculate Electronics Electricity Needs	SV-4::Calculate Propulsion Electricity Needs	SV-4::Create New Measurements File	SV-4::Determine Actions Necessary to Stay on Track	SV-4::Determine Current Location	SV-4::Determine Propulsion Rate	SV-4::Electronically Determine Horizontal Location	SV-4::Electronically Determine Vertical Location	SV-4::Electronically Measure Pipe Diameter	SV-4::Electronically Sense Pipe Surface	SV-4::Generate Drone Status Message	SV-4::Locate Mission Plan File	SV-4::Monitor for Slippage	SV-4::Monitor Horizontal Movements	SV-4::Monitor Pipe Diameter Measurements	SV-4::Monitor Pipe Surface Measurements	SV-4::Monitor Vertical Movements	SV-4::Open Mission Plan File	SV-4::Operate Propulsion Components	SV-4::Perform Message Command	SV-4::Process Horizontal Movements	SV-4::Process Message	SV-4::Process Pipe Diameter Measurements	SV-4::Process Pipe Surface Measurements	SV-4::Process Vertical Movements	SV-4::Provide Electricity to Computing Component	SV-4::Provide Electricity to Lighting	SV-4::Provide Electricity to Propulsion Components	SV-4::Provide Electricity to Sensors	SV-4::Receive Message	SV-4::Reference Mission Plan Waypoint	SV-4::Send Measurements to Memory Storage	SV-4::Transmit Message
OV-5::Determine Horizontal Movement								X							X							X												
OV-5::Determine Vertical movement									X									X								X								
OV-5::Follow Navigation					X	X																										X		
OV-5::Power Electronics		X																									X	X		X				
OV-5::Power Propulsion			X																										X					
OV-5::Recall Mission	X												X						X															
OV-5::Receive Message																					X		X								X			
OV-5::Send Message												X																						X
OV-5::Sense Pipe Diameter										X						X								X										
OV-5::Sense Pipe Surface											X						X								X									
OV-5::Store Measurements				X																													X	
OV-5::Use Propulsion System							X							X						X														

The above SV-5 is balanced where there is no overlap or gaps between the operational activities (from the OV-5a) and functions (from the SV-4). This seems appropriate since ASSETS is a “to-be” system and the SV-4 was created by decomposing the OV-5a to functions – in essence, the level of abstraction on the two source views is the same. Because there is no overlap or gaps, at this time there is no resiliency or redundancy in the system design.

Systems Measures Matrix (SV-7)

Systems Measures Matrix (SV-7)

Architectual Element	Type	Measure	Baseline (specified range)	Objective	Target
SV-4::Receive Message	Function	The maximum distance a message can be received from Drone Message Relay Station to Drone	1,000 ft	3,000 ft	Communicate with ground station (drone interface) with minimum reach of 1,000 ft
SV-4::Transmit Message	Function	The maximum distance a message can be sent from Drone to Drone Message Relay Station	1,000 ft	3,000 ft	Communicate with ground station (drone interface) with minimum reach of 1,000 ft
SV-1::Drone Memory Storage	Subsystem	The data storage capacity of the drone	500 GB	>1 TB	Data storage for 4 hrs of sensor measurements
SV-4::Provide Electricity to Computing Compo	Function	The power/battery capacity available for propulsion	2,000 mAh	5,000 mAh	Power capacity for 4 hr field deployment
SV-4::Provide Electricity to Lighting	Function	The power/battery capacity available for propulsion	2,000 mAh	5,000 mAh	Power capacity for 4 hr field deployment
SV-4::Provide Electricity to Propulsion Compo	Function	The power/battery capacity available for propulsion	2,000 mAh	5,000 mAh	Power capacity for 4 hr field deployment
SV-4::Provide Electricity to Sensors	Function	The power/battery capacity available for propulsion	2,000 mAh	5,000 mAh	Power capacity for 4 hr field deployment
SV-4::Send Measurements to Memory Storag	Function	The speed at which sensor measurements can be converted to memory.	500 MHz	700 MHz	Processing capability to enable a speed of 30 ft/min
SV-4::Operate Propulsion Components	Function	The maximum horizontal velocity of the drone in the pipe	10 ft/min	30 ft/min	The greatest velocity allowable with accurate sensor measurements

The SV-7 model purpose is to identify and describe measures for evaluating systems within a described architecture. The SV-7 model above is limited in that it does not document all functions or ASSETS subsystems or components, rather only critical functions and subsystems were included. In reality, this SV-7 will grow in detail and size with the continued development of this architecture. Baselines, objectives, and targets were set based on assumptions and the anticipated constraints of ASSETS at this time, but will definitely change as the architecture is refined and ultimately when the system prototype is developed and tested.

System Technologies and Skills (SV-9)

Technology	Pipe Characteristics Validation	UCI Algorithm Calculation	Complex Autonomous Maneuvering	Drone Size
Description	A secondary capability of ASSETS is to validate existing pipe characteristics in Air Force records. This would include pipe types and diameters. The validation of the system as a whole needs to be done with sample measurements taken by the drone.	The UCI algorithm calculates a UCI index for large sections of pipe using measurements of the pipe and extrapolation. This concept is in practice for infrastructure like pavements, but has not be fully developed for pipes.	Sewer pipes have a variety of different junction types and sizes. An automated drone needs to be able to navigate any real world configuration.	The minimum pipe diameter for our system is 8 inches. Past prototypes with full measurement capabilies have been too large for to meet this requirement.
Justification	The statistical concepts have been applied in the past to storm sewer pipes with CCTV assessments.	This concept has been applied to other infrastructure types and past research has yielded algorithms with limited capabilites for storm sewer pipes. Further research is required to reach level 7, but this should be easily achieved given success in other fields.	Simplified prototypes have been developed to navigate many different configurations autonomously. The prototypes do not fully match the vision of ASSETs, yet they share the same basic conceptual design.	Drones with limited capabilies have achieved the 8 inch pipe target. Miniaturization of sensors, batteries, and computing devices supports the ability to achieve this goal with research.
Current TRL	7	6	7	3
Time to TRL 7	0	18 months	0	12 months

Technology Readiness Level (TRL) 7 represents a prototype at or near the planned operational system. This threshold defines the point where risk is minimized by operational testing of a prototype in an environment similar to what is expected. The SV-9 above depicts the current and future critical technologies that can be reasonably forecasted. The timeframes estimated to TRL 7 were estimated based on the timeframes documented in the concept of operations.

Integrated Dictionary (AV-2)

Element	Type	Definition
"Malfunction" Message	Capability	A type of message sent from ASSETS to human operators, meaning that ASSETS has lost its ability to return to any possible retrieval point.
"Mission complete" Message	Capability	A type of message sent from ASSETS to human operators, meaning that the pre-determined survey is complete and that the Drone is at the retreival point.
"Obstruction" Message	Capability	A type of message sent from ASSETS to human operators, meaning that there is a significant obstruction in the pipe where ASSETS is deployed
ASSETS	Entity Item	Automated Storm Sewer Evaluation Technical System - the system being architected.
ASSETS Component - Data Analysis System	Entity Item	ASSETS system compent - contains an mathematical algorithm that ultimately predict the condition of the pipe (ref. UCI).
ASSETS Component - Drone	Entity Item	The self-contained data collection system that would be capable of detecting the presence and location of damages inside of storm sewer pipes and collecting asset attribute data (location, diameter).
ASSETS Component - Relay Point	Entity Item	ASSETS system compent - The system to transfer data from the Drone to the Data Analysis System.
ASSETS Component - User interface to Data Analysis System	Entity Item	ASSETS system component - the medium for engineer to manipulate/work with data analysis system.
ASSETS Component - User interface to Drone	Entity Item	ASSETS system component - the medium for utility craftsman to manipulate/work with drone.
BCE	Actor	Base Civil Engineer
Condition	Entity Item	A quantified measure of the physical and functional integrity of the pipeline compared to its initial state when constructed and installed.
Data	Entity Item	measurements and statistics collected together for reference or analysis of the storm sewer pipe.
FOA	Actor	Field Operating Agency
HAF	Actor	Headquarters Air Force
Mission plan	Entity Item	Existing information to be uploaded to the ASSETS prior to deployment. Tentatively will include: 1. Existing pipe attribute data to be verified 2. Pre-determined route that the ASSETS will survey
Pipe characteristics	Actor (secondary) / Capability	Relevant data about the storm sewer pipe that will be used by ASSETS for navigation. Including (at this time): 1. Location in 3D space 2. Diameter 3. Surface features
Pipe database	Actor	Storage system for pipe characteristics.
Pipe measurements	Capability	Relevant data about the storm sewer pipe that will be collected/recorded during the evaluation. Including (at this time): 1. Location in 3D space 2. Diameter 3. Surface features
Retrieval point	Entity Item	A location that can be used to either deploy or retrieve the ASSETS from the storm water network. The most typical example is a manhole.
Significant obstruction	Entity Item	An obstruction that is large enough to impede the ASSETS from navigating through the pipe.
SMS Software System	Actor	Sustainment Management System, a software system used by Civil Engineering community to manage infrastructure assets. Examples: BUILDER, PAVER, Geographic Information Systems (GIS)
Standby mode	Capability	State where ASSETS is powered down to minimal level to conserve battery.
UCI	Capability	Utility Condition Index - part of the Data Analysis System component being architected. UCI is a quantified representation of the condition of the storm sewer pipes; it would be on a standardized scale (e.g. 0 to 100) and could be used by decision makers to assess the state of infrastructure for maintenance and repair investments.
Utility craftsman	Actor	A technically trained person from the WFSM shop.
Waypoint	Entity Item	The coordinates or spatial reference of a specific location.
WFSM	Post	Water Fuels System Maintenance shop.

Architecture Evaluation

OV-1
Purpose: It is meant to serve as a graphical overview of the system and includes most of the components and relationships which interact with the system.
Focus: Before being deployed, the ASSETS Autonomous Drone receives mapping information from a Pipe Database. The Utility Craftsman deploys the Drone through a manhole or similar opening. While the Drone is in the storm sewer, it autonomously measures and detects different features. The inspection process is completely autonomous. The only time that the Drone needs human assistance is if it malfunctions, detects an obstruction which may be impassible, or needs to be removed from the storm sewer. In those situations, the Utility Craftsman would interact with the Drone to solve the problem. After a mission, the Utility Craftsman retrieves the Drone. The pipe measurements are transferred from the Drone to the Algorithm which is processed into a UCI. The Algorithm updates the Pipe Database and sends the UCI to the SMS Software. Ultimately, the BCE uses the SMS to make decisions.
Limitations: The OV-1 does not go into significant detail due to the overall/big picture view of the diagram. In order to cover all of the ASSETS scope, specifics such as the malfunction actions listed above, are not included.

