Development of a Vision-based System for Aerial Pipeline Right-of-Way surveillance.

By

EHIDIAMEN Osesenaga Daniel	**ENG1905687**
IYAMAH Eseosa Timothy	**ENG1805487**
OMUSUKU Favour Omusuku	**ENG1805506**

DEPARTMENT OF MECHATRONICS ENGINEERING
FACULTY OF ENGINEERING
UNIVERSITY OF BENIN, BENIN-CITY.

IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF BACHELOR IN ENGINEERING DEGREE
DEPARTMENT OF MECHATRONICS ENGINEERING, UNIVERSITY OF BENIN.

MAY, 2024.

CERTIFICATION

This is to certify that the research project submitted to the Department of Mechanical

Engineering was carried out by EHIDIAMEN Osesenaga Daniel, IYAMAH Eseosa Timothy and OMUSUKU Favour Omusuku of the Department of Mechatronics Engineering, University of Benin, Benin City, Edo State, Nigeria, under the supervision of Professor Godfrey O. Ariavie.

______________________ ______________________

PROF. G.O. ARIAVIE **DATE**

PROJECT SUPERVISOR

______________________ ______________________

PROF. P.O OLAGBEGI **DATE**

PROJECT COORDINATOR

______________________ ______________________

PROF. G.O. SADJERE **DATE**

HEAD OF DEPARTMENT

DEDICATION

We dedicate this project to God almighty for all his mercies and grace upon us and to our loving parents and family for their love and care during our stay in the University of Benin.

ACKNOWLEDGEMENTS

We would like to express our sincere appreciation to all those who have supported me in completing this project. First and foremost, we want to thank God Almighty for watching over us and guiding us throughout our life.

Also, we are grateful to my supervisor, Prof. G.O Ariave, for his guidance, feedback, and encouragement throughout this project. We acknowledge the impact over the years of some of my lecturers and the influence these individuals have had in shaping my live as a Mechatronics engineering student in training. The persons of Prof Ebunilo, Engr. Peter Olagbegi, Dr. kwasi, Engr. G. Linus, Dr. O. Ighodaro, Dr. H. Egware and others, to name a few.

We also thank our classmates for their valuable suggestions and constructive criticism. Finally, we want to thank my family and friends for their moral support and understanding during this challenging period. This project t would not have been possible without the help of all these people.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

APM	ArduPilot Mega
CARS	Coherent Anti-Raman Spectroscopy
CEPA	Canadian Energy Pipeline Association
CP	Cathodic Protection
DEM	Digital Elevation Model
eia	Energy Information Administration
EMR	Electro-Magnetic Radiation
EMS	Electro-Magnetic Spectrum
EO	Electro-Optical
EP	Endpoint
ESC	Electronic Speed Controller
FAA	Federal Aviation Administration
FOV	Field-Of- View
FPV	First-Person View
FTIR	Fourier Transform Infrared Spectroscopy
GBRA	Guadalupe-Blanco River Authority
GCS	Ground Control Station
GPS	Global Positioning System
GUI	Graphical User Interface
HIS	Hue, Intensity and Saturation
HT	Hough Transform
IACC	Impressed Alternating Cycle Current
IMU	Inertial Measurement Unit
IDE	Integrated Development Environment
IR	Infra-Red
LIDAR	Light Detection and Ranging
LIF	Laser Induced Fluorescence

LUV	Luminance, Chrominance U and Value
ML	Machine Learning
mmW	milli-meter Wave
MRT	Multi-Rotor Technology
MVD	Minimum Vector Dispersion
OPS	Office of Pipeline Safety
PHMSA	Pipeline & Hazardous Materials Safety Administration
PID	Proportional, Integral, Derivative
QVGA	Quarter Video Graphics Array
RANSAC	Random Sample Consensus
RC	Remote Control
RF	Radio Frequency
RGB	Red, Green, and Blue colours
RGBD	Red, Green, Blue, and Depth
ROS	Robot Operating System
ROW	Right-of -Way
SAR	Synthetic Aperture Radar
SAS	Stability Augmentation System
SXGA	Super Extended Graphics Array
TAD	Turn Anticipation Distance
TAPS	Trans-Alaska Pipeline System
TCP	Transmission Control Protocol
TDLAS	Tunable Diode Laser Absorption Spectroscopy
ToF	Time of Flight
UAV	Unmanned Air-Vehicle
UDP	User Datagram Protocol
UV	Ultra-Violet
VGA	Video Graphics Array
VTOL	Vertical Take-Off & Landing
WP	Waypoint

LIST OF SYMBOLS

$\boldsymbol{\rho}$	The shortest distance from the corner of the image space tothe straight line
$\boldsymbol{\theta}$	The angle between the shortest distance ρ and the horizontalaxis (width) in image space
$\boldsymbol{x}, \boldsymbol{y}$	Width and height of the image (pixels)
$\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$	3D-world coordinates
$x_{p}(\mathbf{1}), y_{p}(\mathbf{1})$	Width and height of the first endpoint of the real pipelinesegment (pixels)
$x_{p}(\mathbf{2}), y_{p}(\mathbf{2})$	Width and height of the second endpoint of the real pipeline segment (pixels)
$\boldsymbol{x_{UR}}, \boldsymbol{y_{UR}}$	Width and height of the upper-right corner of the image frame (pixels)
$\boldsymbol{x_{UL}}, \boldsymbol{y_{UL}}$	Width and height of the upper-left corner of the image frame(pixels)
$\boldsymbol{x_{DR}}, \boldsymbol{y_{DR}}$	Width and height of the down-right corner of the image frame (pixels)
$\boldsymbol{x_{DL}}, \boldsymbol{y_{DL}}$	Width and height of the down-left corner of the image frame (pixels)
$\boldsymbol{F_{x_{max}}}, \boldsymbol{F_{y_{max}}}$	Maximum width and height of the boundary frame
$\boldsymbol{F_{x_{min}}}, \boldsymbol{F_{y_{min}}}$	Minimum width and height of the boundary frame
$\boldsymbol{P_{x_U}}, \boldsymbol{P_{y_U}}$	Width and height of the line segment interception with theupper line of the image frame (pixels)
$\boldsymbol{P_{x_D}}, \boldsymbol{P_{y_D}}$	Width and height of the line segment interception with thedown line of the image frame (pixels)
$\boldsymbol{P_{x_L}}, \boldsymbol{P_{y_L}}$	Width and height of the line segment interception with the leftline of the image frame (pixels)
$\boldsymbol{P_{x_R}}, \boldsymbol{P_{y_R}}$	Width and height of the line segment interception with theright line of the image frame (pixels)

$P_{X_{max(1)}}, P_{Y_{max(1)}}$	Maximum width and height of the correspond endpoint in the image frame (pixels)
$P_{X_{min(1)}}, P_{Y_{min(1)}}$	Minimum width and height of the correspond endpoint in theimage frame (pixels)
l	Length of the real pipeline segment
x_{p1}, y_{p1}	Width and height of the first endpoint of the real pipelinesegment relative to the image frame
x_{p2}, y_{p2}	Width and height of the second endpoint of the real pipeline segment relative to the image frame
l_{11}	The length between the first endpoint of real pipeline segmentto the first endpoint of the Hough line segment
l_{12}	The length between the first endpoint of real pipeline segment to the second endpoint of the Hough line segment
l_{21}	The length between the second endpoint of real pipelinesegment to the first endpoint of the Hough line segment
l_{22}	The length between the second endpoint of real pipelinesegment to the second endpoint of the Hough line segment
$x_{H(1)}, y_{H(1)}$	Width and height of the first endpoint of the Hough linesegment (pixels)
$x_{H(2)}, y_{H(2)}$	Width and height of the second endpoint of the Hough line segment (pixels)
a_1	The angle between the real pipeline segment and the connected line between the first endpoint of real pipeline segment and the second endpoint of the Hough line segment
a_2	The angle between the real pipeline segment and the connected line between the second endpoint of real pipeline segment and the first endpoint of the Hough line segment
ρ_1	The shortest distances of the first endpoints the pair of segments
ρ_2	The shortest distances of the second endpoints the pair of segments

$\boldsymbol{\rho_a}$	The average distance between a pair of segments
$\boldsymbol{\rho_p}$	The parallelism factor or difference between a pair of segments
$\boldsymbol{u}, \boldsymbol{v}$	Width and height of the depth image, respectively
$\boldsymbol{d_{raw}}$	Raw of the depth value
$\boldsymbol{I_d}$	Depth matrix
$\boldsymbol{I_{dh}}$	Depth homogenous matrix
$\boldsymbol{P_{3D}}$	3D point cloud matrix
$\boldsymbol{P_{h3D}}$	Homogenous 3D point cloud matrix
$\boldsymbol{K_{IR}}$	Intrinsic calibration matrix of the IR sensor
$\boldsymbol{X_w}$	x-axis of the 3D point cloud coordinates
$\boldsymbol{Y_w}$	y-axis of the 3D point cloud coordinates
$\boldsymbol{Z_w}$	z-axis of the 3D point cloud coordinates
$\boldsymbol{W_w}$	Homogenous factor of the 3D point cloud coordinates
a	Minimum probability of finding at least one good set of observations in (***N)*** trials.
$\boldsymbol{P_{3D}edges}$	3D point cloud of the edges
$\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D}$	Plane parameters
$\boldsymbol{H_{max}Pipeline}$	Maximum height of the existing pipeline structure
$\boldsymbol{S}$	endpoints list of the detected Hough lines
$\boldsymbol{t_w}$	maximum width of the actual pipeline in meter
$\boldsymbol{t_l}$	Minimum acceptable length of the pipeline in meter
$\boldsymbol{N_{v}m}$	Minimum number of 3D point cloud in vicinity of pipelinestructure

N_v	number of 3D point cloud in vicinity of pipeline structure
l_M	Centreline segment length
P_{s_M}	Start-point of the centreline segment
P_{e_M}	End-point of the centreline segment
$L(P_s, P_e)$	Line segment
$L_M(P_{s_M}, P_{e_M})$	Centreline segment
P_s	Start-point of the line segment
P_e	End-point of the line segment
$d(P, L)$	Distance between point to line
$\vec{V}_L$	Direction vector of the line segment
$\vec{V}_P$	Direction vector of the point to the line segment
P_{s_1}	Start-point of the first line segment
P_{e_1}	End-point of the first line segment
P_{s_2}	Start-point of the second line segment
P_{e_2}	End -point of the second line segment
x_{s_M}	x-axis of start-point of the centreline segment in 3D pointcloud coordinates
y_{s_M}	y-axis of start-point of the centreline segment in 3D pointcloud coordinates
z_{s_M}	z-axis of start-point of the centreline segment in 3D pointcloud coordinates
x_{e_M}	x-axis of end-point of the centreline segment in 3D point cloud coordinates
y_{e_M}	y-axis of end-point of the centreline segment in 3D point cloud coordinates
z_{e_M}	z-axis of end-point of the centreline segment in 3D point cloud coordinates

$\boldsymbol{P_{In}}$	Inlier 3D point cloud
$\boldsymbol{P_{Out}}$	Outlier 3D point cloud
$\boldsymbol{R}$	Matrix rotations
$\boldsymbol{T}$	Matrix translations
$\boldsymbol{I_{RGB}}$	Matrix of the RGB data
$\boldsymbol{Ih_{RGB}}$	Homogenous matrix of the RGB data
$\boldsymbol{u', v'}$	Width and height of the RGB image
$\boldsymbol{ii}$	The value of the integral image
$\boldsymbol{i}$	The value of the original image
$\boldsymbol{s}$	the sum of the cumulative row
$\boldsymbol{w_{t,i}}$	Weight of image at each boosting trail and each image frame
$\boldsymbol{t}$	Sequence of boosting trial
$\boldsymbol{n}$	Sequence of image frame
$\boldsymbol{h_j}$	Weak learner
$\boldsymbol{ns}$	Number of the negative sample in Haar classifier
$\boldsymbol{ps}$	Number of the positive sample in Haar classifier
$\boldsymbol{x_{sw}}$	Width and height of the sub-window of the image
$\boldsymbol{f}$	Feature
$\boldsymbol{p}$	Polarity or the direction of the inequality sign
$\boldsymbol{\theta}$	Threshold
$\boldsymbol{j}$	Feature number
$\boldsymbol{\epsilon_j}$	Error of each feature
$\boldsymbol{x_{3rd}}$	Third-party interference location at x-axis
$\boldsymbol{y_{3rd}}$	Third-party interference location at y-axis
$\boldsymbol{h_d}$	Desired altitude
$\boldsymbol{\psi_d}$	Heading demand
$\boldsymbol{\phi_d}$	Bank angle demand
$\boldsymbol{x_a}$	Position of the air-vehicle at x-axis

y_a	Position of the air-vehicle at y-axis
z_a	Position of the air-vehicle at z-axis
V_A	Speed of the air-vehicle
v_x	Speed of the air-vehicle at x direction
v_y	Speed of the air-vehicle at y direction
v_z	Speed of the air-vehicle at z direction
WP_c	Current waypoint of the course
WP_t	Target waypoint of the course
WP_f	Future waypoint of the course
EP_c	Current endpoint of the pipeline segment
EP_t	Target endpoint of the pipeline segment
EP_f	Future endpoint of the pipeline segment
$dx_{c,n}$	North vector of the current endpoint
$dy_{c,n}$	East vector of the current endpoint
$\Delta\chi_T$	Course angle change
χ_c	Current segment angle
χ_n	Next segment angle
$WP_{t_{north}}$	North coordinate of target waypoint
$WP_{c_{east}}$	East coordinate of current waypoint
$WP_{f_{north}}$	North coordinate of future waypoint
$WP_{c_{north}}$	North coordinate of current waypoint
$WP_{t_{east}}$	East coordinate of target waypoint
$WP_{t_{north}}$	North coordinate of target waypoint

$\bar{\chi}$	Air-vehicle average rate of turn
P_{A_n}	North coordinate of the aircraft position
P_{A_e}	East coordinate of the aircraft position
WP_{t_n}	North coordinate of the target waypoint
WP_{t_e}	East coordinate of the target waypoint
ψ_l	Loitering heading demand
C_p	Power coefficient
C_T	Thrust coefficient
J	Advance ratio

ABSTRACT

This project presents the developments of a vision-based system for aerial pipeline Right-of-Way surveillance using optical/Infrared sensors mountedon Unmanned Aerial Vehicles (UAV). The aim of research is to develop a highly automated, on-board system for detecting and following the pipelines; while simultaneously detecting any third-party interference. The proposed approachof using a UAV platform could potentially reduce the cost of monitoring and surveying pipelines when compared to manned aircraft.

To evaluate the performance of the system, the algorithms were coded using Python programming language. The pipeline endpoints are identified by transforming the 16-bits depth data of the explored environment into 3D point clouds world coordinates. Then, using the Random Sample Consensus (RANSAC) approach, the foreground and background are separated based on the transformed 3D point cloud to extract the plane that corresponds to the ground. Following that, these boundaries were filtered out, after being transformed into a 3D point cloud, based on the real height of the pipeline for fast and accurate measurements using a Euclidean distance of each boundary point, relative to the plane of the ground extracted previously. The filtered boundaries were used to detect the straight lines of the object boundary (Hough lines), once transformed into 16-bit depth data, using a Hough transform method. The pipeline is verified by estimating a centre line segment, using a 3D point cloud of each pair of the Hough line segments, (transformed into 3D). Then, the corresponding linearity of the pipeline points cloud is filtered withinthe width of the pipeline using Euclidean distance in the foreground point cloud. Then, the segment length of the detected centre line is enhanced to match the exact pipeline segment by extending it along the filtered point cloud of the pipeline.

The third-party interference is detected based on four parameters, namely:foreground depth data; pipeline depth data; pipeline endpoints location in the3D point cloud; and Right-of-Way distance. The techniques include detection, classification, and localization algorithms.

CHAPTER ONE: INTRODUCTION

1.1 Background

Hazardous transmission pipeline failures are an occasional occurrence but have the potential to cause major risks to population, properties and the environment, besides economic costs. In terms of the Nigerian constitution, all minerals, oil and gas in Nigeria belong to the Federal Government. Section 44(3) states that 'notwithstanding the foregoing provisions of this section, the entire property in and control of all minerals, mineral oils and natural gas in, under or upon any land in Nigeria or in, under or upon the territorial waters and the Exclusive Economic Zone (EEZ) of Nigeria shall vest in the Government of the Federation and shall be managed in such manner as may be prescribed by the National Assembly' (FRN, 1999).Armed gangs have carried out numerous attacks on oil facilities and kidnapped more than 500 oil workers and ordinary Nigerians for ransom during this period. The amnesty offer, announced in June 2009, followed a major military offensive in May against militants in the creeks of Delta State, which left scores dead and thousands of residents displaced. (HRW, 2010)

Though considered an act of sabotage by Petroleum Act and covered by the Criminal Justice Decree of 1975 (Miscellaneous provisions) and though oil pipeline sabotage is prohibited in section 1 of the Petroleum Production and Distribution (Anti-Sabotage) Act (Act 353 of 1990), the incidence of pipeline vandalization occasioned by third party activities has been on the increase. Shell Petroleum Development Company, SPDC, on behalf of the joint venture partners has incurred the majority of shut-in oil production (477,000 bbl/d), followed by Chevron (70,000 bbl/d) and Agip (40,000 bbl/d). Militant attacks on oil infrastructure have also crippled Nigeria's domestic refining capabilities. In February 2006, militant attacks in the western delta region forced the Warri (125,000 bbl/d) and Kaduna (110,000 bbl/d) refineries to shutdown due to a lack of feed stocks.

Major oil companies such as Chevron Nigeria Limited, Nigeria National Petroleum Company, Exxon Mobil Nigeria limited, Agip Oil Company have inspection programs for their pipelines and flow lines. Inspections are carried out on a scheduled with the aim of determining the remaining life of pipeline.

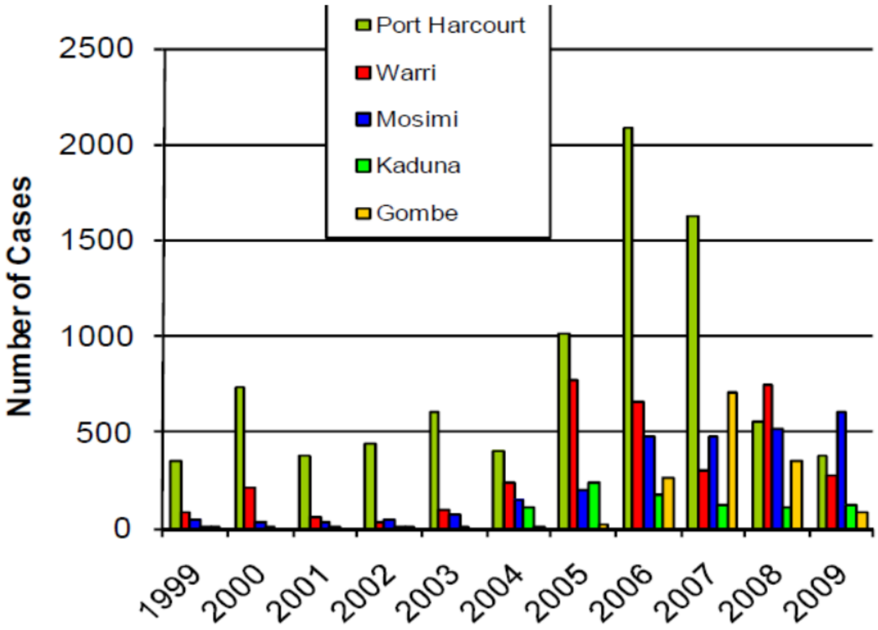


Figure 1.1 Pipeline Breaks by Zones, 1999 – 2009. (Source : NNPC 2009)

Reported cases of pipeline vandalization and associated fire disaster in Nigeria recently, include:

i. Petrol Pipeline burst in Ijegun, Lagos state caused by excavation equipment during a road construction killing over 15 people in June 2007

ii. Ruptured pipeline supplying crude oil to the 125,000 barrels per day to Warri Refinery and Petrochemical plant.

iii. Pipeline explosion at Abulu Egba in Lagos on December 26, 2006 killing over 500 people with a complete incineration of over 40 vehicles, a dozen homes including a mosque and two churches.

iv. Abule Egba pipeline explosion in Lagos, on December 26th, 2006 caused by oil scavengers and killing over 500 people.

v. Pipeline vandalization at Ijeododo, Lagos on December 2, 2006 killing atleast one person with its consequent environmental pollution and destruction of farmlands.

vi. Makaraba oil Pipeline, operated by Chevron Nigeria limited was shut down on May 11, 2005 after scavengers were scoping from pipelines network for illegal sales.

vii. Ilado village, Lagos pipeline explosion on May 12, 2006 killing over 150 people with dozens of others wounded affecting area of over 20 metre radius.

viii. January 13, 2006 at Iyeke village in Edo State where the death tolls was at least 7 people and injuring many.

ix. May 30, 2005 at Akinfo, Oyo State pipeline burst where one person died immediately with 15 dying later.

x. Pipeline fire in Imore Village, Lagos in December 2004 killing over 500 victims with the attendant environmental pollution.

xi. September 16, 2004 pipeline pollution at Ijagun, Lagos killing at least 60 people pollution.

xii. January 6, 2004 pipeline breakup at Elipokwodu in Rivers State where the causality runs into millions of naira.

xiii. Pipeline destruction at Onitsha Amiyi-Uhu in Abia State on June 19, 2003 killing over 125 people.

xiv. Shell pipeline ruptured on May 12, 2003 near Ubianuge village near the oil port of Warri, Delta State, evidently caused by vandals, streaming into large acres of farmlands.

xv. On October 15, 2002, fire erupted at Akute in Ogun State after series of attacks by vandals.

xvi. On September 2, 2002, a rupture in a Shell pipeline spread through creeks and streams spreading over 40 square meters.

xvii. January3, 2002, Nigerian National Petroleum Corporation (NNPC) shut down the main pipeline carrying crude from the Escravos terminal in the Niger Delta to two refineries after an explosion. The pipeline carries crude to the 120,000 barrels per day (bbl/d) Warri refinery in the southwest and 110,000 bbl/d Kaduna refinery in the northern region.

xviii. November 5, 2001 at Umudike in Imo State where pipeline vandals activities wrecked havoc on the community, destroying farmlands and killing atleast 3 people burning several bicycles.

xix. July 10, 2000 in separate incidents at Adeje/Egborode, Okpe and Oviri Court, both in Delta State killing with a combined casuality of over 500 damaging acres of farmlands and polluting the environment.

xx. On June 3, 2000, Adeje community in Delta State erupted with petroleum product damaging forest land and destroying high-tension power cables of electricity plants with a resultant clash between police and youth of the community.

xxi. June 8, 1999 at Akute Odo, Ogun State where over 15 people lost their lives.

xxii. At Jesse, October 17, 1998, a petroleum pipeline exploded, killing about 1200 villagers, some of whom were scavenging gasoline and damaging acres of farmlands.