CV-2
Purpose: The CV-2 depicts the high-level capabilities of the ASSETS and how they are broken down into more specific operational activities.
Focus: The CV-2 is relatively balanced and shows approximately equal focus on both Command and Control and Logistics capabilities. We believe this is appropriate given that the complexity of ASSETS is matched equally by the abstract nature of the Logistics capabilities within civil engineering. The CV-2 leafs (at the bottom of the diagram) are the operational activities that feed the Operational Activity Model (OV-5a).
Limitations: In a fully dressed CV-2, there would be addition intermediates between some nodes. For example, a node for "Maintain" and "Field Maintenance" would be above "Field Maintenance Inspect." "Engineering" would be above "General Engineering." "Installation Services" would be above "Launch Services." Finally, "Organize Information" would be above "Compile Information." Each of these intermediates have many other sub-capabilities, besides ASSETS's. These nodes were not included in this CV-2 because the goal was to only contain ASSETS capabilities. To do this some intermediates would only include one sub-capability of ASSETS.

CV-6
Purpose: The CV-6 identifies how operational activities in the OV-5a fulfill the capability elements in the CV-2.
Focus: All the operational activities shown map onto "Field Maintenance Inspect." "Determine Vertical Movement," "Determine Horizontal Movement," "Sense Pipe Diameter," "Sense Pipe Surface," and "Store Measurements" all map onto many capability elements. This indicates that these are critical activities to the success of the system.
Limitations: We have only performed an OV-5a on the capability element "Field Maintenance Inspect." The other remaining operational activities do not map onto any capability elements besides "Field Maintenance Inspect," which proves the specifics of how the system communicates and moves is not as important.

OV-2

Purpose: The diagram is meant to show the "resource" flow, primarily of commands, within the ASSETS system during an inspection mission.

Focus: The main nodes for this ASSETS OV-2 are the; Engineer, Utility Craftsman, Drone, and the User Interface to Drone. While the Drone itself contains the majority of the operational activities, the User Interface to Drone creates 4 needlines and receives 8, which is just one shy of the total. The amount of User Interface to the Drone needlines depicts how the ASSETS requires it for not only the communication between the main actors/nodes, but also for the medium in which the navigation, messages, and pipe measurements are exchanged. The Drone Message Relay Station is used in this situation purely as a medium for the resource transition. The Pipe Database contains the information (Pipe Characteristics) the Engineer needs to create the Inspection Mission while the SMS software system receives the newly collected Pipe Measurements.

Limitations: SMS Software System and Pipe Database nodes currently do not contain any operation activities because we have not yet created an OV-5a for every leaf level capability in the CV-2. If we were to create an OV-5a for the "Information Analysis" capability, some of its activities would be included under the SMS Software System node. If we were to create an OV-5a for "Information Compilation" capability, some of its activities would be included under the Pipe Database node.

OV-3

Purpose: This chart is a simplified list of all the resources/products from the OV-2.

Focus: 8 of the products shown are commands or messages which apply directly to the execution of the inspection mission. Of note, the User Interface to Drone is either a Producer or a Consumer of all connectors except for the Pipe Characteristics.

Limitations: Similar to the OV-2, limitations in this diagram exist in the non-documented "Information Analysis" and "Information Compilation" capabilities.

OV-4

Purpose: The OV-4 depicts the organizations having a part in the activities of using the ASSETS and implementing its capabilities.

Focus: The OV-4 reinforces that the preponderance of the involvement is at the base level, which is the critical vantage point of effective asset management. There most knowledgeable people about the infrastructure are typically those base CE personnel that manage that infrastructure on a day-to-day basis. Additionally, the analysis of the data within SMS is critical for base asset management. However, funding for maintenance, repair, and construction of infrastructure has been centralized at the Air Force Civil Engineering Center (AFCEC), and policy and guidance continues to be issued from Headquarters Air Force (HAF). As a result, it is critical that information collected by ASSETS be transparent to the Headquarters Air Force (HAF) level for strategic policy implementation and AFCEC for investment decisions.

Limitation: For the goal of this diagram, there are minimal limitations. However the diagram does not fully depict how ASSETS will connect Air Force wide with implementation on multiple bases.

OV-5a

Purpose: The OV-5a depicts the decomposition of "Field Maintenance Inspect" operational activity of the ASSETS.

Focus: The diagram decomposes the operational activities directly associated with the ASSETS drone. The movement, measurement collection, and communication activities are shown.

Limitation: The OV-5a is a bit unbalanced with more definition (an additional level) under the "Facilitate Drone" activity. However, this makes logical sense since it represents the critical thinking that has taken place with regards to the ASSETS Drone that is the focal part of a member's thesis at this time. In addition, there are many other aspects of the ASSETS which is not encompassed in this OV-5a which would require significant more decomposition from additional top level activities.

OV-5b
Purpose: The activity diagram shown in the OV-5b is an Operation Activity Model of the Field Maintenance Inspect activity structured in a logical, left-to-right sequence of events.
Focus: Before the activity even starts, the mission needs to be determined, to include the waypoint(s), by the engineer and uploaded and made available for the Utility Craftsman via the User Interface to Drone. From there, the Utility Craftsman can initialize the mission by; turning on the User Interface and drone, locating the mission specific for the pipes set to be inspected, and uploading them onto the drone. The drone autonomously runs the inspect mission by; recalling the mission, moving through the pipe, and collecting measurements. There will be continuous collection of measurements along with the continued check of if the drone has stopped movement for any reason or reached the retrieval waypoint. If either of these are true, the drone will send a corresponding message to the User Interface for the Drone. Then, the Utility Craftsman will collect and turn off the drone, thus ending the inspect mission.
Limitation: The Field Maintenance Inspection activity was decomposed in the OV-5a on page 23, however there is a lack of concordance as additional activities from the decomposition of other operational capabilities in the CV-2 were used besides those detailed on page 23.

OV-6a
Purpose: The OV-6a Rules Model specifies operational constraints relating to what the ASSETS architecture will do.
Focus: Most of the operational rules, five of six in the diagram, are mapped to operational activities from the OV-5a. However, the rule pertaining to integrating ASSETS into the existing systems was mapped to high-level capabilities from the CV-2. This was appropriate because the existing systems are currently being developed, and there are no specific details yet available on how ASSETS will tie into them; but the capability that integration is achieved is absolutely critical.
Limitations: The OV-6a is very focused on the ASSETS Drone since this represents the critical thinking that is the focal part of a member's thesis at this time.

OV-6b
Purpose: The OV-6b is the State Transition Diagram for the ASSETS Drone which depicts the different states of the drone and the logical sequence.
Focus: The initial state assumes that the Drone is in storage and the final state is when the Drone is turned off after a mission. In reality, there are other state which the Drone could be in, and other ASSETS comments which have their own states. For the scope of this class, we only evaluated the states of just the Drone during pipe inspection. The diagram uses activities from the OV-5a and OV-5b, or notes, as the events which trigger state transition. Note, the transition arrows in the model do not include solid triangles because of EA challenges. A choice node is used to show that there are two options when an issue occurs. The Utility Craftsman will use judgement to attempt to resolve the issue or abort the mission.
Limitations: This OV-6b represents a typical single use of the ASSETS Drone transition from storage to operation. In reality, the drone may not always start from storage. As implied, this diagram does not cover the entirety of the ASSETS.

OV-6c
Purpose: The OV-6c is the Event-Trace Description for the overall, high level, mission scenario of ASSETS to show how messages are sequenced between nodes.
Focus: Nodes and messages from the OV-2 are used and the diagram is consistent with elements from the DIV-2. "Request Pipe Characteristics()" is included as the trigger for this diagram even though it is not include in other diagrams since it is assumed to be true.
Limitations: The diagram only represents the "happy case" and does not account for alternate scenarios.

DIV-2
Purpose: The DIV-2 is a visual representation of the data requirements and information transfers in ASSETS.
Focus: The most critical data requirements of the ASSETS operational concept are included. From the DIV-2 it is clear the "ASSETS Drone" is a key entity of the data model, as it has the most connections of any entity. It is the lynchpin for the operational capability of ASSETS, since the sensor and sensor measurements entities are contained within the Drone. There are very detailed attributes contained within the "Pipe Section" and "Sensor Measurements" because these will be inputs into the algorithm within the "Analysis Tool" entity.
Limitations: This DIV-2 is not exhaustive to all data requirements (e.g. it does not include all person actors from the Use Case Diagram; interfaces are not detailed).

SV-1
Purpose: The primary purpose of the SV-1 is to show the methods and mediums of interaction between the ASSETS systems and actors.
Focus: The process starts with the Engineer accessing the Pipe Database (which holds the known pipe characteristics) through his workstation and creating the inspection mission. From the Utility Craftsman's workstation, he can locate the Inspection Mission and download it onto the portable User Interface to Drone via USB. The Utility Craftsman can then plug in the Drone to the User Interface to Drone and upload an Inspection Mission prior to the mission; and after the mission download the collected Measurements using the same method. At the start of the mission, the Utility Craftsman can confirm everything is working correctly by connecting to a hardline run from the Drone Message Relay Station. During the mission, the Drone can communicate with the Utility Craftsman via a wireless signal sent through the Drone Message Relay Station and then sent from the Relay Station to the Utility Craftsman through a cellular transmitter. After all inspection missions are completed, the Drone can send a mission complete message to the Utility Craftsman, notifying him to pick up the Drone and Relay Station. He then can upload the Pipe Measurements from the User Interface to Drone to the SMS Software system when he plugs the User Interface into his workstation.
Limitations: For the completion of the entirety of the ASSETS, many more connections would be made from both the Pipe Database and the SMS Software system. However, the extent of this diagrams depth is enough to portray a primary inspection mission.

SV-4
Purpose: The SV-4 provides the system functions under the "Field Maintenance Inspection" capability.
Focus: Each of the leaf level activities from the OV-5a were included in the diagram as the higher level blocks. Since the system is not yet known, the diagram was created by starting with the OV-5a leaf level activities and determining system functions which would be needed for the activity to occur.
Limitations: The OV-5a was very detailed so only one layer of system functions are included under each block.

SV-5

Purpose: The SV-5 traces all leaf-level activities from the ASSETS OV-5 to the SV-4 in a matrix.
Focus: The SV-5 is balanced where there is no overlap or gaps between the operational activities (from the OV-5a) and functions (from the SV-4). This seems appropriate since ASSETS is a "to-be" system and the SV-4 was created by decomposing the OV-5a to functions – in essence, the level of abstraction on the two source views is the same. Because there is no overlap or gaps, at this time there is no resiliency or redundancy in the system design.
Limitations: This diagram contains the same limitations from the OV-5a and the SV-5.

SV-7

Purpose: The SV-7 model purpose is to identify and describe measures for evaluating systems within a described architecture.
Focus: Baselines, objectives, and targets were set based on assumptions and the anticipated constraints of ASSETS at this time, but will definitely change as the architecture is refined and ultimately when the system prototype is developed and tested.
Limitations: The SV-7 model is limited in it does not document all functions or ASSETS subsystems and/or components, rather only critical functions and subsystems were included. This SV-7 would grow in detail and size with the continued development of this architecture.

SV-9

Purpose: The SV-9 depicts the current and future critical technologies that can be forecasted.
Focus: Technology Readiness Level (TRL) 7 represents a prototype at or near the planned operational system. This threshold defines the point where risk is minimized by operational testing of a prototype in an environment similar to what is expected. The timeframes estimated to TRL 7 were estimated based on the timeframes documented in the concept of operations.
Limitations: We cannot accurately predict changes in the future and the changes we predict are only 18 months out.

Additional Insights

There were many different difficulties when creating architecture for something unknown like ASSETS. The most reoccurring difficulty was one of scope. Throughout most DODAF view construction, group members had to work together to clarify "to what extent" and "how deep" each diagram would reach. The system itself is so massive with so many actors and branches, the group found the best approach to this first iteration of views was to primarily focus on what we saw as the foundational purpose of ASSETS; to collect pipe measurements. Because of this focus, many of the diagrams we created center around the inspection mission of the ASSETS Drone. There are diagrams which reach out past the inspection mission give a glimpse of; where the information from the mission will come from, where the collected information will go, and who all will be involved in the mission before, during, and after. However, the bulk of the detail is really centered on how ASSETS will gather data.

The OV-4 does a great job at showing all the many hands which touch and are effected by the ASSETS. It gives an excellent view of how "big" the system could be. However, diagrams like the OV-5b and the SV-1 provide an in depth look at how the pipe measurements are collected and the actual execution of ASSETS's root operation.

Two DODAF models which could be beneficial to the ASSETS architecture would be the CV-3 and the StdV-2. The Capability Phasing (CV-3) model is meant to show different phases of capability at different points in time. This could help the ASSETS architecture because the system could be modeled as first a data collection system, and then add data analysis, then and analysis implementation, and so on. The Standards Forecast (StdV-2) describes emerging standards and their potential impacts. ASSETS is set to integrate with a changing SMS system and an evolving Pipe Database, in addition to the always changing UCI standards.

Critique

ASSETS itself looks to have great potential. It is no mystery the DoD and the AF are trying to make the evermore dwindling budget stretch as far as possible. It is also apparent when you look at our KC-135s and our B-52s still in action that the AF likes to make our systems last. However those airframes have had continued maintenance, our facilities pipelines, not so much.

With that said, I completely encourage more work towards better maintenance of our facilities and base pipe networks. The thought of cutting down work force necessary to do this by use of drones is also something I agree with.

It is this next part of the architecture which I am yet to be sold on. The ASSETS has my full buy in on execution of these inspection missions (given the ASSETS Drone comes into existence with the promised capabilities), but the men and women make the AF. I need to see more of how ASSETS works with humans and how my people can make ASSETS a reality.

References

(n.d.). Retrieved April 16, 2015, from Clip Shrine: http://ar.clipshrine.com/%D8%AC%D9%87%D8%A7%D8%B2-%D9%83%D9%85%D8%A8%D9%8A%D9%88%D8%AA%D8%B1-12852-cv-a.html

Defense, U. D. (1982). *Wastewater Treatment System Operations and Maintenance Augmenting Handbook.* U.S. Department of Defense.

Defense, U. D. (2013). *Unified Facilities Criteria (UFC) 3-201-01, Civil Engineering.* June. Retrieved April 14, 2015, from https://www.wbdg.org/ccb/DOD/UFC/ufc_3_201_01.pdf

Deputy Under Secretary of Defense (Installations and Environment) (DUSD(I&E)). (2013). *FY 2013 Base Structure Report.* Washington, D.C.: Department of Defense.

Durden, T. (2014, August 10). *The Wonders Of The Modern World: China's Amazing Tower Of Garbage*. Retrieved April 16, 2015, from Zero Hedge: http://www.zerohedge.com/news/2014-08-10/wonders-modern-world-chinas-amazing-tower-garbage

Gross, M. (2012, August 18). *Three keys to turn the art of leadership in the art of leadership.* Retrieved April 16, 2015, from Imaginative Thinking: http://manuelgross.bligoo.com/20120818-tres-claves-para-convertir-la-tecnica-del-liderazgo-en-el-arte-del-liderazgo

HQ AF/A7C. (2011). *U.S. Air Force Civil Engineering Strategic Plan.* U.S. Air Force. Retrieved from https://app.eis.af.mil/a7cportal/SiteCollectionDocuments/2011%20U.S.%20Air%20Force%20Civil%20Engineering%20Strategic%20Plan.pdf

Pan and Tilt Duct Inspection Robot. (2015, April 16). Retrieved from Alibaba: http://www.alibaba.com/product-detail/Pan-and-Tilt-Duct-Inspection-Robot_60118925456.html

Shel Daat, T. (n.d.). *TLC.* Retrieved April 9, 2015, from Gnostic Students Network : https://plus.google.com/communities/102119423336749494689/stream/7b2e19a2-a91f-40ee-b648-9803550cc9be

USAFE MAJCOM Asset Management Plan (MCAMP). (2013, January). *Asset Details report.* Unpublished raw data.

Worker. (2015, March 1). Retrieved April 16, 2015, from Cloud Clipart: http://www.cloudclipart.com/clipart/free-clipart/worker-detail

Appendix B. Programming Code

ImageAcquisition.py

```
#Image acquisition script to take 5 sets of 10 images (50) consecutively with no delay

from pymba import *
import time
import cv2
import numpy as np
import matplotlib.pyplot as plt
import time
from ImgCap import ImageCapture
import random

Fltnum =1
Imgnum = 1
Setnum = 1
cnt = 1
while Setnum <= 5:
    cnt = 1
    while cnt <= 10:

        ImageCapture(Fltnum, Imgnum)
        time.sleep(0)
        Imgnum += 1
        cnt += 1
    #spacertime = random.randint(3,7)
    #time.sleep(spacertime)
    Setnum += 1
```

ImgCap.py

```
#ImgCap script - finds and uses camera via Vimba software, uses camera settings in Vimba during execution

from pymba import *
import time
import cv2
import numpy as np
import matplotlib.pyplot as plt
import time

def ImageCapture(Fltnum, Imgnum):
#start Vimba
    with Vimba() as vimba:
        #get system object
        system = vimba.getSystem()

        #list available cameras (after enabling discovery for GigE cameras)
```

```
        if system.GeVTLIsPresent:
            system.runFeatureCommand("GeVDiscoveryAllOnce")
    #         time.sleep(0.2)
        cameraIds = vimba.getCameraIds()
    #     print vimba.getCameraIds()
        #for cameraId in cameraIds:
            #print 'Camera ID:', cameraId

        #get and open a camera
        camera0 = vimba.getCamera(cameraIds[0])
        camera0.openCamera()

        #list camera features
        cameraFeatureNames = camera0.getFeatureNames()
        #for name in cameraFeatureNames:
            #print 'Camera feature:', name

        #get the value of a feature
        #print camera0.AcquisitionMode

        #set the value of a feature
    #     print 'camera Acquisition Mode:', camera0.AcquisitionMode
        camera0.AcquisitionMode = 'SingleFrame'

        #create new frames for teh camera
        frame0 = camera0.getFrame()    #creates a frame
        frame1 = camera0.getFrame()    #creates a second frame

        #announce frame
        frame0.announceFrame()

        #capture a camera image
        camera0.startCapture()
        frame0.queueFrameCapture()
        camera0.runFeatureCommand('AcquisitionStart')

        camera0.runFeatureCommand('AcquisitionStop')
        frame0.waitFrameCapture()
        dtime = time.strftime("h%Hm%Ms%S")

        #get image data...
        imgData = frame0.getBufferByteData()

        ##...or use NumPy for fast image display
        moreUsefullImgData = np.ndarray(buffer = frame0.getBufferByteData(),
                                        dtype = np.uint8,
                                        shape = (frame0.height,
                                                 frame0.width,
                                                 1))

        #clean up after capture
        camera0.endCapture()
        camera0.revokeAllFrames()
        #close camera
        #print moreUsefullImgData.shape
        imgRGB = cv2.cvtColor(moreUsefullImgData, cv2.COLOR_BAYER_RG2RGB)
        #print imgRGB.shape
        #img = cv2.cvtColor(imgRGB, cv2.COLOR_RGB2GRAY)
    #     print img.shape
```

```
    #     print moreUsefullImgData.shape
    #Save Image
        #plt.imshow(imgRGB, cmap = 'gray')
    #     plt.show()

        fname = '/Flight%03d Img%03d Time %s.png' %(Fltnum, Imgnum, str(dtime))
        print fname
        cv2.imwrite('C:/Test' +fname, imgRGB)

if __name__ =='__main__':
    Fltnum = 1
    Imgnum = 1
    ImageCapture(Fltnum, Imgnum)


```

SimpleImageConvert.py

```
from PIL import Image

img = Image.open('C:/Users/NUCANT/CrackTest/Roads2/Need to convert/90.png').convert('L')
img.save('C:/Users/NUCANT/CrackTest/Roads2/90.png')
```