Table 1.1: Oil Spills in the Petroleum Industry (1976-1996) in Barrels

Year	Number of spills	Quantity Spilled	Quantity Recovered	Net quantity loss to environment	Percentage of quantity loss to environment
1976	128	26157.00	7135.05	19021.50	72.72
1977	104	32879.25	1703.01	31176.24	94.82
1978	154	489294.75	391445.00	97849.75	20.00
1979	157	694117.13	63481.83	630635.95	90.85
1980	241	600511.02	42416.83	558094.19	92.94
1981	238	42722.50	5470.20	37252.30	87.72
1982	257	42841.00	2171.40	40669.60	94.03
1983	173	48351.30	6355.90	41995.40	86.85
1984	151	40209.00	1644.80	38564.20	95.91
1985	187	11876.60	1719.30	10157.30	85.52
1986	155	2905.00	552.00	12353.00	95.72
1987	129	31866.00	6109.00	25757.00	80.83
1988	208	9172.00	1955.00	7217.00	78.69
1989	228	5956.00	2153.00	3803.00	63.85
1990	166	14150.35	2092.55	12057.80	85.21
1991	258	108367.01	2785.96	105581.05	97.43
1992	378	51187.90	1476.70	49711.20	97.12
1993	453	8105.32	2937.08	5168.24	63.76
1994	495	35123.71	2335.93	32787.78	93.35
1995	417	36677.17	3110.02	33567.78	91.52

1996	158	39903.67	1183.81	38719.86	97.03
Total	4835	23823737	550,234. 19	1832189.49	

(SOURCE: DPR, 2000)

Figure 1.2: Oil Spill Impacts, Niger Delta Area (Ogri, 2001).

Figure 1.3a: Women drying garri using flames fron Shell's Utorogu flow Station in the NigerDelta. (George Esiri/Reutershttp://in.reuters.com/articl

Figure 1.3b: A local market located on top of crude Oil flowline row at a PortHarcourt suburb. (Awonusi, 2007)

Figure 1.3c:A young boy using oil pipeline as a foot bridge (Awonusi, 2007).

Figure 1.4: Various pictures depicting various acts of vandalization. (Eyo Essien, 2005)

Maintaining the integrity of oil and gas pipelines, that may run hundreds of miles through desolate terrain, is a highly demanding and manpower-intensive proposition. An unchecked leak can result in environmental disasters and costly disruptions to business. It is in the interest of any company to maintain the valueof its pipelines and guard them effectively against any damage caused by third- parties or any other defects.

Today, the pipeline corridors are surveyed by regular foot, vehicle patrols or using fixed-wing aircraft or helicopters. These patrols prevent developmentsand events, which could place the pipelines, the surroundings of pipelines or security of supplies at risk.

As a result of global progress in high-resolution remote sensing and computer vision technology, it is now possible to design a highly automated airborne surveying system with remote sensors and context-oriented image processing software, to carry out these types of surveillance and monitoring missions.

The main aim of this project is to develop a vision based low-cost aerial technology for monitoring and routine inspections of pipeline's rights-of-way.The technology is envisaged to compose a suite of sensors (for example Optical/Infrared, 3D LIDAR). In addition, computer vision techniques integrated

onboard the UAV platform, which comprise of; detection and visual tracking of the pipeline structure, anomalies detection algorithms and video data relay to the ground station. The development of these algorithms is performed in this research by designing and programming methods in terms of mathematical, physical and statistical functions for acquiring, processing, analysing, and understanding the images.

The developed system is envisaged to provide efficient and continuous support to pipeline infrastructures. The technology can be adjusted to the particular requirements to define the most suitable cost package depending on fuel, communication support, embedded sensors, etc. It is envisaged that the data could then be fused with other vital information (for example, internal erosion) obtained from other sensors and fed into the operating surveying system to define the status of the infrastructure, as well as of the surrounding area (for example environmental risks caused by leakages).

1.2 Project Aim & Objectives

The main aim of this research is to develop a vision based system for low- cost aerial technology for surveillance and routine inspections of lengthy pipelines and their Rights-of-Way. The system includes a suite of remote sensors (such as, Optical/Infrared, 3D scanner system) integrated on-board a UAV platform. In addition to that, a developed computer vision algorithms for image processing in real-time fordetection and visual following of the pipeline structure, anomalies detection, andvideo data relay to the ground station, were developed as part of the system.

The key objectives of this project are summarized as follows:

I. Develop a computer vision algorithm to estimate the position of the endpoints of the pipeline segments in the frames sequence in near real- time to automatically keep track/follow the pipeline.

II. Develop a vision-based pipeline auto tracking algorithm, able to operatewithout GPS, based on the detected pipeline endpoints information in real-time.

III. Develop a computer vision algorithm to detect anomalies and third- party activities in the vicinity of the pipeline structure and relay this information to the ground operator for visualisation and analysis.

IV. Integrate a video transmission datalink to transmit the data from the aerial platform to the Ground Control Station (GCS) with minimum latency..

1.3 Research Limitations

The focus of the project was to test the system for the surveillance and inspection of the over-ground pipeline structure. The surveillance and inspection missions of the pipeline structure and its Right-of-Way covered only third-party activates.

The remote sensor mounted on the system to provide the 3D visual data can only work indoor. A small-scale environment of pipeline structure and third-party activates were would be used for this project. Change in environmental factors e.g. sands, rainfall would not be considered.

CHAPTER TWO :LITERATURE REVIEW

2.1 Introduction

This chapter presents the review of related research and work that were carried-out in this project as part of the literature review. This review starts with an overview of a pipeline's integrity. Then, moves to the characteristics of the existing pipeline infrastructure. Following that, the current systems that have already been developed to monitor the pipeline, in general, are discussed. Then, the types of suitable aerial platforms that could be employed are described. The "second" part of the literature review covers existing computer vision techniques and applications. In addition, there is a brief survey of existing wireless communications systems and technology. Finally, a brief overview of the costs of current monitoring systems is covered at the end of the chapter.

2.1.1 Pipeline Integrity Overview

So, many companies have recorded many incidents throughout their pipeline's route. Their defects are analysed and reported quantitatively that were either because of physical or operational causes. The Pipeline and Hazardous Materials Safety Administration (PHMSA) (Baker, 2009) reported primary causes of the significant incidents of potentially hazardous liquid transmission pipeline with percentages between 1988 to 2008 as shown in Figure 2-1.

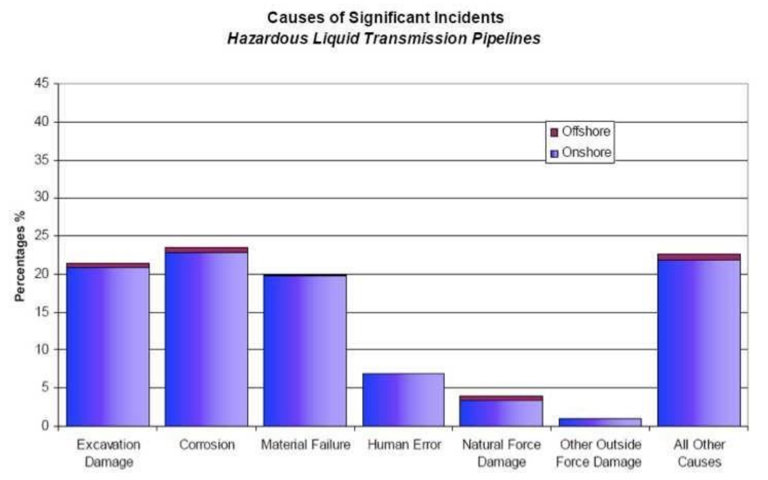


Figure 2-1: Causes of Significant Pipeline Incidents on Hazardous LiquidTransmission

2.1.2 Pipelines from 1988-2008 (Baker, 2009)

Because of these failures in the transmission pipelines, it is necessary to increase attention of public safety, also the direct cost of the failures affecting the pipeline operators, and lessen these risks as much as possible by surveying the integrity of these pipelines and their Right-of-Way. Based on these results obtained from (Baker, 2009; Hopkins et al., 1999), it is essential to find areliable technique that is capable of improving the integrity of these pipelines by early detection of third-party activities, or any leak already occurring in real-time and reporting their positions. So, some issues are required to be identified to help design such a system, which includes the necessary specifications of main pipeline networks (width, length, materials, surrounding environment, and so forth) and the possible methods that could be used to survey the pipelines and its corridor.

For pipelines that are deemed dangerous, such as those carrying gas and high- pressure oil, preventive measures to detect potential threats are more important than measures to detect real damages. In actual practice, dangerous pipelines are regularly patrolled by specialised personnel, either on foot, by vehicles, or even by helicopters. However, this kind of manual checking is laborious, economically expensive and is not seen to be efficient or necessarily useful. Therefore, there is an increasing necessity for automatic, continuous, and low- cost pipeline monitoring systems.

2.1.3 Pipeline Infrastructure Characteristics

Initially, some of the pipeline specifications are required to be known to design such a surveillance system. Based on the reports carried-out by (Canadian Energy Pipeline Association (CEPA), 2010; Hopkins, 2002; U.S. Energy Information Administration (eia), 2010), there are many types of pipelines utilised to transport oil and gas around the world which are listed below:

i. Flow lines & Gathering Lines — These short distance lines gather variety products in an area and move them to process facilities. They are usuallysmall diameter from 50 mm (2”) to 305 mm (12”).

ii. Feeder Lines - These pipelines move the oil and gas fluids from processing facilities, storage, and so forth, to the main transmission lines.They can be up to 508 mm (20”) in diameter.

iii. Transmission Lines — These are the main conduits of oil and gas transportation as shown in Figure 2-2. They can be very large in diameter as, for example, in Russia, where they are around 1422 mm (56”) in diameter; or very long, such as in the case in the USA’s liquid pipeline system that is over 250,000 km (155,000 miles) in length. Crude oil transmission lines carry different types of product, to refineries or storage facilities.

iv. Product Lines - Pipelines carrying refined petroleum products from refineries to distribution centres are called product pipelines.

v. Distribution Lines - These allow local, low-pressure, allocation from a transmission system. Distribution lines could have in some cases large diameter, but most are under 152 mm (6”) diameter.

Figure 2-2: Transmission Pipeline in the Americas (Hopkins, 2002)

2.1.4 Pipeline Monitoring Systems

Several Systems are currently available or under development for observing and reporting third-party interference and activity in the vicinity of the pipeline. Some of them were already reviewed by (Burkhardt and Crouch, 2003), or reporting leaks from pipes as in (Murvay and Silea, 2012). These methods are classified in this review into aerial or built-on/in monitoring systems, within pipeline or Right-of-Way infrastructures.

2.1.4.1 Built-On/In Monitoring Systems

The built-on/in surveillance systems are technologies utilized for ground surveys of pipelines integrity, which are performed inside the pipeline or on its surface or its district. These methods include fiber optic (Huang et al., 2007), evanescent sensing (Culshaw and Dakin, 1996), acoustic sensing (Jin andEydgahi, 2008; Park et al., 2007), Cathodic Protection (CP) (Joseph and Winslow, 1982), Impressed Alternating Cycle Current (IACC) (Anupama et al., 2014), flow monitoring (Murvay and Silea, 2012), and Software based dynamic modelling (Murvay and Silea, 2012).

The fibre optic system is typically utilized to generate a broadband acoustic signal to detect leaks or third-party intrusion. This acoustic pressure will induce an optical phase signal in the parallel optical fibre fixed on the surface of the pipes. The system can detect and locate several leaks or intrusions along the pipeline at the same time. On the other hand, this technology requires particular and complicated installation to enhance sensitivity in the pipeline (leak) and the Right-of-Way (intrusions).

The evanescent sensing is a kind of on-pipe monitoring system. It is an optical fibre buried along with the pipe to sense and monitor the local changes of pressure or concentration using lasers and optical detectors. Once natural gas leaks, the local pressure or density of the gas will change, which leads to change in the transmission characteristics of the optical fibre.

Acoustic or sonic monitoring system is a non-destructive method, which typically analyses pressure measurements throughout the pipeline. If there is a region at which a noise is generated, it indicates a leak. The technology could detect the location of any leak as well as third-party activities. However, to monitor long pipelines, many sensors need to be installed. One downside of thistechnology is that it has high false alarm rates when detecting small leaks.

Cathodic Protection (CP) monitoring system can be used for detection of corrosion, leaks, and third-party contacts. This technology works by measuring the variation in the current paths of cathodic protection once any interference is made. The detection range is short; and again it has a high false detection dueto breaches in pipe coating and the influence from 60 Hz (and harmonics) signals from other sources.

The IACC technology consists of impressing electrical signals on the pipe by generating a time-varying voltage between the pipe and the soil at alternate locations where pipeline access is available. It has the same advantages and disadvantages of the CP method.

Flow monitoring devices measure the rate of change of pressure or the mass flow at different sections of the pipeline. If the rate of change of pressure or the mass flow at two locations in the pipe differs significantly, it could indicate a potential leak. The major advantages of the system include the small cost of the system as well as non-interference with the operation of the pipeline. The two disadvantages of the system include the inability to pinpoint the leak location, and the high rate of false alarms.

Software based dynamic modelling monitors various flow parameters at different locations along the pipeline for leak detection. These flow parameters are then included in the model to determine the presence of natural gas leaks in the pipeline. The major advantages of the system include its ability to monitor continuously, and also the non-interference with pipeline operations. However,

dynamic modelling methods have a high rate of false alarms and are expensive for monitoring an extensive network of pipes.

In general, the built-on/in systems are very effective in terms of, cost and effort for installation and maintenance and have difficulties in dealing with ageing and long pipelines and complex terrains.

2.1.4.2 Aerial Monitoring Systems

The aerial monitoring systems are a remote sensing technique that can automatically provide a real-time surveillance of pipelines and Right-of-Way at a distance; delivering optical, acoustical, or microwave information. These technologies have been widely used in the aerial surveillance field for over a decade now. As quoted by (Fung et al., 1998), the National Energy Board and the gas pipeline industry concluded, after several studies, that current remote sensing technologies have the potential to be applied to pipeline Right-of-Way monitoring. A project was launched to identify where the use of remote sensing may provide cost-effective solutions and reduce the current amount of ground surveys. However, some key elements are required to develop and build an aerial monitoring system, which include:

i. Sensors (Hardware);
ii. Platform to carry the payload and perform the mission (Hardware);
iii. Data processing (Hardware & software);
iv. Communication technology for data rely (Hardware & software).

The remote sensing techniques are classified, based on the sensor used, into passive or active (Schowengerdt, 2006). The active sensor uses its energy source to illuminate the target and measure the reflected radiation. For leak detection, the active sensor illuminates the area around the pipeline with a laser or a broadband source. Monitoring the absorption or scattering that are caused due to the presence of substance molecules around the surface can be performed using a set of sensors operating at specific wavelengths. If there is a notable absorption or scattering around the pipeline, then a leak could potentially exist. In recent years, many active monitoring systems have been developed to detect leaks from pipelines; these include, Tunable Diode Laser Absorption Spectroscopy (TDLAS) (Zhang et al., 2009), Laser InducedFluorescence (LIF) (McStay et al., 2007), Coherent Anti-Raman Spectroscopy (CARS) (Eckbreth, 1977), and Fourier Transform Infrared Spectroscopy (FTIR) (Mahamuni and Adewuyi, 2009), diode laser absorption (Iseki et al., 2000), millimeter Wave (mmW) Radar systems (Gopalsami and Raptis, 2001), backscatter imaging (McRae and Kulp, 1993; Spoonhower et al., 2006), broadband absorption (Spaeth and O'Brien, 2003), and LIDAR systems (Owechko et al., 2010; Prasad and Geiger, 1996; Roper and Dutta, 2005; Tao and Hu, 2002).

Diode laser absorption uses the same technology as LIDAR with the crucial difference being that diode lasers are used instead of the more expensivepulsed lasers. However, one downside is that if only a single wavelength is used, the system can be prone to false alarms since the laser can be absorbed equally well by dust particles.

Broadband absorption systems utilize low-cost lamps as the source; thus, significantly reducing the cost of the active system. Monitoring is conducted at multiple wavelengths, so that the system is less prone to false alarms.

Millimeter wave radar systems have been used to detect the chemicals used in the pipeline leak detection system itself. It measures the variation of scattering properties of the radioactive substance leaks around the pipeline, which could potentially indicate a leak.

Backscatter imaging utilizes a carbon-dioxide laser to illuminate the area above the pipeline. They are mainly used in natural gas pipelines, since the gasscatters the laser light very strongly. This scattered signature is imaged usingan infrared imager or an infrared detector in conjunction with a scanner. LIDAR systems typically employ a pulsed laser as the illuminating source. It measures the molecules' energy absorption at specific wavelengths in the electromagnetic spectrum.

All the active systems described previously use a source and obtain either transmitted or scattered images to determine the presence of a leak. These systems can be mounted on moving vehicles, air-vehicle or on-location. The advantages of these systems include the capability to monitor over an extended range. Furthermore, the ability to monitor third-party interference or leaks even if there are no differences in temperature between the liquid and the surroundings. Also, under specific conditions, these techniques have a high spatial resolution and sensitivity. However, these systems still have high, false rate alarms.

Passive monitoring systems are similar to active ones in many aspects. However, the major difference between active and passive techniques is that passive techniques do not require a source. Either the radiation emitted by the natural gas or the background radiation serves as the source. It makes passive systems less expensive in some respects. However, since a strong radiation source is not used, far more expensive detectors and imagers have to be used with passive systems.

The two major types of passive systems used for monitoring leaks from natural gas pipelines are thermal imaging (Kulp, 1997; Weil, 1993) and multi- wavelength imaging (Althouse and Chang, 1995; Bennett et al., 1996; Marinelli and Green, 1996; Smith and Laubscher, 1999).

Thermal imaging detects natural gas leaks from pipelines due to the differences in temperature between the natural gas and the immediate surroundings. This method can be used in moving vehicles, helicopters or portable systems andcan cover several miles or hundreds of miles of pipeline per day. Usually, expensive thermal imagers are required to pick up the small temperature differential between the leaking natural gas and the surroundings. Also, thermal imaging will not be effective if the temperature of the natural gas is not different from that of the surroundings.

Multi-wavelength or hyperspectral imaging can be accomplished either in absorption mode or emission mode. For obtaining gas concentrations utilizing multi-wavelength emission, the gas temperatures have to be much higher than the surrounding air. Multi-wavelength emission measurements have been typically used in the past to obtain single point concentrations in hot combustion products (Sivathanu and Gore, 1991; Sivathanu et al., 1991).

Multi-wavelength absorption imaging utilizes the absorption of background radiation at multiple wavelengths to directly image the gas concentration, even in the absence of temperature gradients between the gas and the surrounding air. This technique has been used to monitor natural gas leaks in industrial settings very successfully. The advantage of employing multi-wavelength passive systems is that they are relatively immune to false alarms, and can be utilized for remote monitoring without being constantly watched over. The only drawback, however, of multi-wavelength or hyperspectral imaging is that it typically utilizes very sensitive sensors which are expensive.