Test1b.py

```
#import Thesis code files
import compare5xb
import plot21b

#import libraries to process images
from PIL import Image
import cv2

diff=40#the value the intensity is shifted.  This value is changed for each run ofthe script
num_imgs = 91
#Different File names for the different variations of the algorithm that were run
#name='-'+str(diff)+' histogram '
#name='-'+str(diff)+' histogramstatic '
name='-'+str(diff)+' run2normal '#the last file name for processing images from one flight

for x in range(69,num_imgs):
    print 'count number:', x
    #creates a red image for the algorithm image.  crack identification algorithm
    #is successful, the red image is overwritten.  if the algorithm runs into memory issues
    #the compare script skips processing the image
    im=Image.new("RGB",(1280,960),"red")
    im.save('C:/Users/NUCANT/CrackTest/algocrack/'+str(name)+ str(x)+'.jpg')

```

```
    plot21b.plot(name,x,diff)

#opens csv for each iteration.  The csv is evaluated in excel
f = open('effeciency'+str(name)+'.txt','a')
count=0
total=0


#compares each algorithm image produced with each ground truth image
for x in range(69,num_imgs):
    total+=1
    im3=cv2.imread("C:/Users/NUCANT/CrackTest/algocrack/"+str(name)+str(x)+".jpg",1)

    #if image is red, algorithm image was not created, so the image is skipped
    if im3[1&1].any()==[0&0&254]:
        continue

    f.write('File '+str(x)+'\t'+compare5xb.compare(x,name)+'\n')
    count+=1

#writes percentage of 30 images that were successfully processed
f.write('& comp\t'+"{:.1%}".format(float(count)/float(total))+'\n')

f.close()
```

Plot21b.py

```
#plot21b.py

#This Function takes the filename of the image to be processed as well as how much to shift
#the intensity level from what the algorithm calculates to be the brightest crack pixel

def plot(name,file,thresh):

    #Imports to be able to draw a node
    import networkx as nx
    import matplotlib.pyplot as plt
    import matplotlib.image as mpimg
    import matplotlib

    #Imports numberical tools and arrays
    import numpy as np
    from scipy import spatial

    #Imports computer vision tools
    import cv2

    #Tools to determine how long each step takes for operator awareness
    import sys
    import time

    #Image Loading Tools
    from PIL import Image
```

```

    #Load image as grayscale
    #Takes File from function call in order to load the correct image
    gray=cv2.imread('C:/Users/NUCANT/CrackTest/Roads2/'+str(file)+'.png',cv2.CV_LOAD_IMAGE_GRAYSCALE)

    print 'C:/Users/NUCANT/CrackTest/Roads2/'+str(file)+'.png'

    print "Loaded Image"

    #Loaded Image to show user it loaded correctly
    '''''
    namedWindow('dst_rt')
    #cv2.resizeWindow('dst_rt', window_width, window_height)

    imshow('dst_rt', gray)

    waitKey(0)
    destroyAllWindows()
    '''

    #Determines Image Dimensions,
    width,height = gray.shape
    width-=1
    height-=1
    gradient=[]
    pair=[]
    findthresh=[]
    threshold=0
    maxT=0
    intensities=[0]*256

    #Start Clock to Determine how long this step takes
    begin=time.time()

    gray[0,0]=0
    gray[0,height]=0
    gray[width,height]=0
    gray[0,height]=0
    # Ctrl Q to comment and uncomment blocks
    #Runs Algorithm to determine brightest crack pixel
    #loops through each pixel in loaded image and determines its brightness relative to its neighbors
    #Positive result indicates it is a darker pixel compared to its neighbors
    #each result is added to an array index from 0 to 255, the array index is determined by the intensity of the center pixel
    for j in range( 1, (height-1)):
        for i in range( 1, (width-1)):
            pixelintesity=0
            pixelintesity=pixelintesity+int(gray[(i-1),(j-1)]-gray[i,j])
            pixelintesity=pixelintesity+int(gray[i,(j-1)]-gray[i,j])
            pixelintesity=pixelintesity+int(gray[(i+1),(j-1)]-gray[i,j])
            pixelintesity=pixelintesity+int(gray[(i-1),j]-gray[i,j])
            pixelintesity=pixelintesity+int(gray[(i+1),j]-gray[i,j])
            pixelintesity=pixelintesity+int(gray[(i-1),(j+1)]-gray[i,j])
            pixelintesity=pixelintesity+int(gray[i,(j+1)]-gray[i,j])
            pixelintesity=pixelintesity+int(gray[(i+1),(j+1)]-gray[i,j])
            intensities[gray[i,j]]=intensities[gray[i,j]]+pixelintesity
        if j%100 == 0:
```

```
                print "......"+str(int(100*(float(j)/float(height))))+"%\r",

        #Ends Clock and prints time to determine how long step took
        end=time.time()
        print "step time: " + str(int(end-begin))+" seconds"

        #print intensities
        # Bar Chart
        #fig, ax = plt.subplots()

        n_groups=len(intensities)

        index=np.arange(256)

        bar_width=.1
        opacity=4
        error_config={'ecolor': '0.3'}

            #rects=plt.bar(index,intensities,bar_width)
            #plt.show()

            max=0

            #Finding brightest probable crack pixel by finding the intensity with the hi
ghest difference from its neighbors
            begin=time.time()
            i=0
            for i in range(256):
                if intensities[i]>max:
                    max=intensities[i]
                    threshold=i

            end=time.time()
            print "step time: " + str(int(end-begin))+" seconds"
            print threshold

            #Adjusting threshold by value of the function call
            threshold-=thresh
            pointmatrix=np.array
            begin=time.time()

            #turning any pixel below threshold white and all others black
            for j in range( 0, height):
                for i in range( 0, width):
                    if gray[i,j]>threshold:
                        gray[i,j]=0
                    else:
                        gray[i,j]=255

            end=time.time()
            print "step time: " + str(int(end-begin))+" seconds"
            print "thresholded image"

            """
            Checks for debugging
            namedWindow('dst_rt', WINDOW_NORMAL)
            #cv2.resizeWindow('dst_rt', window_width, window_height)

            imshow('dst_rt', gray)
```

```
        waitKey(0)
        destroyAllWindows()

        """

        G =nx.Graph()
        #Creating list of crack pixels
        pos={}
        k=int(0)

        for j in range( 0, height):
            for i in range( 0, width):
                if gray[i,j]>0:
                    gray[i,j]=int(255)
                    pos[k]=((height-j)*(-1),(width-i))
                    k+=int(1)

            if k%100 == 0:
                print "......"+str(int(100*(float(j)/float(height))))+"%\r",
        #print pos[0]

        print "created crack pixels"

        #creating nodes from crack pixels
        G.add_nodes_from(pos.keys())

        print "Created Nodes"

        nx.draw_networkx(G,pos,with_labels=False)
        #plt.axis('off')
        #plt.show()
        #print k

        print len(pos)

        #Creating list of pixel locations
        dictlist=[]
        temp=[]
        for key in range(0, len(pos)):
            temp = pos[key]
            dictlist.append(temp)

        #print len(dictlist)

        kdtree = spatial.KDTree(dictlist)
        other = kdtree
        #print type(dictlist)

        k=0
        print "made kd tree"

        #Takes the nodes of each crack pixel in a KD-tree and finds the nearest neighbor for th value specified
        #This is a much faster way of pairing nodes that are close to each other, rather than looking at each
        #node and comparing it to each other node
        begin=time.time()
        try:
```

```
            pairs=kdtree.query_pairs(7)
        except MemoryError:
            return

        #pairs=kdtree.query_ball_tree(other,r=40)
        #print type(pairs)

        end=time.time()
        print "step time: " + str(int(end-begin))+" seconds"

        #print type(pairs)
        #print len(pairs)
        print "made pairs"

        #adds edges between nodes that are close enough
        try:
            G.add_edges_from(pairs)
        except MemoryError:
            return

        #creates minimum spanning tree with kruskals algorithm in order to remove unneccessary edges
        #This is done to save memory and runtime
        s=nx.Graph()
        s=nx.algorithms.mst.minimum_spanning_tree(G)

        print "added edges"

        #connected crack pixels are drawn on a graph
        nx.draw_networkx(s,pos,False,width_labels=False,node_size=0)
        #plt.show()

        cv2.waitKey(0)
        cv2.destroyAllWindows()
        h=nx.Graph()
        h=s

        #removes nodes with only one connection, reduces errors
        outdeg = h.degree()
        to_remove=[n for n in outdeg if outdeg[n] ==1]
        h.remove_nodes_from(to_remove)

        plt.figure(figsize=(12.8,9.6))
        nx.draw_networkx(h,pos,False,with_labels=False,node_size=0)
        plt.axis('off')
        plt.subplots_adjust(left=0, bottom=0, right=1, top=1, wspace=0, hspace=0)
        plt.xlim((-1280,0))
        plt.ylim((0,960))

        #Saves Graph as a jpg
        plt.savefig('C:/Users/NUCANT/CrackTest/algocrack/'+str(name)+str(file)+'.jpg', dpi=100,pad_inches=0)
        #plt.show()
        plt.close()

        return(None)
```

Compare5xb.py

```
#compares Pixels
def compare(file,name):

    import networkx as nx
    import matplotlib.pyplot as plt
    import matplotlib.image as mpimg
    import matplotlib

    import numpy as np

    import cv2

    import sys
    import time

    from PIL import Image

    #Loads File crack file developed by algorithm from called arguments as well as
    #hand identified crack file
    im1=cv2.imread("C:/Users/NUCANT/CrackTest/Cracks2/"+str(file)+".png",cv2.CV_LOAD_IMAGE_GRAYSCALE)
    im2=cv2.imread("C:/Users/NUCANT/CrackTest/algocrack/"+str(name)+str(file)+".jpg",cv2.CV_LOAD_IMAGE_GRAYSCALE)
    print "test"

    #checks files are loaded
    #cv2.namedWindow('dst_rt')
    #cv2.imshow('dst_rt', im1)
    #cv2.waitKey(0)
    #cv2.destroyAllWindows()

    #image sizes
    width=960
    height=1280

    #checks files are loaded
    #cv2.resizeWindow('dst_rt', window_width, window_height)
    #cv2.resizeWindow('dst_rt', window_width, window_height)
    #cv2.waitKey(0)
    #cv2.destroyAllWindows()
    #cv2.resizeWindow('dst_rt', window_width, window_height)
    #plt.imshow(im2)
    #cv2.waitKey(0)
    #cv2.destroyAllWindows()

    span=15#distance to check pixels to check for truth,
    #if less, crack pixel is a false positive or false negative
    foundcrackpixels=0
    count=0
    truecrack=False
    #begins iteration of checking every ground truth crack pixel against the corresponding pixel
    #of the algorithm image within the specified range.  The loop counts all true positives and
    #all positives possible found by the ground truth.  False positives are delta between true positives and total positives.