The proposed approach in this project is similar to the use of an imagerysystem, such as the Light Detection And Ranging (LIDAR) system. A commercial manned air-vehicle monitoring pipeline Right-of-Way could need relatively minimum modifications for the installation of such a system on the aircraft. On operation, it could reveal any JCBs, trucks or other earth-moving equipment in the vicinity of the pipeline (Fung et al., 1998; Randell, 2010). The advantage of the proposed system is that it can detect the pipeline structure and Right-of-Way in real-time, provide the location of defects due to a third-party and reduce the amount of data transmission, i.e. required bandwidth. Thereafter, it can use computer vision techniques to detect and classify any object in the pipeline surrounding environment. The other advantage of this technology (Shaochuang et al., 1999), is that it has an active laser pulse that is not influenced by the shadow angle of the sun which, in turn, reduces their influence on data acquisition. Moreover, when compared with Photogrammetry, it avoids loss of information when transformed from 3D to 2D, has an accurate elevation; has multi-beam echo acquisition to get high-density data, is suitable for a high level of automation, and has real-time capability to produce a digital elevation model (DEM).

To emulate the LIDAR system, the sensor proposed to be used in this project is

"ASUS Xtion Pro Live". It uses a projected, structured light pattern to estimate the depth from the sensor to the view of the environment. The depth sensor is proposed to optimize the description of objects based on elevation contrast.

2.2 Aerial Platforms

One of the main, aerial monitoring elements is the platform itself that will bear the sensor instruments. Figure 2-6, shows some common platforms used for aerial monitoring, for example, air-vehicle, balloons, satellites, spacecraft, probes, rovers, launch air-vehicles, etc. (NASA tutorial on remote sensing, 2011)

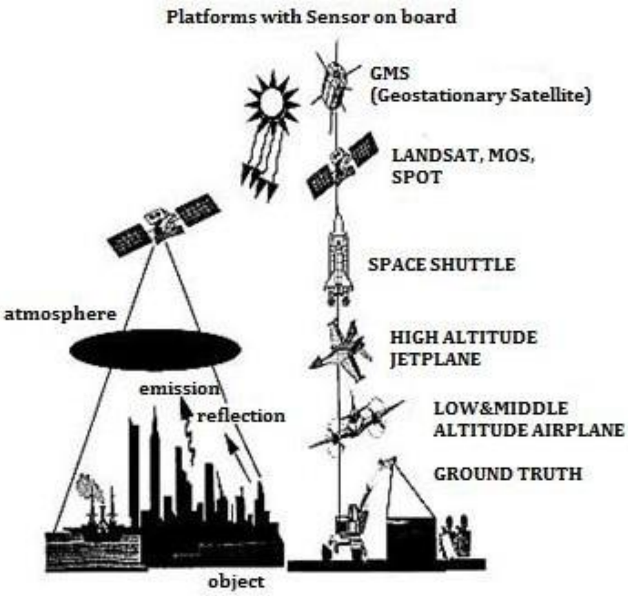


Figure 2-3: Common remote sensing platforms (NASA tutorial on remote sensing,2011)

Nowadays, the Unmanned Aerial Air-vehicle (UAV) is an emerging technology being adapted for use in a wide range of applications in the field of remote sensing, and monitoring. The UAV can be remotely operated or, for more complicated and expensive systems, fly autonomously based on pre- programmed flight paths (Witayangkurn et al., 2011). This type of dull andrepetitive mission makes UAVs an attractive solution for monitoring the length of pipeline routes and detecting any threats or leaks around it.

According to (Zongjian, 2008), the limitation of using satellites is due mainly to the high launch/flight costs, slow and weather-dependent data collection, restricted manoeuvrability, limited availability, flying time, and lower ground resolution. On the other hand, the UAV could operate relatively close to the pipeline, which in turn means it needs to use less expensive sensors to achieve the same high-resolution image as a satellite system. Also, it could provide real-time information and can cover the rights-of-way of a large segment of the pipeline quickly and efficiently (for example, when compared to foot patrols) (Roper and Dutta, 2006).

Figure 2-4: [Left] Breguet-Richet Gyroplane No.1, 1907 [Right] Quadrotor (Devaudet al., 2012)

2.2.1 Computer Vision Techniques

Many security and surveillance systems, that have already been developed and built, rely on the use of computer vision algorithms. Using image processingand analysis techniques, these monitoring applications automatically and remotely generate intelligent and useful descriptions of the monitoring environment (Koo et al., 2012; Nagai et al., 2009; Piciarelli et al., 2013). The descriptions in the computer vision field are interpreted based on features, boundaries, regions, and 3D models. Each technique is proposed based on the required description. However, in this project, four interpretations are needed to achieve this goal, namely: extraction and tracking of the pipeline structure, third- party interference identification near the pipeline route, and leak detection.

2.2.2 Pipeline Detection & Tracking

For autonomous control purposes, several of the proposed methods are based on GPS (Dillman, 2002; Hasan et al., 2009; Low and Wang, 2008), vision (Dobrokhodov and Kaminer, 2006; Frew et al., 2004; Mondragón and Campoy, 2010; Mondragon et al., 2007; Rathinam et al., 2005), Digital Elevation Map (DEM) (Collins et al., 1998; Kanade et al., 2004; Nagai et al., 2009; Navarro-Serment, 2010), and a fusion of data (Du and Teng, 2007; Sohn et al., 2008). It is known that the GPS estimates the target position in the global frame while the other approaches estimate the position locally; therefore, some of the proposed approaches can follow the target objects locally as waypoints, landmarks or paths/trajectories.

In this project, the target required to be tracked is the pipeline, so in the computer vision field tracking the pipeline needs to be described first then its position can be localised. Due to the nature of the pipelines, the robust description that could be used as an indicator of pipeline structure is the linear segment. A number of vision-based techniques have been developed to extract this feature, such as combining the points of edges into lines (Cook and Delp, 1998; Nevatia and Ramesh Babu, 1980), Hough Transform (HT) (Agrawal et al.,1996; Antolovic, 2008; Gonzalez and Woods, 2003; Illingworth and Kittler, 1988), Random Sample Consensus (RANSAC) (Behrens et al., 2003; Nistér, 2005), vanishing point (Rathinam et al., 2008; Schindler et al., 2006; Tsai,Chang and Chen, 2006; Wang et al., 2004), learning algorithm (Rathinam et al., 2005), spline model (Wang et al., 2000).

Detecting pipelines by linking edge points can be achieved by determiningpositions and orientations of these points, then approximate the pipeline bypiecewise linear segments. However, it is a weak method to detect pipelines with the line because sometimes it is difficult to differentiate in colour.

Hough Transforms (HT) is one of the robust methods for finding the pipeline structure in an image. Once the feature points in the image space are detected, converting them into parameter space by using Hough transform can be done using the approach as shown in Figure 2-9. Then, the peak point in parameter space is evidence that there might be a line passing through that point that can be tuned by a number of points threshold.

(pixel)

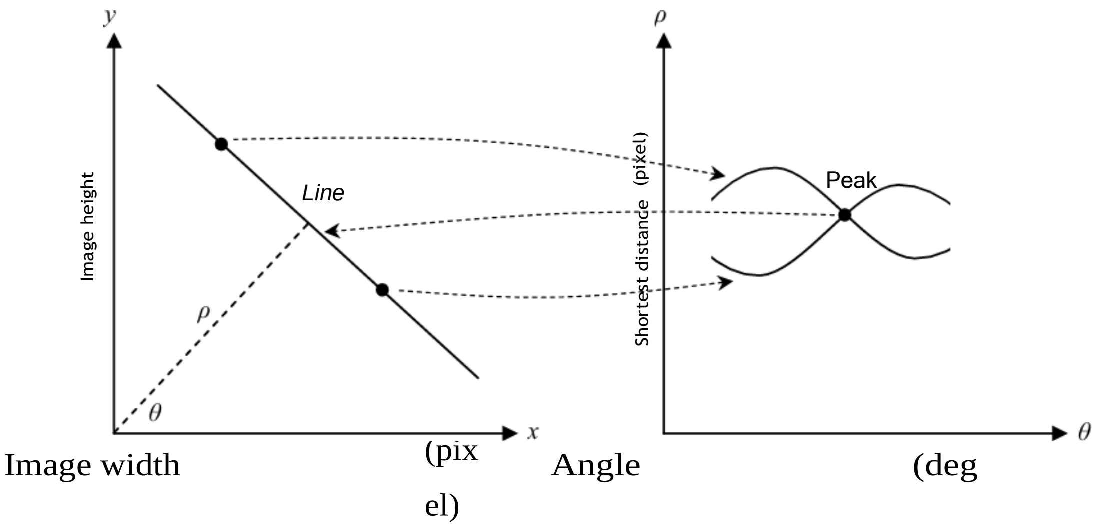


Figure 2-5: [Left]: Image Space, [Right]: Parameter space or Hough space (Antolovic, 2008)

RANSAC is a voting algorithm that uses the minimum number observations (data points) required to estimate the underlying model parameters (Fischler and Bolles, 1981). It works by selecting a random sample of the minimum required size to fit the pipeline model, then it computes a putative pipelinemodel from the sample set. Following that, it detects the set of inliers to this pipeline model of the whole data set and loops the same process until a model with the most inliers over all samples is found. It is a robust method for detecting pipelines. However, computational time grows quickly with a fraction of outliers and the number of parameters; moreover, it is not suitable for detecting multiple inlier structures. The vanishing point is the candidate point based on the probability of at least two perspective projections of intersected line segments which are, in 3D, mutually parallel. The advantage of this approach is that it accurately detects the 3D linear structure. However, it is computationally complex and intensive.

Learning algorithms could be used to detect the pipeline using off-line modellinga cross-section profile of the pipeline at one raw of pixels and detect it by using the profile matching algorithm in real time. However, the algorithm fails with noisy boundaries of pipelines which is an issue considering that pipelines are built/found in different environments and terrains.

Spline modelling describes the prospective effect of parallel line boundaries of the pipeline by modelling piecewise polynomials of degree n with function values and the first n-1 derivatives that agree at the points where they join. Although it can detect pipeline structure, it cannot describe it in the case of any noise or broken boundaries line points; and, it also needs extra computations.

As a result, the proposed approach for this project to describe the pipeline structure as a combination of Hough transforms and RANSAC algorithms. These methods are robust and require lower computation than the others. The Hough transforms are used to describe the boundaries of the pipeline and RANSAC to emphasize that the boundaries are modelling the pipeline structure.

2.2.3 Third-Party Interference Monitoring

One of the main monitoring tasks is the detection and warning of any third- party, interference activities near the pipeline route in real-time. Since many sources of intrusions that potentially exist in the vicinity of the pipeline which include personnels, cars, trucks, and so forth, the proposed sensing techniqueis based on optical/point cloud data since the point cloud was already used in detecting the pipeline in the previous step. However, based on the point cloud data, there is different computer vision algorithms, can be used in this project to determine the regions of interest, then analyse the synchronized optical data at that region to detect the targets such as third-parties interference.

Many algorithms have been developed to reconstruct a 3D point cloud modelfor segment regions of interest such as 3D region growing (Jarząbek-Rychardet al., 2010; Revol-Muller and Peyrin, 2000), 3D Hough transform (Borrmann et al., 2011; Tarsha-Kurdi, 2007), 3D recognizing structure (Vosselman and Gorte, 2004), and 3D RANSAC (Behrens et al., 2003; Huang et al., 2011; Kanade et al., 2004; Nagai et al., 2009; Rathinam et al., 2005).

A 3D, region growing, segmentation algorithm works on the principle of merging all neighbouring pixels to satisfy a homogeneity criterion starting from an initial set of seeds. The first process in an automated system is the unsupervised location of the seeds which is based on prior-knowledge of anatomical structures, such as humans, cars, trucks, and so forth; and their typical tracer uptake. The outcome of this data is taken into account to calculate certain parameters required for locating the seeds. This method is suitable to fit the high variability of the tracer uptake. Since both local and global parameters are taken into account in the merging process, this type of algorithms are competationally demanding.

The 3D Hough transform algorithm is often used to segment the 3D models of objects in point cloud sets. It is achieved by detecting the parameters of straight line segments, circular or elliptical cross sections that are subsequently tracked through the 3D point cloud. It maps the input sets of point cloud into zero- dimensional point sets in parameter space whose maxima represent object candidates. This approach is robust for 3D-dimensional parametric object detection. For a relatively large number of parameters it leads to extra space and time complexity.

A 3D recognizing structure algorithm has been used for object segmentation based on point cloud sets. This approach requires a predefined structure cue in the 3D point cloud set that is indicative of the object categories present in the scene. Data training is then carried-out to match those cues for semantic segmentation.

A 3D RANSAC algorithm has been used to detect basic shapes in random point clouds. The algorithm decomposes the point cloud into an inliers specific structure of inherent shapes and outliers random residual points. Each detected inliers point serves as a representation of a set of corresponding points of the object shape. However, this algorithm is robust even in the presence of many outliers' points and in the presence of a high degree of noise. Moreover, this approach has lower computational load, compared to the other methods discussed previously.

Hence for this project, the proposed method to segment objects in the vicinity of pipeline routes will be based on RANSAC. It scales well to the size of the input point cloud, the number and size of the shapes within the data. Point sets with large samples are robustly decomposed within a reasonable computational time. Moreover, the algorithm is conceptually simple and easy to implement. Application areas include measurement of physical parameters, scan registration, surface compression, hybrid rendering, shape classification, meshing, simplification, approximation and reverse engineering.

Since it is difficult to classify the third-party activities using only point cloud data,the combination with optical data is proposed in this project. In computer vision there is a variety of algorithms, which have been proposed that are capable of assigning the segmented targets from the derived inliers reconstructed 3D models, using the alignment optical image data to obtain the desired category inorder to consume the processing time and reduce the resulting classes. These supervised Machine Learning algorithms deal with single pixels or group of pixels in a segmented area. They include parallelepiped classifier (Rahman and Afroz, 2013), maximum likelihood classifier (Abkar et al., 2000), decision tree classifier (Pal and Mather, 2001), minimum distance classifier (Haala and Brenner, 1999; Wacker and Landgrebe, 1972), and multiple trained cascaded Haar classifier (Breckon and Barnes, 2009; Gaszczak et al., 2011).

The parallelepiped classifier is one of the most commonly used supervised classification algorithms for multispectral images, where the threshold of each class is defined in the training data to determine whether a given pixel is within the class or not (as shown in Figure 2-10). The interest region for each category is defined by a minimum and maximum pixel value on each axis of the segmented region. The accuracy of the classification depends on the selection of the minimum and maximum pixel values in lieu of the population statistics of each class. In this respect, it is very important that the distribution of the population of each category is well understood. This classifier is simple and easy to run having to make fewer assumptions regarding the character classes. In addition, the processing time will be minimum when compared to other classifiers. However, the accuracy will be low, especially when the distribution infeature space has covariance or dependency with oblique axes while the parallelepipeds are rectangular, which leads the classes to overlapping.

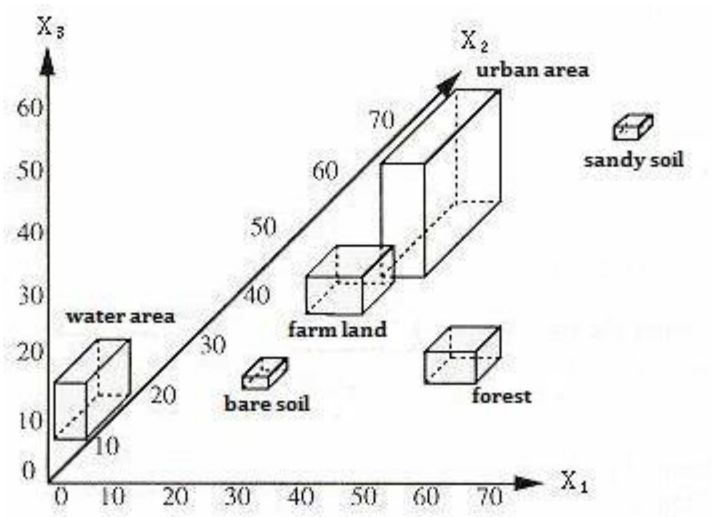


Figure 2-6: Semantic concept of parallel piped classifying in 3D feature space(Rahman and Afroz, 2013)

The decision tree classifier is a multistage based classifier that compares the data with a range of properly selected features to break up a complex decision into simpler decisions. The selection of features is determined by anassessment of the spectral distributions of separability of the classes. The procedure of this method, in general, involves the following steps: first splitting nodes, then determining which nodes are terminal; finally class labels are assigned to those terminal nodes based on a majority vote when it is assumed that certain categories are more likely than others. This classifier requires lower computing time than the other classifiers and, by comparison, the statistical errors are avoided. However, the accuracy depends completely on the design ofthe decision tree and the selected features.

The minimum distance classifier is used to classify new image data to classes that minimize the distance between the image data and the class in multi- feature space as shown in Figure 2-11. The distance is defined as an index of similarity so that the minimum distance is identical to the maximum similarity. The distances which are often used in this method include Euclidian distance when the variance of the population classes is different to each other; Normalized Euclidian distance when there is a difference in variance; and Mahalanobis distance when there is a correlation between axes in feature space.

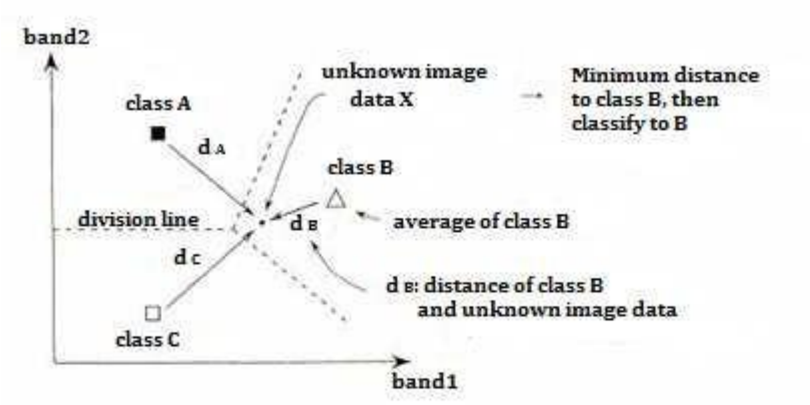


Figure 2-7: Concept of minimum distance classifier (Rahman and Afroz, 2013)

The maximum likelihood classifier is another popular method of classification in computer vision and remote sensing. It is based on a statistical decision criterion to assist in the classification of overlapping signatures where a pixel with the maximum likelihood is classified into the corresponding class of highest probability. When the variance-covariance matrix is symmetric, the likelihood is the same as the Euclidian distance, while the determinants are equal to each other; the likelihood becomes the same as the Mahalanobis distances. The maximum likelihood method has the advantage of being more accurate methods, however, at the expense of extra computation processing time.

The multiple trained cascaded Haar classifier algorithms have been used in object detection and recognition applications. This classifier was originally developed for face detection, however, it is not restricted to just detecting faces. Therefore, as described by (Bradski and Kaehler, 2008), it can be used to classify any moving or stationary object that is typically rigid and has blocky features that make it distinct. Based on the work published by (Monteiro et al., 2006) to detect pedestrians, it proves that this classifier is a robust and rapid approach for object detection. It forms a copulative set of weak classifiers into a strong classifier. Each weak classifier uses rectangular areas, called Haar-like features as shown in Figure 2-12, to compare with the trained features of the desired object in a given orientation in the image. The Haar feature values are computed as the sum of differences between differing rectangular sub-regionsat a localized scale which, although limited in scope as individual features, can be computed extremely efficiently. Individually, they are weak discriminative classifiers, but when combined as a conjunctive cascade, a powerful discriminative classifier can be constructed, capable of recognizing common structures over varying illumination, base colour and scale.

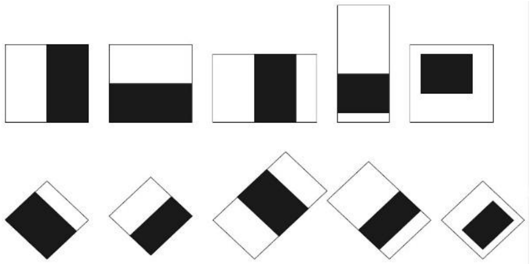


Figure 2-8: Types of Haar-like features (Viola and Jones, 2004)

The classifier is trained using a set of a few hundred positive and negative objects training images (separate set of positives and another for the negatives). The use of boosting techniques then facilitates classifier training to select a maximal discriminant subset of these Haar-like features, from the exhaustive and over the complete set, to act as a multi-stage cascade. In this way, the final cascaded Haar classifier consists of several key simpler (weak) classifiers that all forms a stage in the resultant complex (strong) classifier. These simpler classifiers are essentially degenerative decision-tree classifiers that take the Haar-like feature responses as input to the weak classifiers and return a Boolean pass/reject response. A given region within the image must then achieve a pass response from all of the weak classifiers in the cascade to be successfully classified as an instance of the object that the strong overall classifier has been trained upon. The classifier is then evaluated over a query image at multiple scales and multiple positions using a search window approach. Despite this apparent exhaustive search element of the classifier, the nature of the cascade (sorted in order of most discriminating features) allows earlier rejection of the majority of such windows with a minimal evaluator (and hence computational) requirement. In this way, the Haar cascade classifier thus successively combines more complex classifiers in a cascade structure which eliminates negative regions, as early as possible, during detection but focuses attention on promising regions of the image. This detection strategy dramaticallyincreases the speed of the detector, provides an underlying robustness to changes in scale and maintains achievable real-time performance. Moreover, it is capable of detecting both static and moving objects within the scene.

The algorithm that is proposed in this project to identify and recognize third- party interference is Haar classifier. Where each one classifies specifically trained object features of the known third-party activities at different categories of orientations and then uses a matching algorithm in the test image.

2.2.4 Oil Leak Monitoring

Another important factor that could help with increasing the integrity of the pipeline route is a vision based real-time, oil leak detection system. Leaks, which may be caused by any defects on the pipeline, can be detected using RGB imaging. The leak is detected as a slick of petroleum around the pipeline. In the optical image scene, the visual descriptions of this oil slick could be identified, in general, by using a surface texture or feature colour variation. However, these may benefit from other cues such as shape or time coherence.The surface texture is defined by three characteristics of surface roughness, waviness, and form. It could be used to describe the surface properties of an oil slick. Many applications have been developed to describe the surface texture in computer vision. They include road texture (Paquis et al., 2000), human skin texture (Fiedler et al., 1995; Kenet et al., 1998), fabric texture (Kumar, 2008), and composite material texture (Chen et al., 2011).

Colour variation is based on the wavelength variance of the reflected, emitted, or transmitted light. This information could be used efficiently to extract the oil slick from the surrounding pipeline. In computer vision objects can be identified by using colour variation analysis, which is much faster than using textureanalysis. Again in the literature, different applications can be found that addresses how to extract the objects based on the colour variation; such as roads (He et al., 2004), human skin (Hjelmås and Low, 2001; Kovac et al., 2003), and fires (Ahuja, 2004).

In this project, the same approach of classification of the colour variation to extract the oil slick around the pipeline is proposed.

2.2.5 Wireless Communications

Today, wireless communication has become the most popular method for data and information exchange technology. It provides a real-time data transmission over a distance without the need to use any wires or cables. The data from the platform is received at the ground control station in near real-time. Recently, many standard Wi-Fi technologies have been developed (Toshihiro, 2010) to transmit and receive the data wirelessly using different standards such as IEEE 802.11b, IEEE 802.11g, and IEEE 802.11a. Some factors were taken into account in this project when it came to selecting the most efficient Wi-Fi method. These include coverage range, transmission speed, and power consumption to trade-off between them. As shown in Figure 2-13, the coverage ranges vary from 50 to 150 meters. The rate of data transmitted (bandwidth efficiency) range is from 1.6 Mbps to 54 Mbps.

The proposed Wi-Fi method in this project is IEEE 802.11g. This method is an extension to IEEE 802.11b. It transmits the data at 54Mbps and covers up to150 meters within 2.4 GHz. It has lower power consumption than the other, and the devices are physically smaller. It is also worth mentioning that, it is cheaper than the other options.

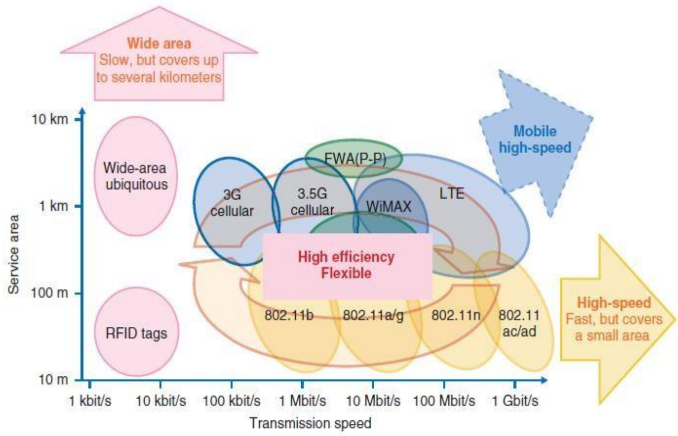


Figure 2-9: Directions of wireless technologies (Toshihiro, 2010)

Pipeline Monitoring Costs

The cost-benefit examples are considered based on the Gasunie pipeline system (Palmer, 2002). The study, which was sponsored by seven companies, shows that using satellites for surveillance costs approximately 3.5 times as much as using helicopters (manned air-vehicle), for obtaining the same benefit, as shown in Table 2-1.

Table 2-1: A summary of cost-benefit comparison result (Palmer, 2002)

	Satellite	**Helicopter**
Frequency of survey (days)	14	14
Basic cost per km ($)	16	3.5
Fraction of activities detected	0.41	0.39
Cost benefit ($)	6,336,221	1,771,411

CHAPTER THREE: GENERAL SYSTEM OVERVIEW AND EXPERIMENTAL SETUP

This chapter presents an overview of the vision-based aerial pipeline surveillance system structure. This system was proposed based on the research requirements and limitations which are also described. In general, the architecture of this system includes both the hardware and software components. These components are distributed into three parts; namely, the aerial platform, Ground Control Station (GCS), and the test-rig. The aerial platform is proposed to carry out the hardware components, which also includes the embedded systems board to run the software that was developed to carry- out the surveillance mission. The Ground Control Station (GCS) is also proposed to assist in performing the pipeline structure surveillance mission by executing part of the image processing algorithm onsite. Finally, the test-rig is used to validate the system in real-time running both the pipeline detection and monitoring algorithms along with third-party detection and classification. The proposed hardware components in each sub-system are described in detail, including the specifications and the general layout of these main parts.

3.1 System Requirements and Limitations

The Unmanned Aerial Vehicle (UAV) is proposed to be used as the primary aerial platform for monitoring and surveying the pipeline structure in this project. However, this platform must be able to perform the following tasks along the pipeline Right-of-Way (when it is scaled in the test-rig setup, it is assumed to be around 10 m on both sides of the pipeline centreline).

a) Identifying the endpoints of the pipeline structure, in real-time, based on the visual depth information, relative to the depth sensor frame. To localize the pipeline segment endpoints, online for the purposes of auto tracking and detecting the objects in vicinity of the pipeline.
b) Tracking the pipeline structure without using a GPS, in real-time, based on the localization of the pipeline segment endpoints to keep monitoring the route of the pipeline structure.
c) Detecting the third-party activites within the Right-of-Way of the pipeline route, in real-time, based on the localization of the pipeline segment endpoints to monitor the pipeline structure from any threat inthe vicinity.

d) Relaying the data into the ground station in near real-time to alert the ground operator of any issues/third-party interference.

3.1.1 The Operational Requirements

The operational requirements of the system are listed below:

1) Feature identification and detection should be at sensitivity andspecificity rates of at least 75%.
2) Feature identification and detection should have false negative andfalse positive rates of no more than 25%.
3) Localization accuracy of the features should be within a few meters atfull scale.
4) 10 m data coverage is required at each side of the pipeline corridor.
5) It must be able to monitor the pipeline infrastructure at least once aday.
6) Data transmission must be in near real-time.

3.1.2 Limitations

The selection of the complete system (i.e. ground and air) components and equipment was made taking into account the limitations and constraints listed in Table 3-1.

Table 3-1: Hardware constraints

Objectives	**Measure for effectiveness of solution**
Max payload	~ 1000 g
Operation	Indoor at low-level (1 m – 4 m)
Power capacity	~ 4900 mAH
Flight time	A total flight time of at least 10 minutes for thequadcopter with the entire required payload.
Pipeline structure	5 cm overground 3-5 m length, 1.2 cm width. (1m:50m scale)
Communication range	~ 10 meters for indoor test purpose
OS support	Linux Ubuntu 10.10, X86, 32/64 bit
Depth Image	VGA (640x480): 30 fps
GCS-PC processor	Intel Core i5 2.4 GHz processor and 8 GB RAM memory.
Embedded processor	Quad-Core ARM® Cortex A9 processor at 1GHz RAM memory of 1GBytes of 64-bit wide DDR3 @ 532MHz

3.2 System Architecture

This section presents the proposed high-level system architecture of the visual-based, pipeline structure surveillance as shown in Figure 3-1. This architecture is divided into three parts, which are the aerial platform, ground control station, and router.

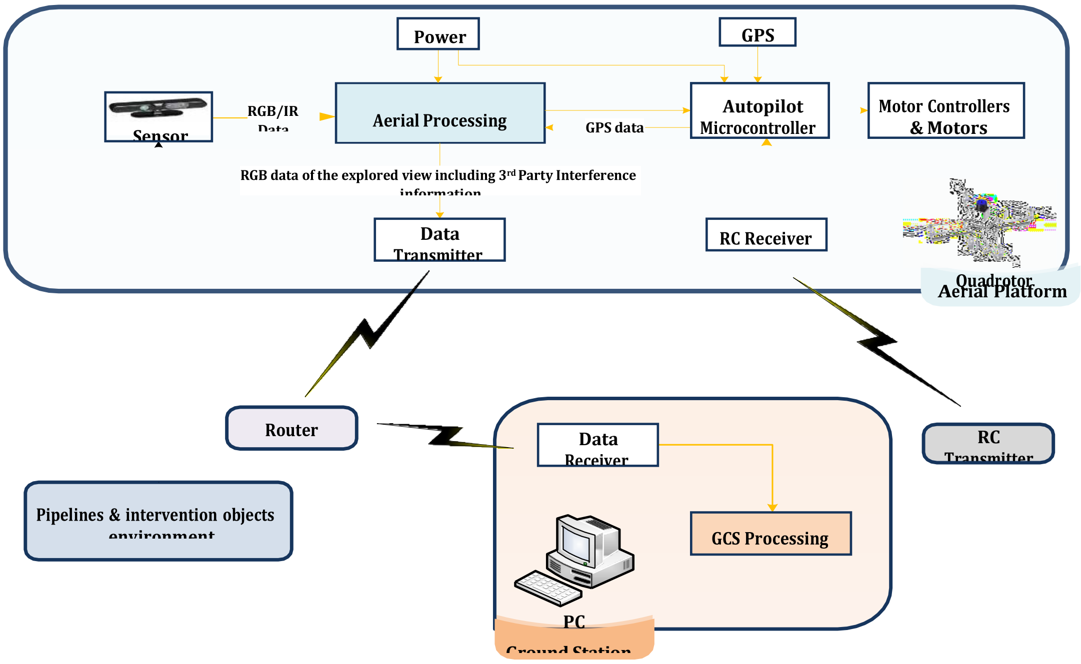


Figure 3-1: General system architecture of the vision-based aerial pipelinesurveillance system

The aerial platform payload includes the remote sensor that provides visualdata of the pipeline structure at high resolution. On-board data processing is carried-out and the data is transmitted to the ground station in real-time, so to track the pipeline structure automatically. The aerial platform consists of an EO/IR camera for image capture, a data recording, an on-board embedded processor for algorithms processing, autopilot controller to navigate the platform, power supply and, finally, a data link for the UAV command and control and data transmission. The ground control station receive/transmit the recorded data, process the computationally demanding parts of the developed algorithms, and stream the video data of the explored environment to the ground operator. The ground station consists of a workstation to process the algorithms and a data link for the UAV command, control and data transmission.The router is proposed to simply pass the data between the aerial platform and the ground station. The test-rig is proposed to evaluate and validate the system which involves a small-scale pipeline structure and small samples of third-party interference objects.

3.2.1 Aerial Platform

This section describes, in detail, the integrated hardware components and the developed software on-board the aerial platform. Also, it describes how to setup and interface them together.

3.2.2 Hardware Components

This section presents the hardware components that are on-board the aerial platform as part of the surveillance system. The hardware includes the following components:

1) The Gaui 500X Quadrotor (aerial UAV platform).
2) Asus Xtion Pro Live (RGB/depth vision sensor).
3) Nitrogen x6 board (processor board to run a computer vision algorithm).
4) ArduPilot Mega (microcontroller board to control the dynamics of the platform with assisting of the built-in autopilot, IMU, and GPS).
5) 3DR uBlox (GPS Module).
6) XBee module (Wi-Fi RF Module for wireless telemetry data transmissions).
7) Li-Po battery (Power supply for the embedded boards).

More information is described in the following subsections.

3.2.2.1 Gaui 500X Quadrotor

The Gaui 500X Quadrotor is a light weight UAV platform as shown in Figure 3-2 that is proposed to carry out the imagery sensor payload and the other corresponding components to perform in real-time the pipeline surveillance mission automatically (Multi-Rotor Technology (MRT), 2012).

The efficiency and load of the propeller are optimized, for improved wind resistance over other quads. It utilizes GU-344 (three-axis stabilizing system) for both beginners and professionals in combination with other electronics for FPV flying that makes it a very stable air-vehicle. The maximum flying weight is between 1100g to 2000g, depending on payloads used, such as batteries, 500XGAUI Motors/ESCs, cameras, processing boards and other related equipment. The main reason for selecting this platform is the maximum take-off weight, which is the total weight of the payload components and the UAV platform itself with all the systems as shown in Table 3-2.

Table 3-2: Weight details of the payload components

Item	**Weight**
Battery 3S	323g
Crane II	96g
ASUS Xtion Pro Live	218g
Battery 2S	83g
Nitrogen 6x board	88g
Mount	150g
TOTAL	**1000g**

Operation mode and flight characteristics are similar to those of helicopters without complex transmissions. It has a collapsible body design that greatly reduces crash damage and allows for easy repairs. With four brushless motors and 18A Electro Speed Controller (ESC), the 500X is really powerful and responsive. The specifications are listed in Table 3-3 below.

Table 3-3: GAUI 500X Quadrotor specifications (Multi-Rotor Technology (MRT),2012)

Item	**Specification**
Weight (g)	670
Max Flying weight (g)	Up to 2000
Platform diameter (mm)	500
Motors	960 kV, scorpion model
Electro Speed Controller (ESC)	Brushless 18A
Flight efficiency	Standard battery (2S 2000mAh), for flight times longer than 12min. With a high-capacity battery, the flight time will be 20 min or longer.

Figure 3-2: GAUI 500X Quadrotor platform (Multi-Rotor Technology (MRT), 2012)

3.2.2.2 ASUS Xtion Pro Live

Two vision data are acquired as main inputs to provide the function of colour (RGB) and the depth (elevation at down projection) information of the vision target scene. So, in the market, many sensors exist to provide them either, individually, using two separate sensors or together as one sensor. However, using one platform that combines all of them is superior in terms of cost, weight, interface, integration and operation. Hence, the proposed sensor used in this project is the ASUS Xtion Pro Live. This sensor provides combined functions of RGB colour information and per-pixel depth information. The sensor includes RGB camera; IR structured light source, and a second camera dedicated to IR detection that provides depth information as shown in Figure 3-3.

Figure 3-3: ASUS Xtion Pro Live Sensor (ASUS, 2012)

It was chosen for its low price £138 and low weight of 218 g, as shown in Figure 3-4 that guarantees that the Quadrotor payload limit remains at around of 1kg.

Figure 3-4: Asus Xtion Pro Live weight measurement

The technology of obtaining the depth is based on the structured light technique. Precision is similar to that of Time of Flight (ToF) cameras, 1cm more or less, with a limited range between 0.8-3.5 m, but they tend to have trouble seeing small objects and have to be indoors. The resolution and speedof this sensor are similar to the conventional VGA cameras, usually 640 by 480 at 30 fps. The technical specifications are shown in Table 3-4.

Table 3-4: ASUS Xtion Pro Live sensor specifications (ASUS, 2012)

Item	**Specifications**
Power consumption	< 2.5W
Depth range	0.8m to 4m
Field of view	58° H, 45° V, 70° D (Horizontal, Vertical, Diagonal)
Sensors	RGB & Depth & Microphone (x2)
Depth Image Size	VGA (640x480): 30 fps QVGA (320x240): 60 fps
Resolution	SXGA (1280*1024)
Platform	Intel X86 & AMD
OS support	Windows 7 32/64, XP, Vista Linux Ubuntu 10.10, X86, 32/64 bit Android
Interface	USB 2.0

Software	OpenNI SDK bundled
Programming language	C/C++ (Windows), C++ (Linux), and JAVA
Dimensions	180mm x 35mm x 50mm
Operating Environment	Indoor
Weight	218g

The IR camera and the IR projector form a stereo pair with a known baseline. Hence, a structured light technique works by projecting a fixed pattern of infrared light from the IR projector such as a grid of lines, or a constellation of points or dark speckles on top of the scene's objects as shown in Figure 3-5. This pattern is seen distorted when looked at from a perspective different from the projector. By analysing this distortion, information about the depth can be retrieved, and the surface reconstructed.

Depth is calculated by triangulation against a known pattern from the projector. The pattern is memorized at a known depth. For a new image, the depth is calculated at each pixel, and a small correlation window (9x9 or 9x7) is used to compare the local pattern at that pixel with the memorized pattern at that pixel and 64 neighbouring pixels in a horizontal window. Then, the best match gives an offset from the known depth, regarding a pixel which is called disparity. The device performs other interpolation of the best match to get the sub-pixel accuracy of 1/8 pixels that provide the known depth of the memorized plane,and the disparity. However, the estimated depth of each pixel is calculated by triangulation.

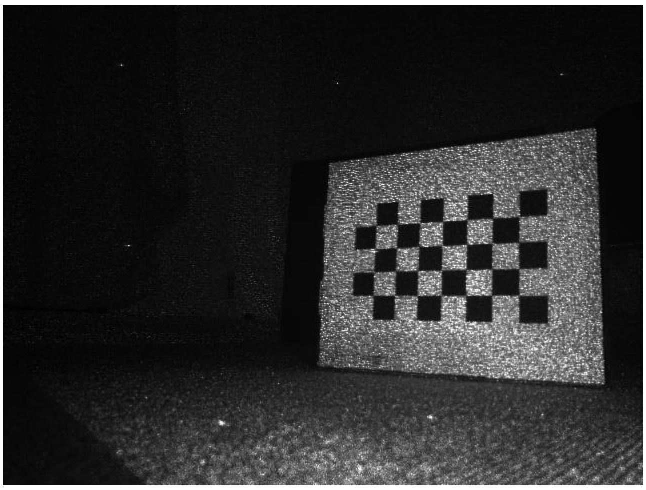


Figure 3-5: Sensor projection of infrared structured light pattern

Then, the depth of each pixel produces 3D data in the form of a point cloud. A point cloud is a set of points in three-dimensional space, each with its XYZ coordinates. So, every point corresponds to exactly one pixel of the captured images in the case of stereo, ToF or structured light cameras.

To mount the Asus Xtion Pro Live on the Gaui 500X Quadrotor platform, a step- down configuration Gimbal was selected to integrate them which is called Gaui Crane II, as shown in Figure 3-6. Moreover, the Asus Xtion Pro Live hardware isconnected to the Embedded Board using a USB cable.

Figure 3-6: Gaui Crane II Gimbal

To interact between the pixels of the depth and colour images with their corresponded real world coordinates, a 3D real world model needs to be constructed. To interact with the depth and colour images, an accurate camera calibration for depth and colour images is carried-out offline. This calibration is aprocess performed to compute the true intrinsic parameters of the camera, such as focal length, centre image position, and lens distortion. Besides, computing the extrinsic parameters (relative transform between the depth and the colour cameras), it also includes rotations and translation parameters.

The calibration was performed offline on the colour and depth cameras to find the intrinsic and extrinsic parameters by using the same approach as (Herreraet al., 2012). This approach was designed to calibrate simultaneous multicolour cameras, a depth camera, and the relative transform between them. First, the intrinsic parameters of both colour and depth cameras are calibrated. Then, the relative transform between the cameras (extrinsic calibration) was calibrated by computing the rotation and translation parameters to enable the alignment between them.

The approach requires only a planar surface with a simple checkerboard patternto be imaged from various poses as shown in Figure 3-7. The checkerboard corners provide suitable constraints for the colour images while the planarity of the points provides constraints on the depth images. The pixels at the borders of the calibration object can be ignored and, thus, depth discontinuities are not used. For this purpose, Kinect Calibration Toolbox for Matlab was used.

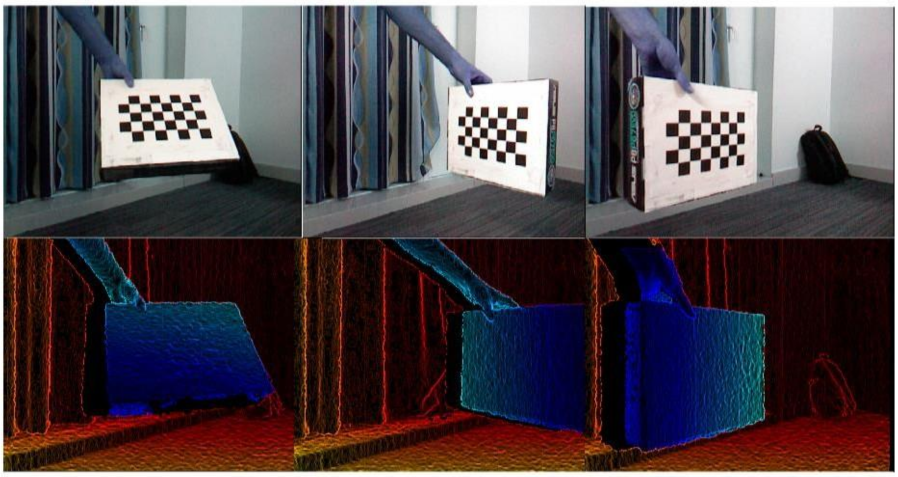


Figure 3-7: Samples of calibration images at various poses

For the colour camera, the checkerboard corners are extracted from the colour intensity image using a corner detection algorithm in image processing to obtainthe colour intrinsic parameters as shown in Figure 3-8. Then, a homography is computed for each colour image using the known positions of the corners in world coordinates and the relative pixels in the colour image coordinates. After that, each homography imposes constraints on the intrinsic parameters that are solved with a linear system of equations. The distortion coefficients are initially set to zero.