```

```
    begin=time.time()
    for x in range(0,width-1):
        for y in range(0,height-1):
            if im1[x,y]<200:
                foundcrackpixels+=1
                for x2 in range(-span,span):
                    for y2 in (-span,span):
                        if (x-x2)<1 or (x+x2)<1 or (x+x2)>width-1 or (x-x2)>width-1:
                            break
                        if (y-y2)<1 or (y+y2)<1 or (y+y2)>height-1 or (y-y2)>height-1:
                            continue
                        if im2[x-x2,y-y2]<200:
                            truecrack=True
                if truecrack==True:
                    count+=1
                truecrack=False
        if x%10 == 0:
            print "......"+str(int(100*(float(x)/float(width))))+"%\r",

    print "First Run Completed"
    end=time.time()
    print "step time: " + str(int(end-begin))+" seconds"

    #begins iteration of checking every algorithm truth crack pixel against the corresponding pixel
    #of the true crack image within the specified range.  The loop counts all false negatives
    FalseNegative=0
    begin=time.time()
    for x in range(0,width-1):
        for y in range(0,height-1):
            if im2[x,y]<200:

                for x2 in range(-span,span):
                    for y2 in (-span,span):
                        if (x-x2)<1 or (x+x2)<1 or (x+x2)>width-1 or (x-x2)>width-1:
                            break
                        if (y-y2)<1 or (y+y2)<1 or (y+y2)>height-1 or (y-y2)>height-1:
                            continue
                        if im1[x-x2,y-y2]<200:
                            truecrack=True
                if truecrack==False:
                    FalseNegative+=1
                truecrack=False
        if x%10 == 0:
            print "......"+str(int(100*(float(x)/float(width))))+"%\r",
    end=time.time()
    print "step time: " + str(int(end-begin))+" seconds"

        #makes sure the script doesnt throw an error for dividing by zero
        if foundcrackpixels!=0:
            Precision=(float(count)/float(foundcrackpixels))
        else:
            Precision=0
        #print "true crack pixels = "+str(count)
        if count==0 and foundcrackpixels==0:
            Precision=1
```

```

        if FalseNegative!=0 and count!=0:
            Recall = float(count)/float(count+FalseNegative)
        else:
            Recall=0
        if count==0 and FalseNegative==0:
            Recall=1

        #print "false crack pixels ="+str(foundcrackpixels-count)
        #False Positive rate
        if Recall!=0 and Precision!=0:
            Fmeasure= 2*(Precision*Recall/(Precision+Recall))
        else:
            Fmeasure=0

        #Prints results of each image to a csv file that will be opened in excel
        #records the file number, Precision, Recall, and F-Measure
        print "File number: " + str(file)
        print "Precision =" + "{:.1%}".format(Precision)
        print "Recall =" + "{:.1%}".format(Recall)
        print "F-Measure =" + "{:.1%}".format(Fmeasure)

        return "{:.1%}".format(Precision)+'\t'+"{:.1%}".format(Recall)+'\t'+"{:.1%}".format(Fmeasure)
```

Appendix C. Quantitative Data from JMP®

Image	Recall	Precision	F-measure	Intensity Threshold	Shaded	Debris	Residual F-measure
File 1	0.992	0.259	0.411	35	0	0	0.018988889
File 2	0.99	0.312	0.474	35	0	0	0.035655556
File 3	0.988	0.193	0.323	35	0	0	0.022322222
File 4	0.995	0.389	0.559	35	0	0	0.016322222
File 5	0.994	0.254	0.405	35	0	0	0.026655556
File 6	0.994	0.262	0.414	35	0	0	0.023655556
File 7	0.994	0.368	0.537	35	0	0	0.036655556
File 8	0.992	0.318	0.481	35	0	0	0.037988889
File 9	0.991	0.31	0.473	35	0	0	0.040322222
File 10	0.993	0.411	0.581	35	0	0	0.036322222
File 11	0.992	0.214	0.353	35	0	0	0.019655556
File 12	0.992	0.121	0.215	35	0	0	0.003322222
File 13	0.985	0.282	0.439	35	0	1	0.009322222
File 14	0.969	0.083	0.153	35	0	0	-0.020011111
File 15	0.997	0.327	0.493	35	0	0	-0.019011111
File 16	0.996	0.086	0.158	35	0	1	-0.022677778
File 17	0.989	0.084	0.155	35	0	1	-0.021677778
File 18	0.994	0.119	0.213	35	0	1	-0.029011111
File 19	0.993	0.166	0.285	35	0	1	-0.027011111
File 20	0.992	0.394	0.564	35	0	0	0.038322222
File 21	0.992	0.252	0.401	35	0	0	0.023988889
File 22	0.993	0.303	0.464	35	0	0	0.036655556
File 23	0.988	0.475	0.641	35	0	0	0.027988889
File 24	0.99	0.387	0.556	35	0	0	0.024988889
File 25	0.99	0.293	0.452	35	0	0	0.030322222
File 26	0.988	0.163	0.279	35	0	0	0.008655556
File 27	0.994	0.202	0.336	35	0	0	0.023655556
File 28	0.988	0.172	0.293	35	0	0	0.012655556
File 29	0.986	0.147	0.256	35	0	1	0.005322222
File 30	0.996	0.387	0.558	35	0	1	-0.028011111
File 31	0.996	0.304	0.465	35	0	1	-0.001011111
File 32	0.997	0.322	0.486	35	0	1	-0.007677778
File 33	0.992	0.494	0.66	35	0	1	-0.031011111
File 34	0.942	0.099	0.179	35	0	1	-0.036011111
File 35	0.932	0.076	0.14	35	0	1	-0.035344444
File 36	1	0.164	0.282	35	0	1	0.003322222

File 37	0.988	0.156	0.27	35	0	1	-0.032011111
File 38	0.997	0.202	0.336	35	0	1	-0.005011111
File 39	0.998	0.16	0.276	35	0	1	-0.032677778
File 40	1	0.189	0.317	35	0	1	-0.005344444
File 41	0.997	0.161	0.278	35	0	1	-0.036677778
File 42	1	0.104	0.188	35	0	1	0.008988889
File 43	0.993	0.2	0.333	35	0	0	0.008322222
File 44	0.992	0.273	0.428	35	0	0	0.036322222
File 45	0.989	0.167	0.286	35	0	0	0.001322222
File 46	0.991	0.228	0.371	35	0	0	0.024655556
File 47	0.994	0.21	0.347	35	0	0	0.021655556
File 48	0.995	0.218	0.358	35	0	0	0.038988889
File 49	0.994	0.134	0.236	35	0	0	0.003322222
File 50	0.995	0.111	0.199	35	0	0	-0.008677778
File 51	0.99	0.116	0.208	35	0	0	-0.005011111
File 52	0.994	0.084	0.156	35	0	0	-0.016011111
File 53	0.995	0.108	0.196	35	0	1	-0.004011111
File 54	0.993	0.41	0.58	35	0	1	-0.024677778
File 55	0.995	0.2	0.333	35	0	0	-0.016011111
File 56	0.998	0.415	0.586	35	0	0	-0.015677778
File 57	0.999	0.279	0.436	35	0	1	-0.017011111
File 58	0.976	0.226	0.367	35	0	1	-0.021011111
File 59	0.992	0.122	0.217	35	0	1	-0.015344444
File 60	0.998	0.119	0.212	35	0	1	0.009988889
File 61	0.998	0.304	0.467	35	0	1	0.002655556
File 62	0.918	0.108	0.194	35	0	1	-0.046344444
File 63	0.97	0.079	0.146	35	0	1	-0.039677778
File 64	1	0.128	0.227	35	0	1	-0.029011111
File 65	0.977	0.15	0.26	35	0	1	-0.015344444
File 66	0.999	0.104	0.188	35	0	1	0.013322222
File 67	0.998	0.126	0.223	35	0	1	0.022655556
File 68	0.995	0.15	0.26	35	0	1	-0.014011111
File 69	0.992	0.359	0.527	35	0	0	0.016988889
File 70	0.988	0.323	0.487	35	0	0	0.014655556
File 71	0.991	0.42	0.59	35	0	0	0.017322222
File 72	0.991	0.32	0.484	35	0	0	0.032988889
File 73	0.992	0.246	0.394	35	0	0	0.019655556
File 74	0.991	0.268	0.422	35	0	0	0.019655556
File 75	0.992	0.267	0.421	35	0	0	0.022988889
File 76	0.991	0.306	0.468	35	0	0	0.019322222
File 77	0.996	0.188	0.317	35	0	0	0.010988889
File 78	0.992	0.361	0.529	35	0	1	-0.019011111