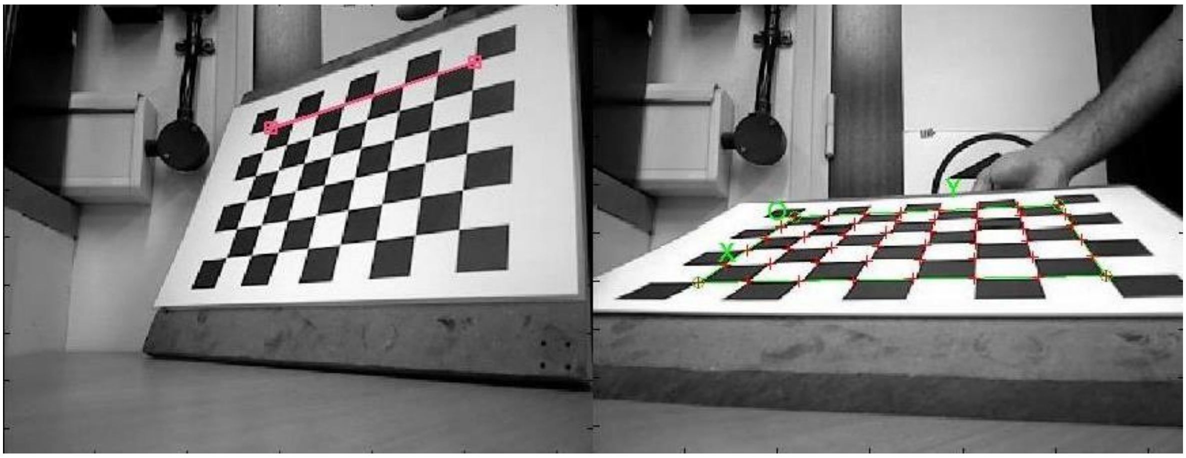


Figure 3-8: Checkerboard corner detection in a colour image

The same method is applied to the depth camera to obtain the depth intrinsic parameters, in which the noisy four corners of the plane are extracted manually in the depth image as an initial guess because the checkerboard is not visible, as shown in Figure 3-9. Then, a homography is computed for each depth image using the corner positions in plane coordinates and the relative pixels in the image depth coordinates. This computation will produce the focal lengths, centre points, and the transformation parameters.

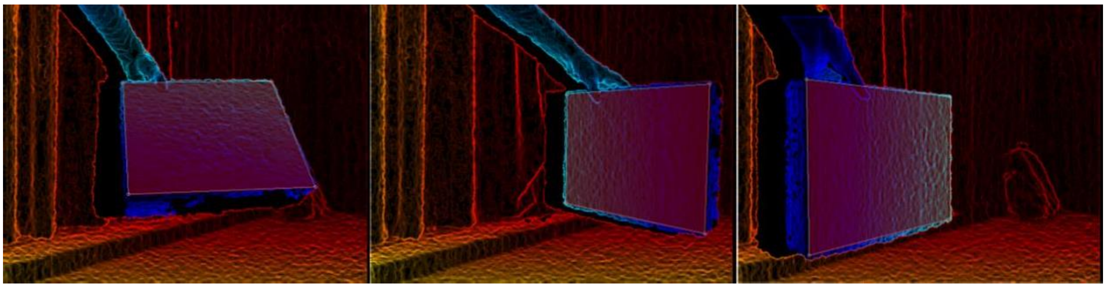


Figure 3-9: Plane's corner detection in depth image (red boxes)

3.2.2.3 Nitrogen 6x Embedded Board

In order to increase the level of autonomy at the quadrotor, an embedded board was added on-board the platform to process the computer vision algorithms that control the dynamic of the platform in real-time. The embedded processor board that is proposed in this project is based on the Nitrogen 6x as shown in Figure 3-10.

Figure 3-10: Nitrogen 6x embedded board (Boundary Devices, 2012)

This specific embedded board was chosen for several reasons, such as the limited size and weight and the built-in Wi-Fi module, as shown in Figure 3-11 that was used to communicate with the ground control station (GCS).

Figure 3-11: Nitrogen 6X Add-on Ti-Wi Module (Boundary Devices, 2012)

The Nitrogen 6X is a highly integrated development system based on the next generation Quad-Core ARM-Cortex A9 processor from Freescale, the i.MX6. This processor supports a wider and faster memory bus (64-bit DDR3 1066 MHz), integrated HDMI, Gigabit Ethernet and additional display channels with a high level of integration. It is a low-cost development platform. Additional highlights of the board are listed below (Boundary Devices, 2012):

i. Quad-Core ARM® Cortex A9 processor at 1GHz.
ii. RAM memory of 1GBytes of 64-bit wide DDR3 @ 532MHz.
iii. Board Dimensions: 4.5″ x 3″.
iv. 2MB Serial Flash.
v. Three display ports (PRGB, LVDS, HDMI).
vi. Parallel camera port with OV5642 Interface.
vii. Multi-stream-capable HD video engine delivering 1080p60 decode,1080p30 encode and 3D video playback in HD.
viii. Superior 3D graphics performance with quad shaders for up to 200 Mt/s.
ix. Separate 2-D and/or Vertex acceleration engines for an optimal userinterface experience.
x. Serial ATA (SATA).
xi. Dual SDHC card slots.

xii. PCI express port.

xiii. Analog (headphone/mic) Audio.

xiv. 10/100/1G Ethernet with Power over Ethernet support.

xv. 2 RS-232 Serial ports.

xvi. 10-pin JTAG interface.

xvii. I2C/GPIO/SPI.

xviii. High-speed USB ports (2xHost, 1xOTG).

xix. CAN port.

xx. Real Time Clock.

xxi. Ti-Wi 802.11 b/g/n Wi-Fi+BT.

xxii. Supports Android 4.3, Embedded Linux, and WinCE7.0 OperatingSystems.

The Nitrogen 6x board runs a Linux-based operating system. The computer vision algorithms were coded in Python with the aid of OpenCv and OpenNi libraries.

The Linux operating system installed on the Nitrogen6x is Debian GNU-Linux. Debian is a Linux distribution with access to online repositories hosting software packages. Debian officially hosts free software in its repositories but also allows commercial 3rd party software to be installed. Furthermore, it gives the possibility to install all the dependencies and libraries required to use the Asus Xtion Pro Live.

The Open Natural Interaction (OpenNI) is a multi-language framework and an open source application that is used to interface the Xtion Pro Live sensor on this system by providing functions capable of acquiring depth data from the Asus Xtion Pro Live. It could be used with any depth sensor such as Xbox Kinect or Asus Xtion. The OpenNI framework gives the interfaces either for the physical apparatus or the middleware components, as seen in Figure 3-12, by compiling it with OpenCV library (OpenNI, 2012).

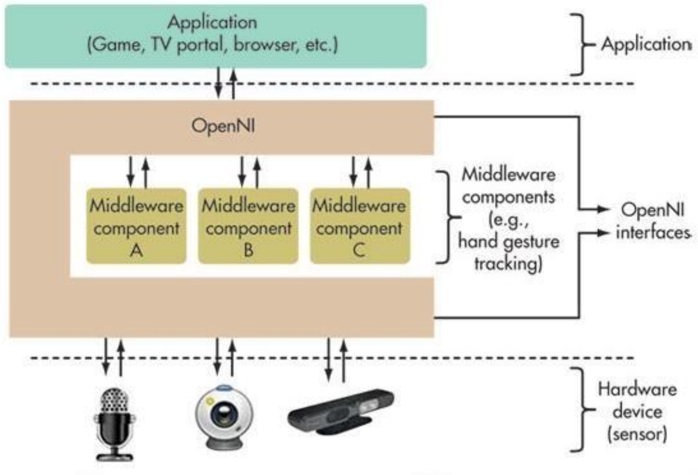


Figure 3-12: OpenNI Framework (OpenNI, 2012)

OpenCV is an Open Source Computer Vision library that includes many built-in algorithms, as described by (Opencv dev team, 2012). It is free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. OpenCV was designed forcomputational efficiency and has a strong focus on real-time applications. Written in optimized C/C++, the library can take advantage of multi-core processing. Enabled with OpenCL, it can benefit from the hardware acceleration of the underlying heterogeneous compute platform.OpenCV has a modular structure, which means that the package includes several shared or static libraries. The following modules are available (Opencv dev team, 2012):

I. Core - a compact module defining basic data structures, including the dense, multi-dimensional array Mat and basic functions used by all other modules.

II. imgproc - an image processing module that includes linear and nonlinear image filtering, geometrical image transformations (resize, affine and perspective warping, generic table-based remapping), colour space conversion, histograms, and so on.

III. Video - a video analysis module that includes motion estimation, background subtraction, and object tracking algorithms.

IV. calib3d - basic multiple-view geometry algorithms, single and stereo camera calibration, and object pose estimation, stereo correspondence algorithms, and elements of 3D reconstruction.

V. features2d - salient feature detectors, descriptors, and descriptor matchers.

VI. Objdetect - detection of objects and instances of the predefined classes (for example, faces, eyes, mugs, people, cars, and so on).

VII. Highgui - an easy-to-use interface to video capturing, image and video codecs, as well as simple UI capabilities.

VIII. gpu - GPU-accelerated algorithms from different OpenCV modules.

To interface the Nitrogen 6X embedded board to the Asus Xtion Pro live hardware, the attached Xtion's USB cable is plugged into one of the USB ports onboard. Furthermore, the Nitrogen6x is interfaced to the Ardupilot Mega viathe serial port UART1.

3.2.2.4 ArduPilot Mega Microcontroller Board

The proposed hardware to control the Quadrotor platform is the ArduPilot Mega (APM). APM is a fully programmable open source autopilot system that requires a GPS module and sensors to create a functioning Unmanned Air- vehicle (UAV) (ArduPilot, 2013). It can control many platforms, such as fixed- wing air-vehicle, multi-rotor helicopters, traditional helicopters, as well as ground rovers. It has full autopilot capability for autonomous stabilization, waypoint based navigation and two-way telemetry using XBee wireless modules. Also, it supports 8 RC channels with four serial ports.

This controller consists of the main processor board as shown in Figure 3-13, and the APM firmware that is a code runs onboard to control the chosen UAV platform as shown in Figure 3-14.The ArduPilot Mega was selected due to the following features (ArduPilot,2013):

1. Controller designed to be used with autonomous air-vehicle, multicopters (tri, quad, hex, oct, and so forth), traditional helicopters, car or boat.
2. Based on a 16MHz Atmega2560 processor.
3. Built-in hardware failsafe that uses a separate circuit (multiplexer chipand ATMega328 processor) to transfer control from the RC system to theautopilot and back again. Includes ability to reboot the main processor in mid-flight.
4. Dual-processor design with 32 MIPS of on-board power.
5. Supports of 3D waypoints and mission commands (limited only by memory).

6. It comes with a 6-pin GPS connector (EM406 style).
7. It has 16 spare analog inputs (with ADC on each) and 40 spare digitalinput/outputs to add additional sensors.
8. Four dedicated serial ports for two-way telemetry and in-flight commandusing the powerful MAVLink protocol.
9. It can be powered by either the RC receiver or a separate battery.
10. Hardware-driven servo control, which means less processor overhead,tighter response and no jitters.
11. The autopilot can process eight RC channels (including the autopiloton/off channel).
12. LEDs for power, failsafe status, autopilot status and GPS lock.
13. 4MB of on-board data logging memory. Missions are automatically data-logged and can be exported to KML.
14. It has full autopilot software, including IMU and ground station/missionplanning code.

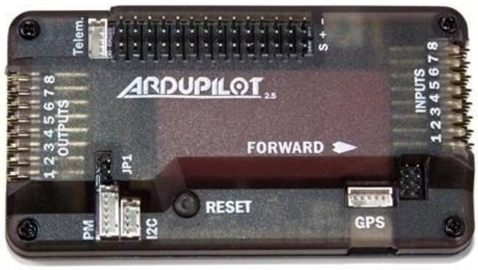


Figure 3-13: ArduPilot Mega Autopilot board (ArduPilot, 2013)

Figure 3-14: ArduPilot Mega onboard firmware (ArduPilot, 2013)

The communication between the Nitrogen 6x and the ArduPilot has been obtained using a serial port UART with the following configuration parameters:

1. Baud rate=9600
2. Stop bit=1
3. Parity=None

The software has been developed for both the Nitrogen 6x and the ArduPilot. Concerning the embedded board, which works with a Linux-based OS, the "Termios" library was used for the ArduPilot, while the Arduino integrated development environment (IDE) provides the main tools for serial portprogramming.

The anatomy of a program performing serial I/O is as follows:

i. An open serial device with standard system called <u>open.</u>
ii. Configure communication parameters and other interface propertieswith the help of specific functions and data structures.
iii. Use system calls <u>read</u> and <u>write</u> for reading from, and writing to theserial interface.
iv. Close device with the system called <u>close</u> when done.

3.2.2.5 3DR uBlox GPS

The 3DR uBlox GPS module has an on-board compass kit, and it is the most recommended and compatible module to be used with the APM mounted on the aerial platform. This module provides the telemetry data of the aerial platform to the Ground Control Station (ArduPilot, 2013).

3.2.2.5.1 Features and Specifications:

i. ublox LEA-6H module.
ii. 5 Hz update rate.
iii. 25 x 25 x 4 mm ceramic patch antenna.
iv. LNA and SAW filter.
v. Rechargeable 3V lithium backup battery.
vi. Low noise 3.3V regulator.
vii. I2C EEPROM for configuration storage.
viii. Power and fix indicator LEDs.
ix. Protective case.
x. APM compatible 6-pin DF13 connector.
xi. Exposed RX, TX, 5V and GND pad.
xii. 38 x 38 x 8.5 mm total size, 16.8 grams.

This GPS is connected with the ArduPilot Mega flight controller Board using twocables, as shown in Figure 3-15, which are:

1. Four-position cable to connect the GPS MAG port to the APM I^2C port.
2. Five-position-to-six-position cable to connect the GPS port to the APMGPS port.

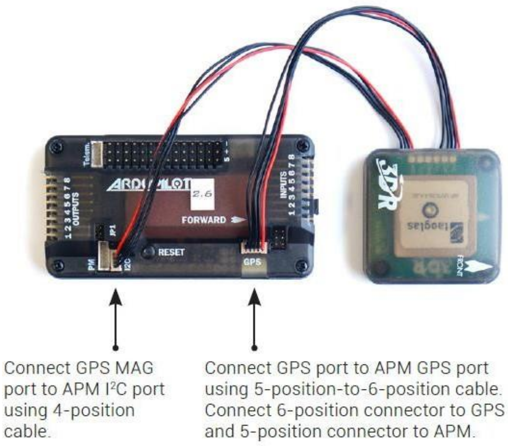


Figure 3-15: 3DR uBlox GPS with ArduPilot Mega hardware interface (ArduPilot,2013)

The 3DR uBlox GPS with Compass is already pre-configured for compatibilitywith APM autopilot.

3.2.2.6 Xbee 802.15.4 Transceiver

The XBee-PRO® OEM RF module is an IEEE 802.15.4 compliant solution, that satisfies the unique needs of low-cost, low-power wireless sensors (DIGI, 2014). This module when mounted on the aerial platform can send/receive the telemetry data wirelessly, as shown in Figure 3-16.

The advantages and limitations of this module are as follows:

i. Operate at ISM 2.4 GHz frequencies.

ii. Easy-to-use.

iii. Requires minimal power and provides reliable delivery of critical databetween devices.

iv. It can send a live telemetry data as the airframe progresses through themission

v. A new mission could be sent while the UAV flies without having to land.

vi. The range of operation has been tested out for half a mile with no loss in connection; the connection has been found to drop off at 3/4 of a mile.

vii. It is capable of operating at a temperature rating (-40 to 85°C).

viii. Approved for use in Europe, Canada, Australia, and the United States.

ix. It supports advanced networking & low-power modes.

Figure 3-16: XBee transceiver (DIGI, 2014)

Onboard the aerial platform, the XBee module (pins) is connected with the ArduPilot Mega (Teleport) using the 3DR four-wire XBee cable and XBee adapter to interface them together as shown in Figure 3-17.

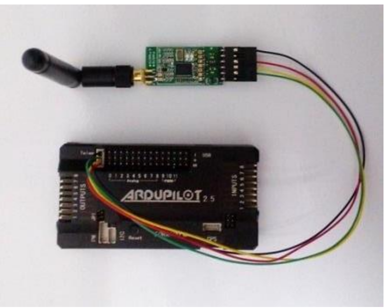


Figure 3-17: XBee & APM hardware interface (ArduPilot, 2013)

Then, to interface them together, it is required to set up the right port and the baud rate (APM default is 57600 bps) by using the moltosenso Network Manager or using the GCS mission planner directly.

3.2.2.7 Li-Po Battery

The Gaui 500X Quadrotor has a standard battery (2S 2000mAh), for flight times longer than 12 min. With a high-capacity battery, the flight time will be 20 min or longer. The Nitrogen 6X board requires a 5V/3A max power source. However, a high discharge Li-Po 1300mAh battery has been chosen using three Cells at 11.1V. A DC/DC converter to 5V has been integrated to provideup to 60 minutes of practical work as shown in Figure 3-18.

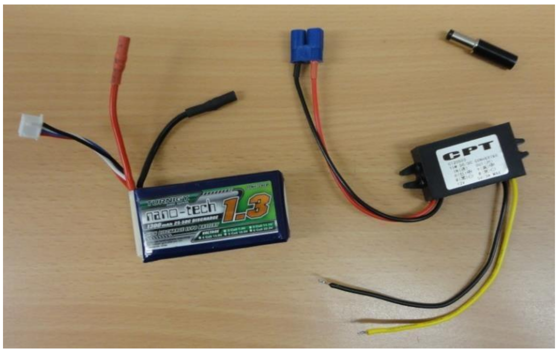


Figure 3-18: Li-Po 1300mAh battery & DC/DC converter

CHAPTER FOUR: PIPELINE ENDPOINTS IDENTIFICATION

4.1 Introduction

This chapter presents the development of the vision-based algorithm thatis used onboard the Unmanned Aerial Air-vehicles (UAV), to identify and localize the endpoints of the pipeline structure in real-time. The first objective of this chapter is to develop a reliable algorithm capable of estimating the endpoints position of the pipeline segments throughout the frame sequences in near real-time. The aims of this development are to keep track/follow the pipeline online, down-sample the data in each frame by filtering the region that is an outlier of the Right-of-Way to focus the processing on the data located within the Right-of-Way region. This leads to reducing the computational cost to detect any third-party interference and any defects.

In this project, two methods are proposed to identify and localize the over- ground pipeline structure. One is based on the standard camera (RGB intensities) and the other is based on the IR sensor (depth intensities). Different computer vision techniques were used with each sensor. The proposed techniques are capable of detecting and localizing the pipeline with high detection and processing rates. The techniques are described in detail in the following sections. Furthermore, the experimental setup, simulation, analysis and performance evaluations are performed for each method and are described in this chapter.

4.1.1 Visible-Based Pipeline Endpoints Identifications

This section describes the proposed visible-based algorithm used to detect and localize the pipeline structure using a UAV mounted with a standard visible camera. This algorithm is proposed to detect and localize the endpoints of the pipeline structure, efficiently in real time, based on the colour intensity in orderto keep track of the pipeline and maintain monitoring and surveying.

Image processing techniques were used to develop this algorithm. The first stepof the processing involves enhancing the image data to improve the edgedetection performance. Then, the boundaries of the features are detected in the sense of the image by using a canny edge detector. Afterward, the boundary candidates are extracted to represent a straight line

using the Hough Transform method. Following this, the straight lines are filtered by comparing them with known pipeline segments in the area, allowing fast and accurate pipeline identification. This step enables the system to reject with high degree of reliablity inaccurate measurements while retaining the correct pipeline detections and location.
Several image sets and two aerial videos, captured by the standard camera, were used to test and analyse the system. The performance of this algorithm was evaluated using a Matlab/Simulation environment to verify that the system is capable to reliably detect and localize the pipeline structure with a low cost aerial platform, with high detection rate and fast processing time.

4.1.2 Image Acquisition

Two images are used to demonstrate how the proposed algorithm detectsthe pipeline as shown in Figure 4-1. These images were captured by (Pipeline Surveillance Company - Not for Public Release, 2012) from a manned air- vehicle from two different views and in various environments, at a resolution estimated to be greater than 0.5 m per pixel. It is worth mentioning that the two images were captured at different altitudes.

Figure 4-1: Two images of a pipeline at various environments (Pipeline Surveillance Company - Not for Public Release, 2012)

4.2 Pipeline Detection & Localization Algorithm

Since the most recognizable feature of the pipeline structure that is easily detectable in the standard RGB image is the shape, the development commenced with detection of the line as a shape. However, there are many different methods to detect line features in image processing applications. Inthis project, the proposed algorithm to detect the pipeline is demonstrated in Figure 4-2.