File 79	0.992	0.33	0.496	35	0	0	-0.006344444
File 80	0.997	0.455	0.625	35	0	0	-0.013011111
File 81	0.999	0.271	0.427	35	1	1	-0.014344444
File 82	1	0.228	0.371	35	1	1	-0.016011111
File 83	0.963	0.112	0.2	35	1	1	-0.030344444
File 84	0.981	0.138	0.242	35	1	1	-0.030677778
File 85	0.999	0.184	0.311	35	1	1	-0.023677778
File 86	0.996	0.186	0.313	35	1	1	-0.013344444
File 87	0.999	0.182	0.307	35	1	1	-0.028011111
File 88	0.999	0.193	0.323	35	1	1	-0.022344444
File 89	0.999	0.163	0.28	35	1	1	-0.020677778
File 90	0.996	0.231	0.376	35	1	1	-0.014011111
File 1	0.994	0.182	0.307	40	0	0	-0.000477778
File 2	0.99	0.214	0.352	40	0	0	-0.001811111
File 3	0.987	0.117	0.21	40	0	0	-0.006144444
File 4	0.996	0.299	0.46	40	0	0	0.001855556
File 5	0.995	0.17	0.29	40	0	0	-0.003811111
File 6	0.994	0.178	0.302	40	0	0	-0.003811111
File 7	0.994	0.266	0.42	40	0	0	0.004188889
File 8	0.993	0.213	0.351	40	0	0	-0.007477778
File 9	0.991	0.213	0.351	40	0	0	0.002855556
File 10	0.994	0.307	0.469	40	0	0	0.008855556
File 11	0.993	0.139	0.244	40	0	0	-0.004811111
File 12	0.991	0.064	0.12	40	0	0	-0.007144444
File 13	0.984	0.213	0.35	40	0	1	0.004855556
File 14	0.95	0.041	0.078	40	0	0	-0.010477778
File 15	0.997	0.275	0.432	40	0	0	0.004522222
File 16	0.996	0.049	0.093	40	0	1	-0.003144444
File 17	0.991	0.045	0.087	40	0	1	-0.005144444
File 18	0.997	0.084	0.155	40	0	1	-0.002477778
File 19	0.993	0.132	0.233	40	0	1	0.005522222
File 20	0.993	0.285	0.443	40	0	0	0.001855556
File 21	0.992	0.169	0.289	40	0	0	-0.003477778
File 22	0.994	0.202	0.336	40	0	0	-0.006811111
File 23	0.989	0.37	0.539	40	0	0	0.010522222
File 24	0.991	0.29	0.448	40	0	0	0.001522222
File 25	0.991	0.203	0.336	40	0	0	-0.001144444
File 26	0.988	0.099	0.18	40	0	0	-0.005811111
File 27	0.995	0.126	0.223	40	0	0	-0.004811111
File 28	0.988	0.107	0.194	40	0	0	-0.001811111
File 29	0.987	0.087	0.159	40	0	1	-0.007144444
File 30	0.997	0.339	0.507	40	0	1	0.005522222

File 31	0.996	0.232	0.376	40	0	1	-0.005477778
File 32	0.998	0.257	0.409	40	0	1	-0.000144444
File 33	0.993	0.444	0.614	40	0	1	0.007522222
File 34	0.933	0.069	0.128	40	0	1	-0.002477778
File 35	0.922	0.046	0.087	40	0	1	-0.003811111
File 36	1	0.108	0.194	40	0	1	-0.000144444
File 37	0.989	0.122	0.218	40	0	1	0.000522222
File 38	0.997	0.151	0.262	40	0	1	0.005522222
File 39	0.999	0.126	0.224	40	0	1	-0.000144444
File 40	1	0.139	0.245	40	0	1	0.007188889
File 41	0.997	0.131	0.232	40	0	1	0.001855556
File 42	1	0.056	0.106	40	0	1	0.011522222
File 43	0.994	0.132	0.233	40	0	0	-0.007144444
File 44	0.994	0.178	0.302	40	0	0	-0.005144444
File 45	0.988	0.111	0.199	40	0	0	-0.001144444
File 46	0.991	0.15	0.261	40	0	0	-0.000811111
File 47	0.994	0.133	0.235	40	0	0	-0.005811111
File 48	0.996	0.131	0.231	40	0	0	-0.003477778
File 49	0.995	0.076	0.141	40	0	0	-0.007144444
File 50	0.997	0.061	0.114	40	0	0	-0.009144444
File 51	0.991	0.064	0.121	40	0	0	-0.007477778
File 52	0.996	0.042	0.08	40	0	0	-0.007477778
File 53	0.997	0.056	0.105	40	0	1	-0.010477778
File 54	0.994	0.354	0.523	40	0	1	0.002855556
File 55	0.997	0.15	0.261	40	0	0	-0.003477778
File 56	0.998	0.353	0.522	40	0	0	0.004855556
File 57	0.999	0.229	0.372	40	0	1	0.003522222
File 58	0.979	0.185	0.311	40	0	1	0.007522222
File 59	0.996	0.081	0.149	40	0	1	0.001188889
File 60	1	0.082	0.151	40	0	1	0.033522222
File 61	0.999	0.238	0.385	40	0	1	0.005188889
File 62	0.917	0.086	0.157	40	0	1	0.001188889
File 63	0.961	0.052	0.098	40	0	1	-0.003144444
File 64	1	0.093	0.17	40	0	1	-0.001477778
File 65	0.977	0.11	0.197	40	0	1	0.006188889
File 66	1	0.049	0.093	40	0	1	0.002855556
File 67	1	0.072	0.135	40	0	1	0.019188889
File 68	0.997	0.106	0.192	40	0	1	0.002522222
File 69	0.992	0.272	0.426	40	0	0	0.000522222
File 70	0.987	0.243	0.39	40	0	0	0.002188889
File 71	0.991	0.327	0.492	40	0	0	0.003855556
File 72	0.991	0.222	0.363	40	0	0	-0.003477778

File 73	0.993	0.167	0.286	40	0	0	-0.003811111
File 74	0.991	0.192	0.322	40	0	0	0.004188889
File 75	0.994	0.187	0.315	40	0	0	0.001522222
File 76	0.991	0.224	0.365	40	0	0	0.000855556
File 77	0.997	0.121	0.216	40	0	0	-0.005477778
File 78	0.993	0.308	0.471	40	0	1	0.007522222
File 79	0.994	0.267	0.421	40	0	0	0.003188889
File 80	0.998	0.383	0.554	40	0	0	0.000522222
File 81	1	0.215	0.354	40	1	1	-0.002811111
File 82	1	0.176	0.299	40	1	1	-0.003477778
File 83	0.959	0.079	0.147	40	1	1	0.001188889
File 84	0.985	0.102	0.184	40	1	1	-0.004144444
File 85	1	0.145	0.253	40	1	1	0.002855556
File 86	0.997	0.138	0.242	40	1	1	0.000188889
File 87	0.999	0.14	0.246	40	1	1	-0.004477778
File 88	0.999	0.15	0.261	40	1	1	0.000188889
File 89	0.999	0.12	0.214	40	1	1	-0.002144444
File 90	0.996	0.179	0.303	40	1	1	-0.002477778
File 1	0.995	0.121	0.215	45	0	0	-0.018511111
File 2	0.991	0.14	0.246	45	0	0	-0.033844444
File 3	0.985	0.067	0.126	45	0	0	-0.016177778
File 4	0.996	0.224	0.366	45	0	0	-0.018177778
File 5	0.996	0.109	0.197	45	0	0	-0.022844444
File 6	0.994	0.119	0.212	45	0	0	-0.019844444
File 7	0.994	0.177	0.301	45	0	0	-0.040844444
File 8	0.994	0.145	0.254	45	0	0	-0.030511111
File 9	0.992	0.131	0.231	45	0	0	-0.043177778
File 10	0.994	0.206	0.341	45	0	0	-0.045177778
File 11	0.994	0.087	0.16	45	0	0	-0.014844444
File 12	0.99	0.03	0.057	45	0	0	0.003822222
File 13	0.98	0.148	0.257	45	0	1	-0.014177778
File 14	0.932	0.023	0.045	45	0	0	0.030488889
File 15	0.998	0.226	0.368	45	0	0	0.014488889
File 16	0.996	0.025	0.048	45	0	1	0.025822222
File 17	0.996	0.023	0.045	45	0	1	0.026822222
File 18	0.999	0.061	0.115	45	0	1	0.031488889
File 19	0.994	0.096	0.175	45	0	1	0.021488889
File 20	0.993	0.196	0.327	45	0	0	-0.040177778
File 21	0.993	0.11	0.198	45	0	0	-0.020511111
File 22	0.994	0.136	0.239	45	0	0	-0.029844444
File 23	0.989	0.263	0.416	45	0	0	-0.038511111
File 24	0.992	0.209	0.346	45	0	0	-0.026511111

File 25	0.991	0.133	0.234	45	0	0	-0.029177778
File 26	0.989	0.057	0.109	45	0	0	-0.002844444
File 27	0.996	0.072	0.135	45	0	0	-0.018844444
File 28	0.987	0.059	0.111	45	0	0	-0.010844444
File 29	0.987	0.05	0.094	45	0	1	0.001822222
File 30	0.998	0.29	0.45	45	0	1	0.022488889
File 31	0.997	0.186	0.314	45	0	1	0.006488889
File 32	0.999	0.207	0.343	45	0	1	0.007822222
File 33	0.993	0.386	0.556	45	0	1	0.023488889
File 34	0.93	0.05	0.095	45	0	1	0.038488889
File 35	0.941	0.029	0.056	45	0	1	0.039155556
File 36	1	0.062	0.117	45	0	1	-0.003177778
File 37	0.992	0.096	0.175	45	0	1	0.031488889
File 38	0.998	0.1	0.182	45	0	1	-0.000511111
File 39	0.999	0.101	0.183	45	0	1	0.032822222
File 40	1	0.088	0.162	45	0	1	-0.001844444
File 41	0.999	0.106	0.191	45	0	1	0.034822222
File 42	0	0.019	0	45	0	1	-0.020511111
File 43	0.995	0.09	0.165	45	0	0	-0.001177778
File 44	0.996	0.112	0.202	45	0	0	-0.031177778
File 45	0.988	0.068	0.126	45	0	0	-0.000177778
File 46	0.993	0.089	0.164	45	0	0	-0.023844444
File 47	0.996	0.082	0.151	45	0	0	-0.015844444
File 48	0.997	0.067	0.125	45	0	0	-0.035511111
File 49	0.996	0.041	0.078	45	0	0	0.003822222
File 50	0.998	0.034	0.067	45	0	0	0.017822222
File 51	0.99	0.035	0.067	45	0	0	0.012488889
File 52	0.997	0.019	0.037	45	0	0	0.023488889
File 53	0.999	0.029	0.056	45	0	1	0.014488889
File 54	0.996	0.306	0.468	45	0	1	0.021822222
File 55	0.998	0.117	0.21	45	0	0	0.019488889
File 56	0.998	0.294	0.454	45	0	0	0.010822222
File 57	0.999	0.182	0.308	45	0	1	0.013488889
File 58	0.979	0.139	0.243	45	0	1	0.013488889
File 59	0.999	0.046	0.088	45	0	1	0.014155556
File 60	0	0.054	0	45	0	1	-0.043511111
File 61	1	0.175	0.298	45	0	1	-0.007844444
File 62	0.923	0.068	0.127	45	0	1	0.045155556
File 63	0.951	0.036	0.07	45	0	1	0.042822222
File 64	1	0.068	0.128	45	0	1	0.030488889
File 65	0.978	0.067	0.126	45	0	1	0.009155556
File 66	0	0.01	0	45	0	1	-0.016177778