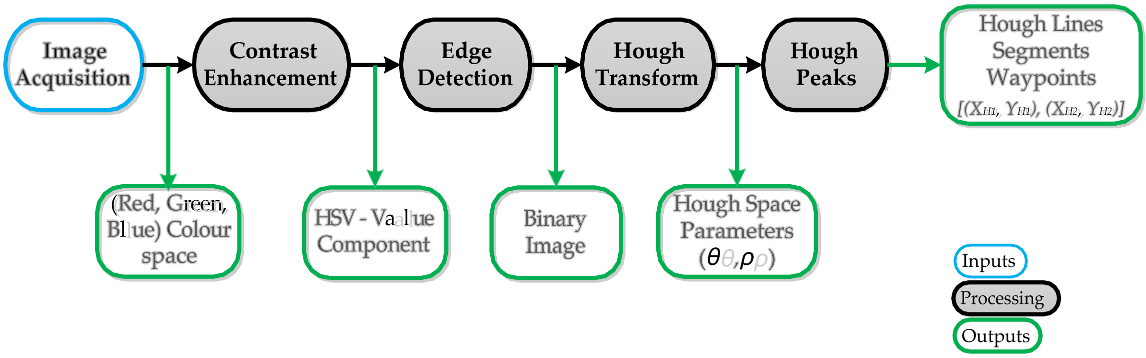


Figure 4-2: Camera-based pipeline detection algorithm

At first, the RGB data is acquired from the sensor. Then, those RGB colours are converted into HSV colours to simplify and enhance the analysis. After that, a canny edge detector is implemented to detect the edges as points based on the differentiation of the colour intensity. Then, based on those points of the edges,a Hough Transform method is implemented to construct a solid line configuration that potentially refers to the pipeline in the image. Finally, the lines constructed are represented as waypoints.

4.2.1 Contrast Enhancement

In order to obtain optimum edge detection, the contrast is enhanced. This enhancement can be done by improving the intensity (luminance) which is the total amount of light passing through a particular area. The RGB colour space describes colours based on the intensities of Red, Green, and Blue channels, respectively, which are not always suitable for colour based applications. Therefore, it is useful to transform it into Hue, Saturation & Value (HSV) colour space. It describes colours in the same way as the human eye senses colour, where a colour is represented by its Hue (H) and Saturation (S). The saturation adds white light to distinguish between the components of each surface by its lightness (such as metallic for pipe, asphalt for the road, water in a lake or sea, and on forth). Value (V) describes the overall intensity or strength of the light. A detailed description of the HSV colour space can be found in Gonzalez and Woods (2003).

Figure 4-3 demonstrates the difference between the intensity of RGB colour space and value components of HSV colour space in two images with different environments. Figure 4-4 shows the effect of this improvement when an edge detector has been applied.

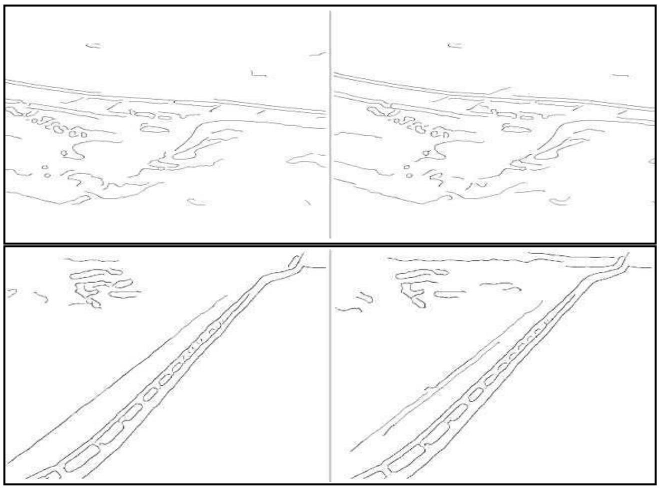
Figure 4-3: (Left): Intensity (Grey colour) of RGB colour space, (Right): Value (V) component of HSV colour space.

4.2.2 Edge Detector

The second step in the algorithm of line detection is finding the edges for the Hough Transform. However, the optimal method used in this research to detect the edges is the canny edge detector which is developed by Canny (1986). The output of this detector is a binary image represented by points based on the differentiation of the colour intensity as demonstrated in Figure 4-4.

The process of this method starts by smoothing (blurring) the image to eliminatenoise. Then, it finds where the high magnitude of image gradients is to highlight regions that have a high spatial derivative. The algorithm then tracks through these regions and suppresses any point that is not at the maximum region (non-maximum suppression) and marks only local maxima as edges. The gradient array is now further decreased by hysteresis. Hysteresis is used to track throughout the remaining points that have not been suppressed. Hysteresisuses two thresholds to the gradient to determine the potential edges. When the gradient magnitude is below the lower threshold, it is set to zero (no edge). Ifthe gradient magnitude is above the high threshold, it makes a strong edge. If the gradient magnitude is between them, it is set to zero, unless it connects to a strong edge with a gradient above the high threshold.

Figure 4-4: The edges detected as a binary image based on the RGB intensity



(Left), and the HSV value component (Right)

4.2.3 Hough Transform

Hough Transforms (HT) is a robust method for finding lines in the image that was developed by (Hough, 1962). The image features (points) that are detected by the canny edge detector at particular (x, y) coordinates is evidence that there might be a line passing through some of those points.

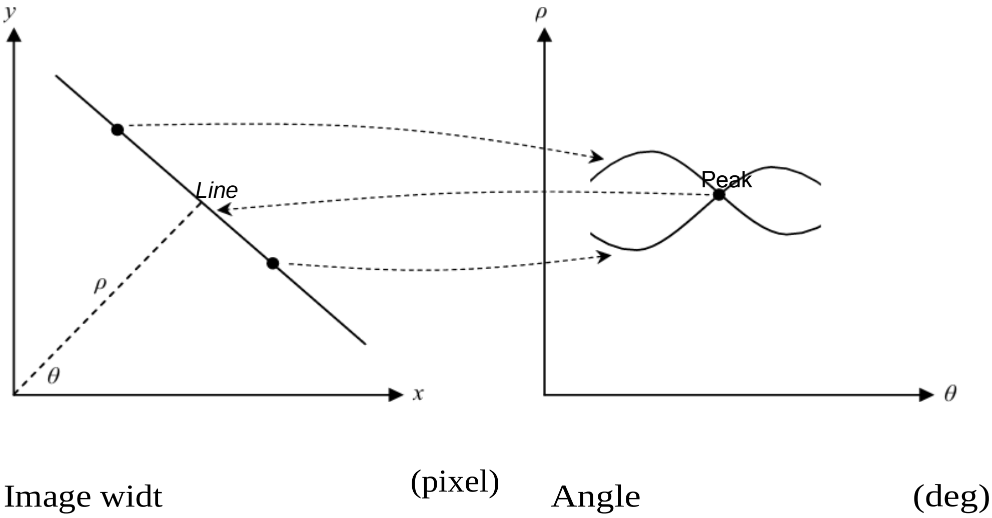


Figure 4-5: (Left) Image space and (Right) Parameter space or Hough space (Antolovic, 2008)

The process steps of achieving Hough Transform method are as follows:

i. Line detection: Accumulate tracings of the parametric sinusoid (ρ, θ) in Hough space for each pixel (x, y) in image space by using the Equation(4-1) from (Antolovic, 2008) as shown in Figure 4-5.

$$\rho = x \cos \theta + y \sin \theta \qquad (4\text{-}1)$$

Where: ρ is the shortest distance from the corner of the image space to the straight line, θ is the angle between the shortest distance ρ and the horizontal axis (width) in image space, and (x, y) represents the width and height of the pixel in image space.

ii. Peaks detection: the intersection points in Hough space between the curves which represent a straight line at parameter values (ρ, θ) as shown at the right graph in Figure 4-5. However, as more curves are intersected at that point, the more the line becomes solid, which represents the number of points corresponded.

iii. Linking the lines: Once a set of peaks has been detected in Hough space, it remains to be determined if there is a line segment associated with those

peaks, as well as start and ending points. For each peak, the first step is to find the location of all non-zero pixels in the image space that contributed to that peak and construct line segments based on those pixels.

Figure 4-6: The longest line detection using Hough Transform

4.2.5 Filtering & Enhancement

Finally, based on the lines detected by Hough Transform; the longest line was identified as a pipeline candidate. This assumption is based on the idea that there will only be one main line feature within the image, Figure 4-6.However, if there are many features in a line, such as multi-segments, then itwill not be suitable to just choose the longest line which will lead to missing the other lines, as shown in Figure 4-7.

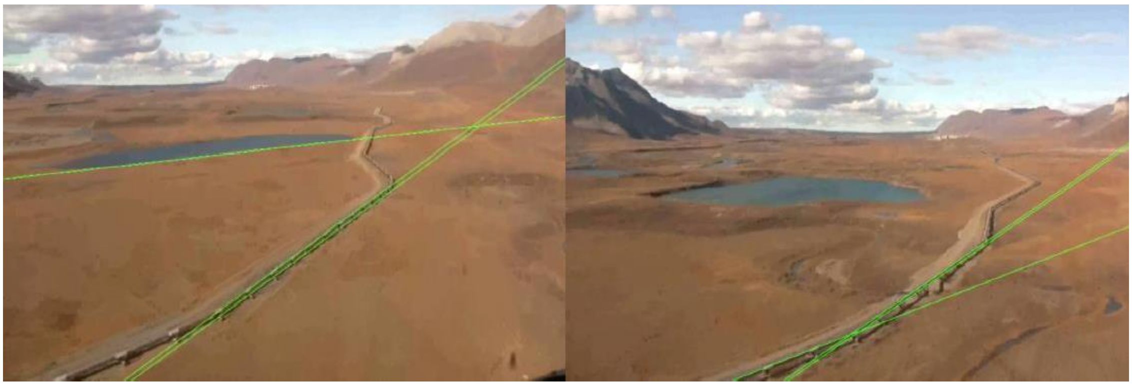


Figure 4-7: True & false detection of multi-lines detection using HoughTransform

However, to identify the right line segments and improve the performance of the detection algorithm, the pipeline candidates are projected from the image frame to the world by transforming the pixel coordinates (x_i, y_i) into 3D-world coordinates (X, Y, Z). These line segments are then compared to the existing pipeline waypoints within each frame to determine the correct detection asdescribed in Figure 4-8.

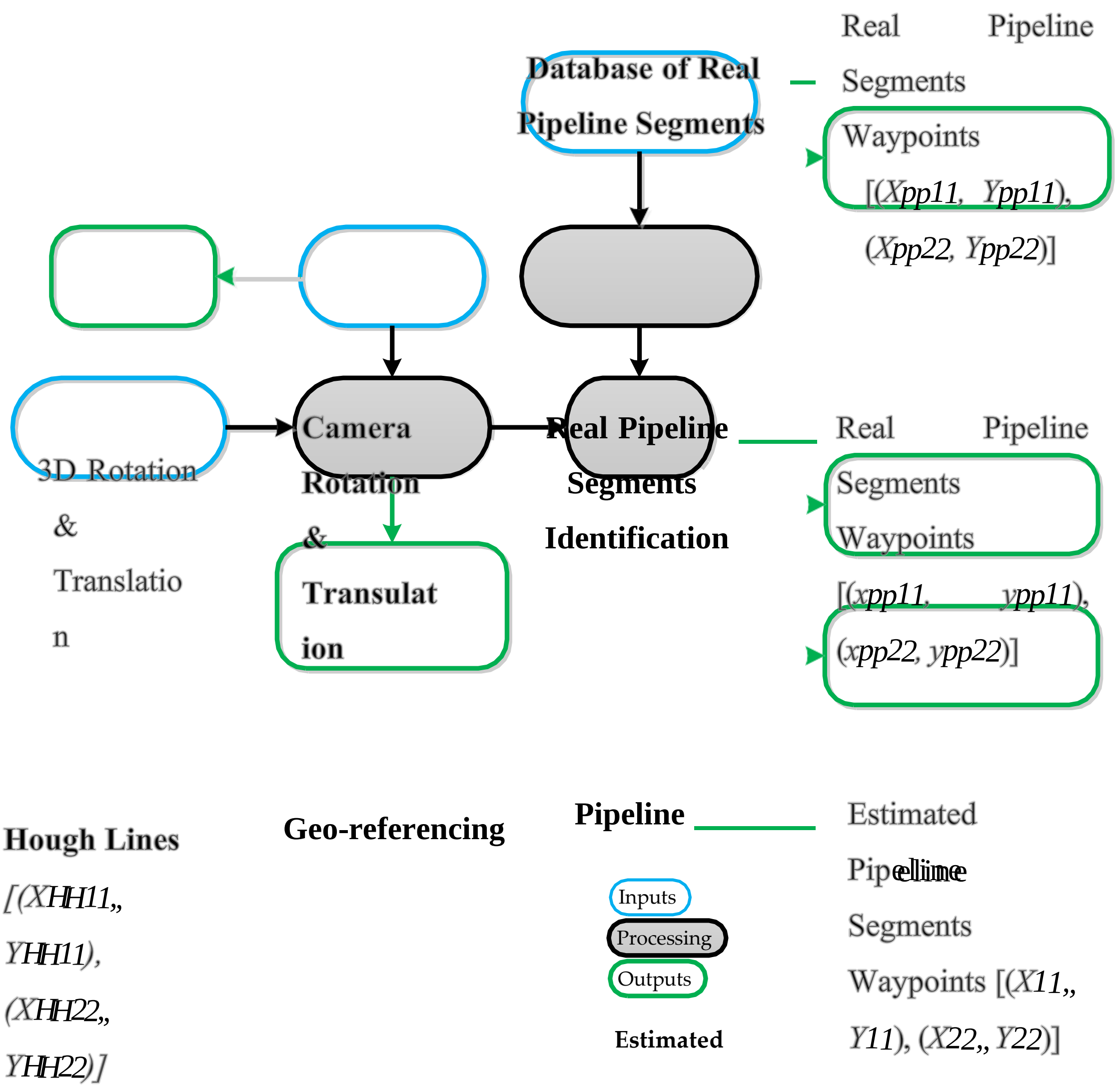


Figure 4-8: Filtering & enhancement of pipeline detection

4.3 Geo-Referencing

In order to calculate the errors between the known pipeline segments and the lines detected by the vision algorithm, the first step is to geo-reference the line features found in the image. It is an important issue to project the camera measurements into measurements in the real 3D world because scenes are not only 3D; they are also physical spaces with physical units.

Consequently, the relation between the camera's coordinate frame and the world coordinate system is a critical component in any attempt to reconstruct a 3D scene. In order to implement this, the camera is calibrated to model the camera's geometry and to estimate the distortion properties of the lens. These measurements define the intrinsic parameters of the camera. When these factors are known it is possible to use homography transform to project points from the image plane to the ground plane (assuming the terrain is flat).

Once the Hough lines are projected into the 3D world coordinate system, the endpoints of the pipe are estimated based on the distance between the existing pipeline, Hough lines, and their parallelism.

4.4 Real Pipeline Endpoints Identification and Localization

Before estimating the position of the pipeline endpoints, the endpoints of the real pipeline segment that pass through the image need to be identified, to avoid matching incorrect areas as described in Figure 4-9.

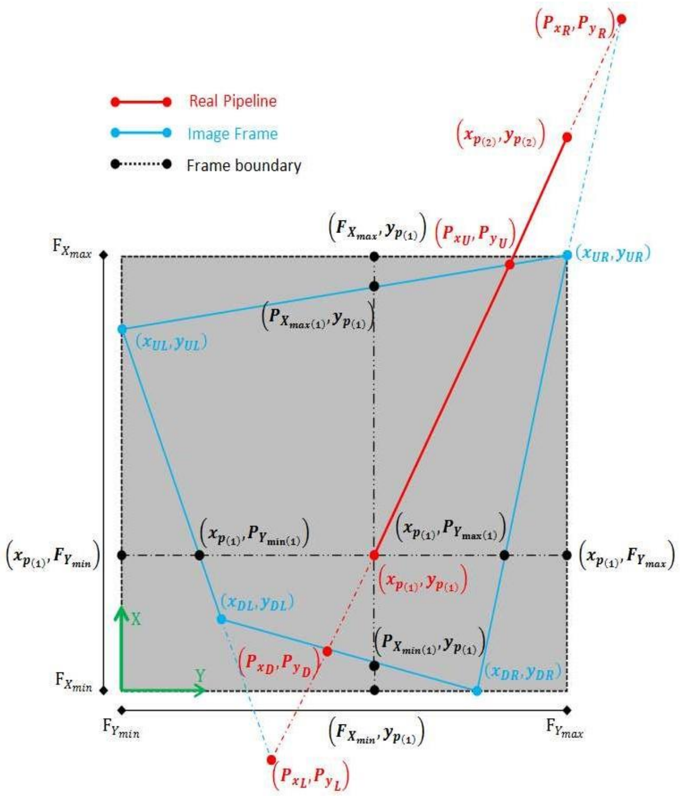


Figure 4-9: Real pipeline endpoint identification

To identify the endpoints of this pipeline segment, the following steps weredeveloped as follows:

1) Predefine the entire real pipeline endpoints for each segment $[(x_{P(1)}, y_{P(1)}), (x_{P(2)}, y_{P(2)})]$.

2) Calculate the image corners in the real world based on the resolution ofthe original image frame $[(x_{UR}, y_{UR}), (x_{UL}, y_{UL}), (x_{DR}, y_{DR}), (x_{DL}, y_{DL})]$.

3) Determine the maximum and minimum frame boundaries $(F_{x_{max}}, F_{x_{min}}, F_{y_{max}}, F_{y_{min}})$.

4) Calculate the interception of each line segment with the edges of theframe $[(P_{x_U}, P_{y_U}), (P_{x_D}, P_{y_D}), (P_{x_L}, P_{y_L}), (P_{x_R}, P_{y_R})]$.

5) Determine the maximum and minimum boundaries for each predefinedendpoint within the edges of the frame $(P_{X_{max(1)}}, P_{X_{min(1)}}, P_{Y_{max(1)}}, P_{Y_{min(1)}})$.

6) Decide if the pipeline segment falls within the image:

   a. If the entire segment falls within the frame, confirm these endpoints.

   b. If one endpoint of the predefined line segment meets the condition, while the other endpoint falls outside the frame, confirm the acceptable point and calculate the second endpoint from the interception endpoint.

   c. If neither of the pipeline endpoints falls within the frame, but the segment is crossing the image, identify the interception points at the edges of the image.

4.5 Pipeline Position Estimation

Estimating the position of the pipeline is achieved by comparing a predefined number of lines detected by Hough Transform with the predefined real position of the pipeline endpoints at each frame and selecting the nearby one. The geometric calculations used in this project are shown in Figure 4-10, where the distance and parallelism between two lines are calculated. First, the length (l) of the real pipeline segment that is passing through the image is calculated using Pythagorean Theorem (Boljanovic, 2006) as shown in Equation (4-2):

$$l = \sqrt{(x_{p1} - x_{p2})^2 + (y_{p1} - y_{p2})^2} \qquad (4\text{-}2)$$

Where: l is length of the real pipeline segment that is covered in the image frame. (x_{p1} , y_{p1}) and (x_{p2} , y_{p2}) are the width and height of the first and second endpoints of the real pipeline segment relative to the image frame, respectively.

Similarly, calculating the distances (l_{11}, l_{12}, l_{21}, and l_{22}), that are located between the endpoints of the real pipeline and the endpoints of the detected Hough lines as denoted in Figure 4-10 using again Pythagorean Theorem (Baer, 2005) as shown in Equation (4-3) to Equation (4-6).

$$l_{11} = \sqrt{(x_{p1} - x_{H1})^2 + (y_{p1} - y_{H1})^2} \qquad (4\text{-}3)$$

$$l_{12} = \sqrt{(x_{p1} - x_{H2})^2 + (y_{p1} - y_{H2})^2} \qquad (4\text{-}4)$$

$$l_{21} = \sqrt{(x_{p2} - x_{H1})^2 + (y_{p2} - y_{H1})^2} \qquad (4\text{-}5)$$

$$l_{22} = \sqrt{(x_{p2} - x_{H2})^2 + (y_{p2} - y_{H2})^2} \qquad (4\text{-}6)$$

Where: l_{11}, l_{12}, l_{21}, and l_{22} are the distances between each endpoint of the real pipeline segment with each endpoint of the detected Hough line segment.x and y refer to the north and east coordinates, respectively. p and H refer to the real pipeline and the Hough line segments, respectively. 1 and 2 denote the first and second endpoints of each segment, respectively. After that, the angles α_1 and α_2, represented in Figure 4-10, are calculated using the law of cosines formula (Boljanovic, 2006) as in Equation (4-7) and Equation (4-8):

$$l_{11}^2 - l_{12}^2 - l^2 + 2l_{12}l \cos \alpha_1 = 0 \tag{4-7}$$

$$l_{22}^2 - l_{21}^2 - l^2 + 2l_{21}l \cos \alpha_2 = 0 \tag{4-8}$$

Then, the shortest distances ρ_1 *and* ρ_2 are determined that are located at the first and second endpoint of the Hough line segment and the real pipeline segment as denoted in Figure 4-10 using the law of sines formula (Boljanovic, 2006) as shown in Equation (4-9) and Equation (4-10).

$$\rho_1 = l_{12} \sin \alpha_1 \tag{4-9}$$

$$\rho_2 = l_{21} \sin \alpha_2 \tag{4-10}$$

Hence, the distance ρ_a between the Hough line segment and the real pipeline segment is calculated by averaging the shortest distances of each endpoint (ρ_1, ρ_2) as indicated in Figure 4-10 using Equation (4-11).

$$\rho_a = \frac{\rho_1 + \rho_2}{2} \tag{4-11}$$

In addition, the parallelism factor (ρ_p) used to measure the parallelism of anypair of straight lines is obtained based on the absolute difference between the normal lines of each endpoint using Equation (4-12):

$$\rho_p = |\rho_1 - \rho_2| \tag{4-12}$$

Finally, based on the obtained distance (ρ_a) and the parallelism factor (ρ_p), the line is selected to represent the candidate pipeline based on the limitations of the distance between any pair of lines as well the parallelism factor.

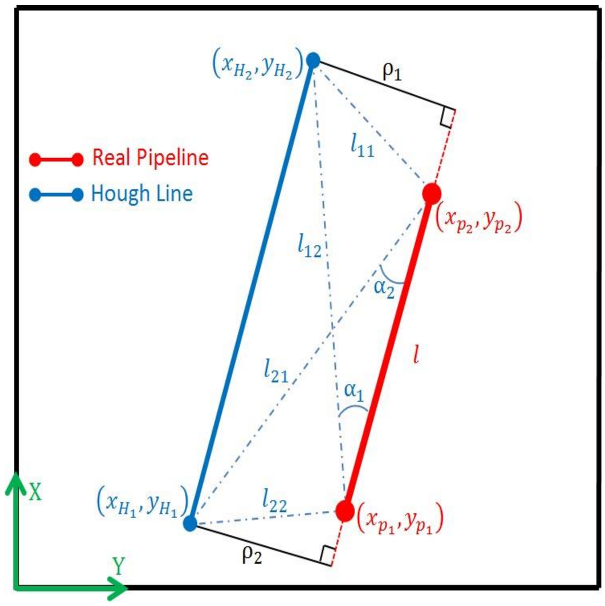


Figure 4-10: Geometric calculations for pipeline position estimation

CHAPTER FIVE: THIRD-PARTY INTERFERENCE DETECTION

5.1 Introduction

Today, one of the main defects of pipeline safety in the world is due to third- party interference, almost involving 40% of the pipeline integrity defects. So, to reduce this kind of problem, it is necessary to develop and build a reliable algorithm, capable of detecting and localizing any third-party interference, automatically in real-time. Therefore, integrating a small aerial platform such asa UAV equipped with a vision sensor and in the appropriate computer vision algorithms is one of the promising solutions to accomplish this mission and that is part of the focus of this project. Hence, the aim of this chapter is to develop and build an efficient algorithm based on aerial IR and RGB vision data, capable of automatically detecting any third-party interference and instantlyalarming the operation centre with vision and location evidence.

The algorithm proposed in this project contains three aspects, namely: detection, classification, and localization. The detection algorithm task is todetect the regions of interest of the objects geometrically based on the remaining point cloud obtained from the pipeline detection algorithm in theprevious chapter after extracting the ground and the pipeline regions. While, the classification part was proposed to recognize third-party objects by using a Machine Learning (ML) technique after being filtered into objects Inlier the ROWand then transformed into the RGB image. Additionally, the localization was proposed to locate the centre position of the region of the third-party object nce classified using the measurements of the relative centre position of the area to the sensor reference.

The reliability of this algorithm is evaluated indoors in this chapter due to the sensor and the pipeline limitations. The evaluation is estimated using a small scale sample (scene) of a pipeline and some third-party objects. The results demonstrate the capability and accuracy of recognizing the third-party objects with a high degree of detection rate and efficient processing speed. A complete overview of the algorithm is illustrated in Figure 5-1 and addressed in detail (section 5.2) below.

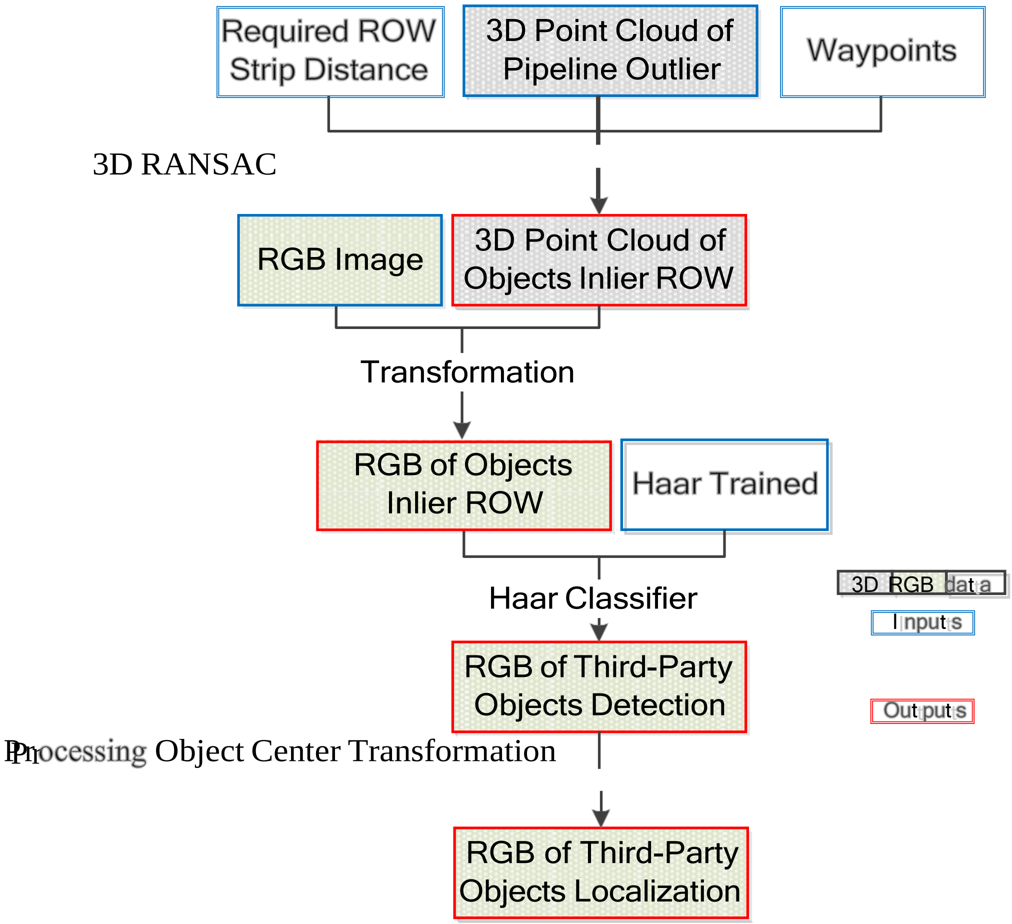


Figure 5-1: Third-party interference detection algorithm

5.2 Third-Party Objects Detection

This section describes the approach used in this project to automatically detect and estimate the position of the third-party interference objects in the vicinity of the pipeline structure in real-time using a fusion of depth and RGB images.

5.2.1 Objects Detection

The detection of objects in this project is to find any object that is found only around the Right-of-Way of the pipeline. The algorithm used is a 3D RANSAC in3D point clouds to find the objects Inlier the ROW of the pipeline and a transformation to align those data with the RGB data. The details of this algorithm are explained in the following section.

5.2.2 Point Cloud of the Objects Inlier the ROW

This section describes the filtration process of the objects that lie within the pipeline Right-of-Way strip based on the filtered Outlier point cloud of the pipeline to concentrate only on the regions of interest over the image scene.The proposed algorithm used in this project to filter those point clouds is a simple Outlier removal algorithm that removes the points that are farther than a predefined threshold distance from a line segment. This threshold refers to the strip distance of the pipeline Right-of-Way. The line segment is known by the endpoints of the pipeline already obtained in the previous chapter.

The simple Outlier removal algorithm is an iterative process which is describedin the following steps:

1) Computing the corresponding linearity of the 3D non-ground point cloud by calculating the shortest distance $d(P_{3D}, L)$ of each point cloud (P_{3D}) into the pipeline segment (L) using ecludean distance fourmuls (Boljanovic, 2006) as in Equation (5-1).

$$d(P_{3D}, L) = \frac{|\vec{V}_L \times \vec{V}_P|}{|\vec{V}_L|} \quad (5\text{-}1)$$

Where $\vec{V}_L$ is the direction vector of the line $L(P_s,\ P_e)$ and $\vec{V}_P$ is the direction vector of the point (P) to the start-point in the line (P_s).

2) Proving if this distance $d(P,\ L)$ is less than or equal to the predefined threshold (t_l), it will then append its corresponding point (P) as Inlier (P_{In}), otherwise it will be Outlier (P_{Out}) using Equation (5-2).

$$P = \begin{cases} P_{In} & if\ d(P, L) \leq t \\ P_{Out} & Otherwise \end{cases} \quad (5\text{-}2)$$

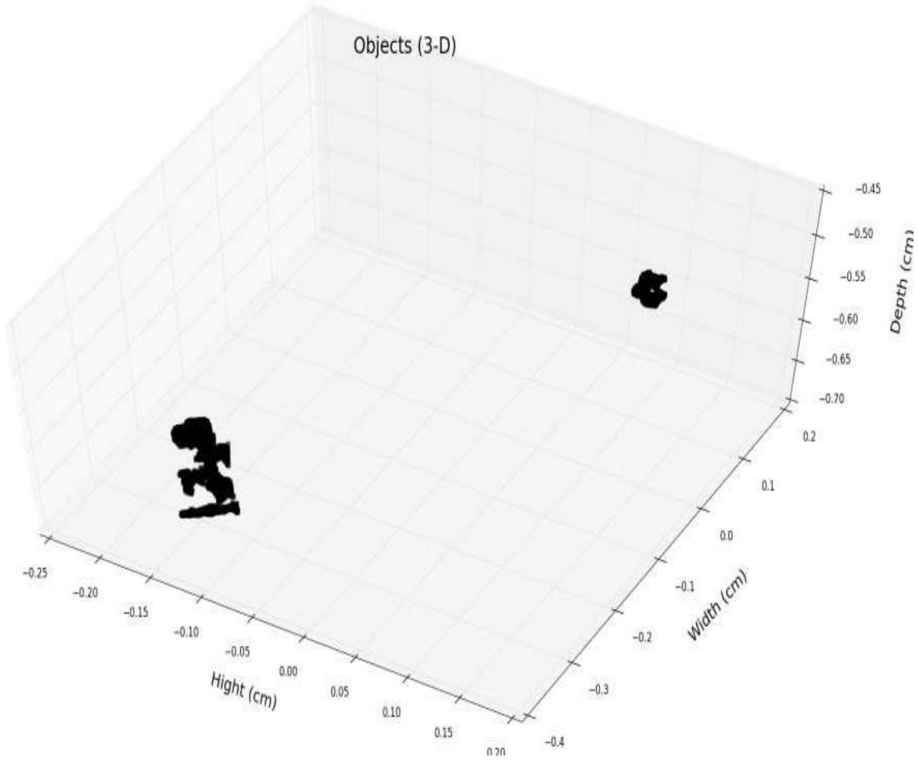


Figure 5-2: 3D point cloud of the detected objects

Figure 5-2 represents the 3D point cloud construction of the detected objectsthat are corresponded to the predefined Right-of-Way strip.

5.2.3 RGB of the Objects Inlier the ROW

Projection of the 3D point cloud map into the RGB map is used in the project to align the interested region in the IR map after being processed with the RGB map in order to achieve more processing on the RGB map or for exposing in colour. Now, the 3D point cloud map only contains geometry information while the RGB has colour information which all of them require for the purpose of this project. However, the problem is that there is a difference between the corresponding pixels in the 3D point cloud and RGB map. So, in this section, a projection of each pixel in the 3D point cloud map into its corresponding pixel of the RGB map is proposed and described.

Building a 4xN homogeneous matrix (Theoharis et al., 2008) of the given IRmap (I_{dh}) from the original 3xN matrix (I_d) is shown in Equation (5-3).

$$I_d = \begin{bmatrix} u \\ v \\ d \end{bmatrix} \rightarrow \begin{bmatrix} u \\ v \\ d \\ 1 \end{bmatrix} = I_{dh} \qquad (5\text{-}3)$$

After that, the 3D rotation (R) with 3D translation (T) matrices in one matrix are combined. Then, their homogenous matrix is built by adding the last raw data to multiply it with the homogenous matrix (Theoharis et al., 2008) of the IR data as shown in Equation (5-4) in order to transform it into a homogenous matrix forthe RGB matrix.

$$RGB_{Ih} = \begin{bmatrix} u' \\ v' \\ ' \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & T_x \\ R_{21} & R_{22} & R_{23} & T_y \\ R_{31} & R_{32} & R_{33} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ d \\ 1 \end{bmatrix} \qquad (5\text{-}4)$$

Finally, the transformed RGB data ($u^{'}$, $v^{'}$) is extracted from the homogenous RGB matrix (Ih_{RGB}) by dividing the fourth element and eliminating the third element ($d^{'}$) to get the default matrix of the RGB (I_{RGB}).

Figure 5-3: RGB data of the detected objects

As shown in Figure 5-3, it represents the result of the object detection in anRGB image after filtering out the background, the pipeline, and the Right-of- Way Outlier objects. These results will be used to classify the objects as an intrusion or not.

5.2.4 Third-Party Objects Classification

The classification of the third-party is to recognize the objects that have already been detected in (Section 5.2.1.2) based on its features. To recognize the third-party objects in real time, the Haar classifier algorithm, firstly published by Viola and Jones (2004), to detect face's features in images in real-time, will be used. The Haar classifier is one of the supervised Machine Learning classification techniques. This classifier is capable of running online and at a high detection rate based on real-time video footage.

Haar-like features consider adjacent rectangular regions at a particular location in a detection window, then sum up the pixel intensity values in each region and compute the difference between these sums (Viola and Jones, 2004). This difference is then used to categorize subsections of an image.

This classifier works as follows:

1) Create sample images (offline)

2) Haar Training (offline)

3) Performance testing of the classifier (Online).

5.2.5 Collect & Create Samples

First, this classifier requires collecting a set of positive and negative sample images to be trained with a few illuminations and pose variations that areundistorted. As much as the number of samples is increased, the performance increases. The positive samples only contain the target object and, in thisproject, some of the third-party objects are selected. Also, it is essential to provide some negative specimens, which do not contain the target object being trained for, to supply the training. A small set of positive and negative samples are shown below in Figure 5-4 and Figure 5-5, respectively.

Figure 5-4: Some of the positive samples used to train the classifier

Figure 5-5: Some of the negative samples used to train the classifier

After collecting the samples, the target object is cropped to get the positive training samples from the positive image with a proper size sub-window of the image with sensible widths and heights to include only the object. Cropping is achieved manually for each sample by using any photo editing tool; clipper image software was used for this purpose.

CHAPTER SIX: DISCUSSIONS AND CONCLUSIONS

6.1 Introduction

This chapter discusses and concludes the development on the automatic pipeline surveillance air-vehicle for pipelines and proffers recommendations for future works.

6.2 Discussion of Results

To emulate the aerial pipeline surveillance mission, a depth sensor (Asus Xtion) mounted on-board a fully-functional quadrotor UAV platform was proposed to provide depth data (represented as a 16-bits format) and RGB data (as 8-bits) of the explored environment in real-time.

The endpoints of the pipeline segment were identified accurately in real-time based on two data types that are visibility and depth. Computer vision techniques were used to develop the visible-based pipeline endpoint identification algorithm. The first step involves the image processing algorithm that enhances the vision data to improve the edge detection performance. Then, the boundaries of the features are detected in the explored environment that are assessed, constructing a candidate straight boundary for the Hough transforms method, by using a canny edge detector. After that, based on the candidate's boundaries, several straight lines representing the straight boundaries of the objects were constructed using Hough Transform method. Following this, those straight lines were filtered by comparing them with known pipeline segment endpoints in the explored area, allowing fast and accurate pipeline endpoints identification. This allows the system to reliably reject inaccurate measurements while retaining the correct pipeline detections and location. Based on the experimental results, the performance is capable of identifying the endpoints of the pipeline in real-time. As was shown in theanalysis carried-out in the project, this vision-based pipeline endpoint identification has several limitations. These include the need to have pipeline end point position information, to fuse/compare with the vision based detection system, to improve the detection rate and increase the confidence level in the system.

Therefore, the depth-based pipeline endpoints identification technique was proposed, to complement the vision based approach. This technique includes3D point cloud mapping, foreground and background extraction, boundary detection, boundaries height filtration, boundaries straight line detection, pipeline verification and pipeline endpoints estimation. First, the 16-bits depth data of the explored environment were transformed into 3D point clouds' world coordinates. Then, the foreground and background were extracted based on thetransformed 3D point cloud to extract the plane that corresponds to the ground, using RANSAC approach. Simultaneously, the boundaries of the explored environment were detected based on the 16-bit depth data using Canny method. After that, these boundaries were filtered out, after being transformed into a 3D point cloud, based on the real height of the pipeline for fast and accurate measurements using a Euclidean distance of each boundary point relative to the plane of the ground extracted before. Then, those filtered boundaries were used to detect the straight lines of the object border (Hough lines), after being transformed into 16-bit depth data, using a Hough transform method. Following that, the pipeline was verified by estimating a centre line segment, in the 3D point cloud, between any two of those Hough line segments,after being transformed into 3D. That satisfy the following statements' parallelism, distance between each other meets the constraints of the real width of the pipeline, and the foreground correspondence that the length of the centre line segment corresponds to the foreground region. The length and endpoints position of the detected centre line segments were enhanced to match the exactpipeline by extending them along their inlier foreground. Finally, the pipeline endpoints were recomputed if locally exist multi-segments intersections representing the pipeline structure in the same frame by estimating the local intersection points between the segments. This technique was tested indoors using a small scale representation of pipeline structures. The results obtained confirm the capabilities of the proposed method in identifying the pipeline endpoints at real-time.

Moreover, a computer vision technique was developed for real-time third-party interference detection based on four parameters; that are foreground 16-bit depth data, pipeline corresponds 16-bit depth data, pipeline endpoint location in the 3D point cloud and ROW proposed distance. This technique includes detection, classification, and localization algorithms. The detection was developed to detect any object, in general, within the pipeline's ROW region and consider them as a third-party interference objects. This detection is then processed by filtering the foreground depth data to concentrate on the area of interest that is expected to have a third-party interference objects present, hence reducing the processing load, and increasing the detection accuracy. This filtration includes the subtraction of the pipeline and the ROW outlier's areas. The pipeline area is filtered by subtracting the 16-bit depth matrix of the pipeline from the foreground directly. Then, the residual data, after being transformed into a 3D point cloud, is used to filter the ROW outliers area by using the Euclidean distance to reject any point outlier in the region of interest relative to the pipeline segment based on the ROW proposed distance. The detected objects are classified using Haar classifier, after data association and transformation is carried-out to sync with vision based data that are beingrecorded simultaneously. They are classified, for example into buildings, known vehicles, trees, and so on. The detected third-party interference objects were then localized based on the camera frame using a centroid contour algorithm. The performance of this technique was experimentally validated using an indoor test, in which a small-scale of expected third-party interference objects are introduced. The results show that is the developed approach and algorithms are capable of efficiently detecting the third-party interference objects quickly at a high detection rate. The advantage of using an approach as the Haar classifieris that these types of can be trained off-line first, hence significantly improving the performance and speed of the algorithm. Another advantage is that new objects can be trained and detected by the algorithm.

Finally, a waypoints-based navigation system was developed to enable the air-vehicle to fly over an online generated course using heading demand to follow the pipeline structure autonomously in real-time. The waypoints are generated based on the online identification of the pipeline endpoints relative to camera frames. The proposed autopilot system, used to track the pipeline, consists of online waypoints generation for the air-vehicle's course, change of course angle, turn anticipation distance estimation, proximity distance estimation, and heading demand calculation. The system was tested indoors. The performance was satisfactory, confirming the ability to follow the pipeline based on the received information.

6.3 Conclusion

In conclusion, the fulfilments of the research aim and objectives were successfully accomplished. The algorithms' developments and the system integration were presented and described. The performance results of the system were analysed and discussed. The limitations of the research were listed. Finally, the recommendations for the future work are covered in thesection below (section 8.6).

REFERENCES

Abkar, A., Sharifi, M. and Mulder, N. (2000), “Likelihood-Based Image Segmentation and Classification: a Framework for the Integration of Expert Knowledge in Image Classification Procedures”, *International Journal of Applied Earth Observation and Geoinformation,* Vol. 2 No. 2, pp. 104–119.

Agrawal, R., Shevgaonkar, R. and Sahasrabudhe, S. (1996), “A Fresh Look at the Hough Transform”, *Pattern Recognition Letters*, Vol. 17 No. 10, pp. 1065–1068.

Ahuja, N. (2004), “Vision Based Fire Detection”, *Proceedings of the 17th International Conference on Pattern Recognition*, pp. 134–137.

Althouse, M. and Chang, C. (1995), “Chemical Vapor Detection and Mapping with a Multispectral Forward-Looking Infrared (FLIR)”, *In Optical Sensing for Environmental and Process Monitoring, International Society for Optics and Photonics,* Vol. Vol. 2366, pp. 108–114.