File 67	0	0.035	0	45	0	1	-0.041844444
File 68	0.998	0.068	0.127	45	0	1	0.011488889
File 69	0.992	0.201	0.334	45	0	0	-0.017511111
File 70	0.987	0.175	0.297	45	0	0	-0.016844444
File 71	0.992	0.245	0.393	45	0	0	-0.021177778
File 72	0.991	0.151	0.263	45	0	0	-0.029511111
File 73	0.994	0.111	0.2	45	0	0	-0.015844444
File 74	0.992	0.124	0.22	45	0	0	-0.023844444
File 75	0.994	0.121	0.215	45	0	0	-0.024511111
File 76	0.992	0.156	0.27	45	0	0	-0.020177778
File 77	0.997	0.077	0.142	45	0	0	-0.005511111
File 78	0.994	0.251	0.401	45	0	1	0.011488889
File 79	0.995	0.21	0.347	45	0	0	0.003155556
File 80	0.998	0.327	0.492	45	0	0	0.012488889
File 81	1	0.176	0.3	45	1	1	0.017155556
File 82	1	0.142	0.248	45	1	1	0.019488889
File 83	0.961	0.053	0.101	45	1	1	0.029155556
File 84	0.989	0.08	0.149	45	1	1	0.034822222
File 85	1	0.109	0.197	45	1	1	0.020822222
File 86	0.998	0.099	0.181	45	1	1	0.013155556
File 87	0.999	0.117	0.209	45	1	1	0.032488889
File 88	0.999	0.117	0.209	45	1	1	0.022155556
File 89	0.999	0.09	0.165	45	1	1	0.022822222
File 90	0.997	0.142	0.248	45	1	1	0.016488889

Appendix D. Published Works

3D printing makes EXPLOSIVE headway

By 1st Lt. Bradford Shields and Maria Meeks
Air Force Institute of Technology Students

In a time of declining budgets and increasing demands, Air Force civil engineering is searching for more efficient technologies and innovative processes to complete mission requirements. Additive manufacturing, commonly known as 3D printing, is a burgeoning technology that offers cost-effective and flexible methods to produce unique objects on demand. For high-value, low-demand items, AM can offer significant cost savings, reduce logistical time and increase flexibility in configuration management. The importance and role of AM cannot be underestimated in austere environments, especially for military applications that often utilize systems made of one-of-a-kind components.

The opportunities for AM within civil engineer squadrons are endless; however, very few applications have been researched. Explosive ordnance disposal operations afforded us this opportunity. A graduate research effort at the Air Force Institute of Technology demonstrates one possible application of AM technology to military operations. Specifically, students researched the mission of the 88th CES EOD flight at Wright-Patterson Air Force Base, Ohio, to attach environmental sensors to a remote-controlled robot.

EOD robots

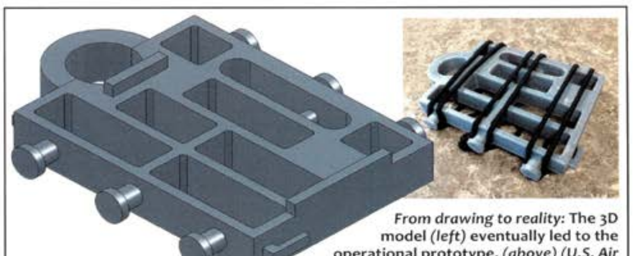
EOD technicians at Wright-Patterson employ the Northrup-Grumman Remotec® unmanned ground vehicle for hazardous duty operations, which include field inspection and detonation of explosive devices. In some operations, the vehicle must be fitted with environmental sensors to detect chemical and radiological threats. Both the vehicle and sensors are specialized equipment and are uniquely paired on a case-by-case basis. Currently, technicians use adhesive tape to secure these sensors to the vehicle's arm and spend valuable time removing each sensor from the robot. Although this is an effective, low-cost solution, it takes a significant amount of time to change sensors in the field and during post-mission clean-up. A universal bracket would reduce that time and effort, but no bracket capable of mounting differently shaped sensors to the UGV is commercially available.

At Wright-Patterson, EOD technicians typically utilize four different sensors for environmental sampling and ordnance testing. All sensors operate independently; there is no recurring need to attach more than one sensor to the robot at any one time. Because of its size, the Victoreen® Fluke® Biomedical 451P sensor was used for the universal bracket prototype design.

In the lab

The laboratory equipment used for the design and production of the EOD bracket prototypes included a 3D Systems® ProJet™ 1500, printer, polycarbonate solvent washer and a UV lamp for curing. The design and production process began with an initial design created in a 3D modeling program. The software allowed the design team to have a firm grasp on the exact shape and dimensions of the bracket before actually creating the prototype.

The driving factor in print time for all machines is the total height of the print, or depth in the z-coordinate direction. The rule of thumb for printing with the ProJet® 1500 is approximately four hours per inch printed in the z-direction. The actual print time changes based on the part's geometry.



From drawing to reality: **The 3D model *(left)* eventually led to the operational prototype. *(above)* (U.S. Air Force photo/1st Lt. Bradford Shields/Released)**

The final step, and one that is often overlooked, is post-processing, which follows three basic steps. First, a solvent wash is employed to remove uncured material from the prototype's surface. Then, curing in a UV lamp cabinet increases the strength of the prototype. Finally, support structures are cut away and the surface is smoothed.

Results

Unmanned ground vehicles help EOD technicians identify ordnance through cameras and video feeds. The vehicle also is used to disarm ordnance. Because these two capabilities are critical to neutralizing threats, the research team determined that any AM solution also must maintain these capabilities. Full range of motion of the UGV arm assembly and visibility of the sensor display were key design drivers.

The research team used the spiral process model commonly used by systems engineers, to ensure all factors were considered in the design. The team designed, analyzed and manufactured four prototypes in nine weeks.

The process also allowed the team to address and resolve two AM development factors: poor tolerances from the 3D printer used and printing time reduction. The design of the operational prototype resulted from the various successes and failures of the first three prototypes. The final design concept employs a base plate to cradle the sensor and integrated studs with commonly available bungee fasteners attached to hold the sensor in place.

The team encountered several challenges during the final stage of development. An imperfect method of detaching the printed part from its supports often left uneven surfaces that required additional tooling. Also, thinner dimensions on the printed part were at risk for breakage. Interior supports were not easily accessible and sometimes required much effort to remove completely. Finally, the printing mat — the surface where the part is produced — was extremely difficult to remove from the printer plate and required rigorous cleaning between prints.

The new bracket exceeded expectations during testing, saving EOD technicians 60 to 90 seconds when changing sensors. *(U.S. Air Force photo/Maria Meeks/Released)*

A successful preliminary test of the final bracket was performed in March 2015 at the 88th CES EOD flight. In early August, the 88th CES EOD flight set up a challenge course to test the bracket under normal field and operating conditions. For this test, EOD training aides were placed in locations similar to those where explosives might be found in an operational situation. The bracket performed exceptionally well, saving between 60 and 90 seconds in switching to different sensors. During all situations, except when in a low-lit area, the robot's main camera was able to capture the readings from each of the EOD biological and HAZMAT sensors. With 3D printing, Airmen can easily design a simple mount for a camera and light to overcome the low-light challenge.

Along with the four different sensors, the EOD Airman conducted an unplanned test of the bracket. A PDX/2 LRM radionuclide sensor, weighing 15 pounds that's used for searching large shipping containers, was strapped to the bracket. Normally, an Airman would wear this sensor while sweeping an area; however, the printed bracket securely held the backpack and found the hidden training aide. Overall, this final round of testing resulted in the bracket exceeding expectations, cutting the time required to switch sensors, and possibly saving EOD flights around the world from having to send their Airmen into harm's way. The design team will make a few minor changes and then present the 88th CES EOD flight with its very own 3D printed EOD HAZMAT sensor mount.

Outlook

AM could revolutionize operations within the Air Force CE community. This research demonstrated that a universal bracket for unmanned ground vehicles could be designed, analyzed and manufactured within nine weeks using systems engineering principles and relatively low-cost 3D printing equipment. The digital file for the operational prototype could be shared with EOD units across the Air Force.

The overall method and parts produced could be duplicated Air Force-wide at a relatively low cost.

We foresee AM being an integral part of CES operations. With simple file sharing, engineers worldwide could collaborate to solve problems encountered in the field. The process of developing and testing prototypes in the field would be greatly expedited because design, production, transportation cost and time requirements are reduced or eliminated. This research demonstrates that AM could provide a solution to the growing demands for CE capabilities and long-term budget constraints within the U.S. Air Force.

Editor's Note: Shields and Meeks are students at the Air Force Institute of Technology Department of Engineering Management and Systems Engineering. The adviser for the authors was Maj. Vhance Valencia.

Bibliography

AFCEC. (2014a). *7115 Report - FY14 USAF Real Property Inventory*.

AFCEC. (2014b). *Business Rules for Execution of the FY15-16 CAMP and FY17-21 AMP Processes*.

Alibaba.com. (n.d.). Pan and Tilt Duct Inspection Robot. Retrieved April 9, 2015, from http://www.alibaba.com/product-detail/Pan-and-Tilt-Duct-Inspection-Robot_60118925456.html

AVT. (n.d.). Prosilica Cameras. Retrieved from https://www.alliedvision.com/en/digital-industrial-camera-solutions.html

Cancilla, S. (2016). *Interview*.

Clipart, C. (n.d.). Worker clipart. Retrieved April 9, 2015, from http://www.cloudclipart.com/clipart/free-clipart/worker-detail

Costello, S. B., Chapman, D. N., Rogers, C. D. F., & Metje, N. (2007). Underground asset location and condition assessment technologies. *Trenchless Technology*, *22*(5–6), 524–542. Retrieved from http://www.sciencedirect.com/science/article/pii/S0886779807000661

DoD Deputy Chief Information Officer. (2015). *DODAF - DOD Architecture Framework Version 2.02*. Retrieved from http://dodcio.defense.gov/Library/DoDArchitectureFramework.aspx

Duran, O., Althoefer, K., & Seneviratne, L. D. (2002). State of the art in sensor technologies for sewer inspection. *Sensors Journal, IEEE*, *2*(2), 73–81.

DUSD(I&E). (2015). DUSD(I&E) - 2015 - Base Structure Report - FY15 Baseline. Retrieved December 17, 2015, from http://www.acq.osd.mil/ie/index.shtml

El-Akruti, K., Dwight, R., & Zhang, T. (2013). The strategic role of Engineering Asset Management. *International Journal of Production Economics*, *146*(1), 227–239. Retrieved from http://www.sciencedirect.com/science/article/pii/S0925527313003149

Executive Order 13327 (2004). FR Doc. 04-2773. Retrieved from https://www.gpo.gov/

FiberScope.net. (2015). Pipe Crawler STORMER S3000. Retrieved December 9, 2015, from http://www.fiberscope.net/pipe-crawler-robot-stormer.html

Galehouse, L., Moulthrop, J., & Hicks, R. G. (2011). Federal Highway Administration - Principles of Pavement Preservation. Retrieved February 1, 2016, from https://www.fhwa.dot.gov/pavement/preservation/ppc0621.cfm

Google. (n.d.). Google Earth.

Grandsaert, P. J. (2015). *Integrating Pavement Crack Detection and Analysis Using Autonomous Unmanned Aerial Vehicle Imagery*. Retrieved from http://www.dtic.mil/docs/citations/ADA615401

Guo, W., Soibelman, L., & Garrett Jr., J. H. (2009). Automated defect detection for sewer pipeline inspection and condition assessment. *Automation in Construction*, *18*(5), 587–596. Retrieved from http://www.sciencedirect.com/science/article/pii/S0926580508001842

Guotte, C., & Gaussier, E. (2005). A Probabilistic Interpretation of Precision, Recall and F-Score, with Implication for Evaluation. In D. E. Losada & J. M. Fernández-Luna (Eds.), *Advances in Information Retrieval* (Vol. 3408). Berlin, Heidelberg: Springer Berlin Heidelberg. http://doi.org/10.1007/b107096

Harmonic Mean Calculator, Formula & Calculation. (n.d.). Retrieved February 2, 2016, from http://ncalculators.com/statistics/harmonic-mean-calculator.htm

Isaacs, L. (2015). *Interview*.

ISO. (2010). ISO 24765 - Systems and Software Engineering Vocabulary. *ISO/IEC/IEEE 24765:2010(E)*. http://doi.org/10.1109/IEEESTD.2010.5733835

Kirkham, R., Kearney, P., & Rogers, K. (2000). PIRAT--A System for Quantitative Sewer Pipe Assessment. *The International Journal of Robotics Research, 19*(11), 1033–1053. http://doi.org/10.1177/02783640022067959

Klette, R. (2014). *Concise Computer Vision - An Introduction into Theory and | Reinhard Klette | Springer*. Springer Science and Business Media. Retrieved from http://www.springer.com/us/book/9781447163190

Koo, D.-H., & Ariaratnam, S. T. (2006). Innovative method for assessment of underground sewer pipe condition. *The First Conference on the Future of the AEC Industry (BFC05)*, *15*(4), 479–488. Retrieved from http://www.sciencedirect.com/science/article/pii/S0926580505000725

Machin, T. (2016). *Near Real-Time Implementation of Autonomous Vision-Aided Monocular Navigation of Fixed-Wing Aircraft*.

McKim, R. A., & Sinha, S. K. (1999). Condition assessment of underground sewer pipes using a modified digital image processing paradigm. *Tunnelling and Underground Space Technology*, *14, Supple*(0), 29–37. Retrieved from http://www.sciencedirect.com/science/article/pii/S0886779800000213

NASSCO. (n.d.). Pipeline Assessment and Certification Program (PACP) Technical Manual. Retrieved December 9, 2015, from http://nassco.org/

Nassiraei, A., Honda, M., & Ishii, K. (2010). New approach to the self-localization of autonomous sewer inspection robots, 1–7. Retrieved from http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=5665397

Nassiraei, A., Kawamura, Y., Ahrary, A., Mikuriya, Y., & Ishii, K. (2007). Concept and Design of A Fully Autonomous Sewer Pipe Inspection Mobile Robot “KANTARO.” In *Robotics and Automation, 2007 IEEE International Conference on* (pp. 136–143).

RedZone-Robotics. (n.d.). RedZone® Solo Robots. Retrieved November 24, 2015, from http://www.redzone.com/

RobotShop. (n.d.). LIDAR-Lite 2 Laser Rangefinder (PulsedLight) - RobotShop. Retrieved January 1, 2016, from http://www.robotshop.com/en/lidar-lite-2-laser-rangefinder-pulsedlight.html

Romanova, A., Horoshenkov, K. V, Tait, S. J., & Ertl, T. (2012). Sewer inspection and comparison of acoustic and CCTV methods. *Proceedings of the ICE-Water Management*, *166*(2), 70–80.

Shel-Daat, T. (n.d.). Gnostic Students Network. Retrieved April 9, 2015, from https://plus.google.com/communities/102119423336749494689/stream/7b2e19a2-a91f-40ee-b648-9803550cc9be

Shields, B. (2016). *United States Air Force Additive Manufacturing Applications: Tools and Jigs*.

Ting, K. (2011). Precision and Recall. In *Encyclopedia of Machine Learning* (p. 1031). Springer Science & Business Media. Retrieved from https://books.google.com/books?hl=en&lr=&id=i8hQhp1a62UC&pgis=1

TMI Africa. (n.d.). Asset Management Introduction. Retrieved September 28, 2015, from http://www.tmiafrica.org/2015/06/23/asset-management-a-introduction/

Tran, H., Perera, B., & Ng, A. (2009). Predicting Structural Deterioration Condition of Individual Storm-Water Pipes Using Probabilistic Neural Networks and Multiple Logistic Regression Models. *Journal of Water Resources Planning and Management*, *135*(6), 553–557.

Uddin, W., Hudson, W. R., & Haas, R. (2013). *Public Infrastructure Asset Management* (2nd ed.). McGraw-Hill Education.

USAF. (2011). *US Air Force Civil Engineering Strategic Plan*. Retrieved from https://cs1.eis.af.mil/sites/ceportal/SiteCollectionDocuments/2011 U.S. Air Force Civil Engineering Strategic Plan.pdf

USAF. (2012). *Programming Plan (P-Plan) for Implementation of Enterprise-Wide Civil Engineer Transformation (PAD 12-03), Volume 3*.

Valencia, V., & Grandsaert, P. (2015). Assessments from the Air. *The Military Engineer*, 42–43. Retrieved from http://themilitaryengineer.com/tme_online/2015_Sept_Oct/TME-SeptOct2015-Online-SM-2.pdf

Vanier, D. J. “Dana.” (2001). Why Industry Needs Asset Management Tools. *Journal of Computing in Civil Engineering*, *15*(1), 35–43. http://doi.org/10.1061/(ASCE)0887-3801(2001)15:1(35)

Wirahadikusumah, R., Abraham, D. M., Iseley, T., & Prasanth, R. K. (1998). Assessment technologies for sewer system rehabilitation. *Automation in Construction*, *7*(4), 259–270. Retrieved from http://www.sciencedirect.com/science/article/pii/S092658059700071X

Zou, Q., Cao, Y., Li, Q., Mao, Q., & Wang, S. (2012). CrackTree: Automatic crack detection from pavement images. *Pattern Recognition Letters*, *33*(3), 227–238. http://doi.org/10.1016/j.patrec.2011.11.004

Vita

Ms. Maria Meeks graduated from Mountain Home High School in Mountain Home, Idaho. She began her civilian service with the U.S. Air Force in 2001 as a student trainee intern at the 366th Environmental Flight, Mountain Home AFB, Idaho. She graduated from Boise State University in 2004 with a Bachelor of Science in Civil Engineering (BSCE).

During her time at Mountain Home AFB, Ms. Meeks managed several environmental compliance programs and in 2007 became the installation Pavements Engineer. In 2009, she was assigned to HQ USAFE staff at Ramstein AB, Germany serving as the Asset Optimization Branch Chief and later as SRM Programmer. In August 2014, she entered the Air Force Institute of Technology at Wright-Patterson Air Force Base, Ohio and earned a Master of Science degree in Engineering Management with a focus in infrastructure asset management. Upon graduation, she will be assigned to the Air Force Civil Engineer Center (AFCEC) Environmental Operations Division, Joint Base San Antonio, Texas.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704–0188*

1. REPORT DATE (DD–MM–YYYY)	2. REPORT TYPE	3. DATES COVERED *(From — To)*
24-03-2016	Master's Thesis	1 Aug 2014 – 24 Mar 2016

4. TITLE AND SUBTITLE Evaluating Storm Sewer Pipe Condition Using Autonomous Drone Technology	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) Meeks, Maria T., Ms., Civilian	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology Graduate School of 2950 Hobson Way WPAFB OH 45433-7765	8. PERFORMING ORGANIZATION REPORT NUMBER AFIT-ENV-MS-16-M-167
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Intentionally Left Blank	10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Distribution Statement A. Approved for Pubic Release, Distribution Unlimited

13. SUPPLEMENTARY NOTES

This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

14. ABSTRACT

The United States Air Force (USAF) owns a total of 30.9 million linear feet (LF) of storm sewer pipes valued at approximately $2.3B in its vast portfolio of built infrastructure. Current inventory records reveal that 78% of the inventory (24.1 million LF) is over 50 years old and will soon exceed its estimated service life. Additionally, the USAF depends on contract support while its business processes undervalue in-service evaluations from long-term funding plans. Ultimately, this disconnect negatively impacts infrastructure performance and overall strategic success, and the USAF risks making uninformed decisions in a fiscally constrained environment.

This research presents a proof of concept effort to automate storm sewer evaluations for the USAF using unmanned ground vehicles and computer vision technology for autonomous defect detection. The results conceptually show that a low-cost autonomous system can be developed using commercial off the shelf (COTS) hardware and open-source software to quantify the condition of underground storm sewer pipes with an efficiency of 36%. While the results show that the prototype developed for this research is not sufficient for operational use, it does demonstrate that the USAF can leverage COTS systems in future AM strategies to improve asset visibility at a significantly lower cost.

15. SUBJECT TERMS

Asset Management, Sewer, Automate, Computer Vision, Drone

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON Major Vhance V. Valencia, PhD (ENV)
a. REPORT U	b. ABSTRACT U	c. THIS PAGE U	UU	177	19b. TELEPHONE NUMBER (Include Area Code) (937) 255-6565, ext 4826 Vhance.Valencia@us.af.mil

Standard Form 298 (Rev. 8–98)
Prescribed by ANSI Std. Z39.18