Anthony, M. (2012), “100 Causes for 100 Spills”, available at: http://homosapienssaveyourearth.blogspot.co.uk/2012/06/100-causes-for- 100-pipeline-oil-spills.html, (accessed 1 July 2023).

Antolovic, D. (2008), “Review of the Hough Transform Method, With an Implementation of the Fast Hough Variant for Line Detection”, *Department of Computer Science, Indiana University*.

Anupama, K.R., Kamdar, N., Kamalampet, S.K., Vyas, D., Sahu, S. and Shah, S. (2014), “A Wireless Sensor Network Based Pipeline Monitoring System”, *International Conference on Signal Processing and Integrated Networks (SPIN),* pp. 412–419.

ArduPilot. (2013), “Open Source Autopilot”, available at: http://ardupilot.com,(accessed 4 March 2024).

ASUS. (2012), “Xtion PRO LIVE”, available at: www.asus.com, (accessed 1 October 2023).

Baer, R. (2005), *Linear Algebra and Projective Geometry*, Courier Corporation.

Baker, J.M. (2009), “Mechanical Damage”, *PHMSA-Office of Pipeline Safety*.

Behrens, T., Rohr, K., Stiehl, H.S. and Nistér, D. (2003), “Robust Segmentation of Tubular Structures in 3-D Medical Images by Parametric Object Detection and Tracking”, *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, Vol. 33 No. 4, pp. 554–561.

Bennett, C., Carter, M. and Fields, D. (1996), “Hyperspectral Imaging in the Infrared Using LIFTIRS”, *In Optical Remote Sensing for Environmental and Process Monitoring*, Vol. 2883, p. 267.

Boljanovic, V. (2006), *Applied Mathematical and Physical Formulas PocketReference*, Industrial Press.

Borrmann, D., Elseberg, J., Lingemann, K. and Nüchter, A. (2011), “The 3D Hough Transform for Plane Detection In Point Clouds: a Review and a NewAccumulator Design”, *3D Research,* Vol. 2 No. 2, pp. 32:1–32:13.

Boundary Devices. (2012), “Nitrogen 6x Embedded Single Board”, available at: https://boundarydevices.com, (accessed 15 October 2023).

Bradski, G. and Kaehler, A. (2008), *Learning OpenCV: Computer Vision with the OpenCV Library*, O’Reilly Media, Inc.

Breckon, T. and Barnes, S. (2009), “Autonomous Real-Time Vehicle Detection from a Medium-Level UAV”, *Proc. 24th International Unmanned Air Vehicle Systems,* pp. 29.1–29.9.

Burkhardt, G.L. and Crouch, A.E. (2003), *Realtime Monitoring of Pipelines for Third-Party Contact, Sensor Systems and NDE Technology Department, Applied Physics Division, Southwest Research Institute*, Texas.

Canadian Energy Pipeline Association (CEPA). (2010), “Types of Pipelines”, available at: http://www.cepa.com, (accessed 3 August 2023).

Canny, J. (1986), “A Computational Approach to Edge Detection”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 8, pp. 679–698.

Canon. (2011), “RGB Camera”, available at: http://www.canon.co.uk, (accessed 10 November 2024).

Chen, W., Cao, L., Xiao, J. and Huang, S.G. (2011), “Surface Texture Detection for Composite Material Placement Based on Vision”, *In Advanced MaterialsResearch*, Vol. 186, pp. 16–20.

Collins, R., Tsin, Y., Miller, J. and Lipton, A. (1998), “Using a DEM to Determine Geospatial Object Trajectories”, *DARPA Image Understanding Workshop*, pp. 115–122.

Cook, G.W. and Delp, E.J. (1998), “Gaussian Mixture Model for Edge- Enhanced Images with Application to Sequential Edge Detection and Linking”, *In International Conference on Image Processing (ICIP98)*, Vol. 2, pp. 540–544.

Culshaw, B. and Dakin, J. (1996), *Optical Fiber Sensors: Components and Subsystems*, Vol 3., Artech House Publishers.

Derpanis, K. (2010), “Overview of the RANSAC Algorithm”, *York University, Toronto, Canada*.

Devaud, J., Najko, S., Pierre Le, N., Maussire, C., Zante, E. and Marzat, J. (2012), “Full Design of a Low-Cost Quadrotor UAV by Student Team”,*International Conference on System Engineering and Technology (ICSET)*, pp. 1–6.

DIGI. (2014), “XBee Multipoint Wireless Networking RF Module”, available at: http://digi.com, (accessed 7 February 2014).

Dillman, D. (2002), “Method and System for Dynamic Surveillance of a Remote Object Using GPS”, *U.S. Patent No. 6,388,611*.

Dobrokhodov, V. and Kaminer, I. (2006), “Vision-Based Tracking and Motion Estimation for Moving Targets Using Small UAVs”, *In American Control Conference*.

Du, J.C. and Teng, H.C. (2007), “3D Laser Scanning and GPS Technology for Landslide Earthwork Volume Estimation”, *Automation in Construction*, Vol. 16 No. 5, pp. 657–663.

Eckbreth, A. (1977), "Laser Raman and Fluorescence Techniques for Practical Combustion Diagnostics", *Applied Spectroscopy Reviews,* Vol. 13 No. 1, pp. 15–164.

Fiedler, M., Meier, W. and Hoppe, U. (1995), "Texture Analysis of the Surface ofthe Human Skin", *Skin Pharmacology and Physiology,* Vol. 8 No. 5, pp. 252–265.

Fischler, M. and Bolles, R. (1981), "Random Sample Consensus: a Paradigmfor Model Fitting With Applications to Image Analysis and Automated Cartography", *Communications of the ACM,* Vol. 24 No. 6, pp. 381–395.

Frew, E., McGee, T. and Kim, Z. (2004), "Vision-Based Road-Following Using a Small Autonomous Aircraft", *Aerospace Conference*, Vol. 5, pp. 3006– 3015.

Fung, K.B., Fraser, D.W. and Gauthier, R.P. (1998), "Application of Remote Sensing Data for Monitoring of Gas Pipeline Right-of-Way", *In Proceedings of GIS*, pp. 6–9.

Gaszczak, A., Breckon, T. and Han, J. (2011), "Real-Time People and Vehicle Detection from UAV Imagery", *IS&T/SPIE Electronic Imaging, International Society for Optics and Photonics*.

Gonzalez, R.C. and Woods, R.E. (2003), *Digital Image Processing*, 2nd ed., Addison-Wesley, NJ.

Gopalsami, N. and Raptis, A.C. (2001), "Millimeter-Wave Radar Sensing of Airborne Chemicals", *IEEE Transactions on Microwave Theory and Techniques*, Vol. 49 No. 4, pp. 646–653.

Guadalupe-Blanco River Authority (GBRA). (2010), *Oil and Gas Pipeline Failure*, Texas.

Gupte, S., Mohandas, P.I.T. and Conrad, J.M. (2012), "A Survey of Quadrotor Unmanned Aerial Vehicles", *Southeastcon, 2012 Proceeding of IEEE*, pp. 1–6.

Haala, N. and Brenner, C. (1999), "Extraction of Buildings and Trees in Urban Environments", *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 54 No. 2, pp. 130–137.

Hasan, K., Rahman, M. and Haque, A. (2009), “Cost Effective GPS-GPRSBased Object Tracking System”, *Proceedings of the International Multiconference of Engineers and Computer Scientists*, Vol. 1, pp. 18–20.

He, Y., Wang, H. and Zhang, B. (2004), “Color-Based Road Detection in Urban Traffic Scenes”, *Intelligent Transportation Systems, , IEEE Transactions on*, Vol. 5 No. 4, pp. 309–318.

Herrera, C., Kannala, J. and Heikkilä, J. (2012), “Joint Depth and Color Camera Calibration with Distortion Correction”, *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, Vol. 34 No. 10, pp. 2058–2064.

Hjelmås, E. and Low, B. (2001), “Face Detection: A Survey”, *Computer Vision and Image Understanding*, Vol. 83 No. 3, pp. 236–274.

Hopkins, P. (2002), “The Structural Integrity of Oil and Gas Transmission Pipelines”, *Comprehensive Structural Integrity, Elsevier Publishers Penspen Ltd., Berlin*.

Hopkins, P., Fletcher, R. and Palmer-Jones, R. (1999), *A Method for the Monitoring and Management of Pipeline Risk – A Simple Pipeline Risk Audit (SPRA), Andrew Palmer and Associates*.

Hough, P.V.C. (1962), “Method and Means for Recognizing Complex Patterns.”

Huang, A.S.A., Bachrach, A., Henry, P., Krainin, M., Fox, D., Roy, N. andMaturana, D. (2011), “Visual Odometry and Mapping for Autonomous FlightUsing an RGB-D Camera”, *International Symposium on Robotics Research(ISRR)*, pp. 1–16.

Huang, S., Lin, W., Tsai, M. and Chen, M. (2007), “Fiber Optic In-Line Distributed Sensor for Detection and Localization of the Pipeline Leaks”, *Sensors and Actuators A: Physical*, Vol. 135 No. 2, pp. 570–579.

Illingworth, J. and Kittler, J. (1988), “A Survey of the Hough Transform”, *Computer Vision, Graphics, and Image Processing*, Vol. 44 No. 1, pp. 87– 116.

Jarząbek-Rychard, M., Revol-Muller, C., Peyrin, F., Carrillon, Y. and Odet, C. (2010), “Implementation of 3D Point Clouds Segmentation Based on Plane Growing Method”, *Practical Aspects of Geodesy and Cartography*, pp. 1–7.

Jin, Y. and Eydgahi, A. (2008), “Monitoring of Distributed Pipeline Systems by Wireless Sensor Networks”, *Proceedings of The 2008 IAJC-IJME International Conference.*, pp. 1–10.

Joseph, D. and Winslow, J. (1982), “Cathodic Protection Monitoring”, *U.S. Patent No. 4,351,703*.

Kanade, T., Amidi, O. and Ke, Q. (2004), “Real-Time and 3D Vision for Autonomous Small and Micro Air Vehicles”, *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, Vol. 2, pp. 1655–1662.

Kenet, B., Kenet, R. and Tearney, G. (1998), “Digital Optical Visualization, Enhancement, Quantification, and Classification of Surface and SubsurfaceFeatures of Body Surfaces”, *U.S. Patent No. 5,836,872*.

Koo, V., Chan, Y. and Gobi, V. (2012), “A New Unmanned Aerial Vehicle Synthetic Aperture Radar for Environmental Monitoring”, *Progress In Electromagnetics Research*, Vol. 122, pp. 245–268.

Kovac, J., Peer, P. and Solina, F. (2003), “Human Skin Color Clustering for Face Detection”, *IEEE*, Vol. 2, pp. 144–148.

Kulp, T. (1997), “Remote Imaging of Controlled Gas Releases Using Active and Passive Infrared Imaging Systems”, *AeroSense’97, International Society for Optics and Photonics*, pp. 269–278.

Kumar, A. (2008), “Computer-Vision-Based Fabric Defect Detection: A Survey”, *Industrial Electronics, IEEE Transactions on*, Vol. 55 No. 1, pp. 348–363.

Low, C. and Wang, D. (2008), “GPS-Based Path Following Control for a Car- Like Wheeled Mobile Robot with Skidding and Slipping”, *Control Systems Technology, IEEE Transactions on*, Vol. 16 No. 2, pp. 340–347.

Mahamuni, N.N. and Adewuyi, Y.G. (2009), “Fourier Transform Infrared Spectroscopy (FTIR) Method to Monitor Soy Biodiesel and Soybean Oil in Transesterification Reactions, Petrodiesel−Biodiesel Blends, and Blend Adulteration with Soy Oil”, *Energy & Fuels*, Vol. 23 No. 7, pp. 3773–3782.

Marinelli, W. and Green, B. (1996), "Infrared Imaging Volatile Organic Carbon Field Sensor", *Optical Remote Sensing for Environmental and Process Monitoring*, Vol. Vol. 2883, p. 245.

McRae, T. and Kulp, T. (1993), "Backscatter Absorption Gas Imaging: A New Technique for Gas Visualization", *Applied Optics*, Vol. 32 No. 21, pp. 4037– 4050.

McStay, D., Kerlin, J. and Acheson, R. (2007), "An Optical Sensor for theDetection of Leaks from Subsea Pipelines and Risers", *Journal of Physics: Conference Series*, Vol. 76 No. 1, p. 012009.

Moghimi, S. (2009), *Waypoint Naviation System Design for UAV*, Cranfield University, MSc Thesis.

Mondragón, I. and Campoy, P. (2010), "3D Pose Estimation Based on Planar Object Tracking for UAVs Control", *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pp. 35–41.

Mondragon, I.I.F., Campoy, P., Correa, J.F. and Mejias, L. (2007), "Visual Model Feature Tracking for UAV Control", *Intelligent Signal Processing, 2007. WISP 2007. IEEE International Symposium on*, pp. 1–6.

Monteiro, G., Peixoto, P. and Nunes, U. (2006), "Vision-Based PedestrianDetection Using Haar-Like Features", *Robotica 24 (2006)*, pp. 46–50.

Multi-Rotor Technology (MRT). (2012), "Gaui 500x Plus Crane (II) Aerial Platform", available at: http://www.gauimrt.com, (accessed 15 October 2023).

Murvay, P.-S. and Silea, I. (2012), "A Survey on Gas Leak Detection and Localization Techniques", *Journal of Loss Prevention in the Process Industries*, Vol. 25 No. 6, pp. 966–973.

Nagai, M., Chen, T., Kanade, T., Amidi, O., Ke, Q., Navarro-Serment, L., Collins, R., et al. (2009), "UAV-Borne 3-D Mapping System by Multisensor Integration", *Geoscience and Remote Sensing, IEEE Transactions on*, Vol. 47 No. 3, pp. 701–708.

NASA tutorial on remote sensing. (2011), "Platforms used by Remote Sensors", available at: http://rst.gsfc.nasa.gov/Intro/Part2_1x.html, (accessed 1 January 2012).

Navarro-Serment, L. (2010), "Pedestrian Detection and Tracking Using Three-Dimensional Ladar Data", *The International Journal of Robotics Research*.

Nevatia, R. and Ramesh Babu, K. (1980), "Linear Feature Extraction and Description", *Computer Graphics and Image Processing*, Vol. 13 No. 3, pp. 257–269.

Nistér, D. (2005), "Preemptive RANSAC for Live Structure and MotionEstimation", *Machine Vision and Applications*, Vol. 16 No. 5, pp. 321–329.

Oil & Gas Pipeline. (2011), "Images were produced from the following sources",available at: https://upload.wikimedia.org/wikipedia/commons/8/8f/Trans-Alaska_Pipeline_System_Luca_Galuzzi_2005.jpg; https://upload.wikimedia.org/wikipedia/commons/b/b0/Alaska_Pipeline_and_caribou.jpg; https://upload.wikimedia.org/wikipedia/commons/6/6a/Trans_Alaska_oil_pipeline_crossing_South_fork_Koyukuk_River.jpg;http://3.bp.blogspot.com/-3uXqYvz1-00/T966FMEWpcI/AAAAAAAAE-o/EcL22Ffp6nI/s1600/Pipeline.jpg; http://www.hickerphoto.com/images/1024/pipeline_snow_t2192.jpg; http://www.hickerphoto.com/images/500/alaska_oil_pipeline_t1851.jpg;

http://www.spectraenergy.com/content/inline-images/operations/New_projects_pipeline.jpg;http://petesortwell.co.uk/wp-content/uploads/2014/01/22128.jpg; http://www.calumetpipeline.net/d/img/bg/01.jpg; http://www.gregsadventure.com/wp-content/uploads/2011/08/Alaska- pipeline.jpg; https://upload.wikimedia.org/wikipedia/commons/b/ba/CSIRO_ScienceImage_6380_Section_of_the_Perth_Kalgoorlie_water_supply_pipeline_near_Merredin_WA_1976.jpg; http://www.energy-solutions.com/wp-content/uploads/2014/07/blue-sky-white-pipeline-300x200.jpg; http://yellowairplane.com/Global_Warming/images/50- Beautiful_Fall_Colors.jpg; http://albawabacdn.albawabamiddleea.netdna-cdn.com/sites/default/files/imagecache/article_headline_node_big//sites/def ault/files/im/misc/pipeline_yemen1.jpg; http://www.betterworldclub.com/static/media/uploads/keystone_pipeline.jpg; https://www.cs.uaf.edu/2011/spring/cs641/proj1/rltorgerson/pipeline.jpg; http://gs- press.com.au/utils/zimage.php?width=1200&height=800&image=images/news_articles/TAPS_pipeline_1.jpg, (accessed 1 January 2011).

Opencv dev team. (2012), "Open Source Computer Vision Library", available at: http://docs.opencv.org, (accessed 1 January 2012).

OpenNI. (2012), "The Standard framework for 3D Sensing", available at: http://openni.ru, (accessed 10 July 2012).

Owechko, Y., Medasani, S. and Korah, T. (2010), "Automatic Recognition of Diverse 3-D Objects and Analysis of Large Urban Scenes Using Ground and Aerial LIDAR Sensors", *Conference on Lasers and Electro-Optics*, pp. 1–2.

Pal, M. and Mather, P. (2001), "Decision Tree Based Classification of Remotely Sensed Data", *Paper Presented at the 22nd Asian Conference on Remote Sensing*, Vol. 5, p. 9.

Palmer A. (2002), *Appraisal of Pipeline Surveillance by High Resolution Satellite*, Vol. I.

Palmer, R., Hopkins, P., Fraser, A., Dezobry, J. and Merrienboer, H. (2004), "Evaluation of Satellite Technology for Pipeline Route Surveillance and the Prevention of Third Party Interference Damage", *Journal of Pipeline Integrity*, Vol. 3 No. 1, pp. 23–54.

Paquis, S., Legeay, V. and Konik, H. (2000), "Road Surface Textures Classification Using Opening-Based Image Processing", *International Archives of Photogrammetry and Remote Sensing*, Vol. 33 No. B3/2, PART3, pp. 685–691.

Park, H.W., Sohn, H., Law, K.H. and Farrar, C.R. (2007), "Time Reversal Active Sensing for Health Monitoring of a Composite Plate", *Journal of Sound and*

Vibration, Vol. 302 No. 1, pp. 50–66.

Piciarelli, C., Micheloni, C. and Martinel, N. (2013), "Outdoor Environment Monitoring with Unmanned Aerial Vehicles", *Image Analysis and Processing–ICIAP 2013*, pp. 279–287.

Pipeline Surveillance Company - Not for Public Release. (2012), *Aerail Footage of a Lengthy Pipeline Structure*.

Pipeline's Third-Party Interference. (2011), "Images were produced from the following sources", available at: http://www.cwplant.co.uk/wp-content/uploads/2013/01/JS220_Studio_image_front211.jpg; http://www.jcb.co.uk/~/asset/4/8889.ashx; http://senoksl.com/wp-content/uploads/JS205LC-Low2.jpg; http://modelbarn.co.nz/images/JCB- 435-Joal243.jpg, (accessed 1 January 2024).

Prasad, N. and Geiger, A. (1996), "Remote Sensing of Propane and Methaneby Means of a Differential Absorption Lidar by Topographic Reflection", *Optical Engineering*, Vol. 35 No. 4, pp. 1105–1111.

Rahman, R.M. and Afroz, F. (2013), *Comparison of Various Classification Techniques Using Different Data Mining Tools for Diabetes Diagnosis*, Vol. 2013.

Randell, C. (2010), *Operational Pipeline Integrity Monitoring Demonstration Using RADARSAT*.

Rathinam, S., Kim, Z. and Sengupta, R. (2008), "Vision-Based Monitoring of Locally Linear Structures Using an Unmanned Aerial Vehicle 1", *Journal of Infrastructure Systems*, Vol. 14 No. 1, pp. 52–63.

Rathinam, S., Kim, Z.K.Z., Soghikian, A. and Sengupta, R. (2005), "Vision Based Following of Locally Linear Structures Using an Unmanned Aerial Vehicle", *Journal of Infrastructure Systems*, Vol. 14 No. 1, pp. 52–63.

Revol-Muller, C. and Peyrin, F. (2000), "Automated 3D Region Growing Algorithm Governed by an Evaluation Function", *Image Processing, 2000. Proceedings. 2000 International Conference on*, Vol. 3, pp. 440–443.

Roper, W. and Dutta, S. (2005), “Remote Sensing and GIS Applications for Pipeline Security Assessment”, *2005 ESRI User Conference Proceedings*, Vol. 15.

Roper, E. W., and Dutta, S. (2006), “Oil Spill and Pipeline Condition Assessment Using Remote Sensing and Data Visualization Management Systems”, *Freshwater Spills Symposium, Natural Disasters, Human Error, and Equipment Failure-Causes for Major Inland Oil Spills and the ResultingMultifaceted Response*.

Saban, D. (2006), *Flight Control System Design for the Demon Vehicle*,Cranfield University, PhD Thesis.

San Martin Texmelucan. (2010), “Mexico Oil Pipeline Explosion, Fire”, *Dallas News*, available at: http://www.dallasnews.com/sharedcontent/dws/news/world/mexico/stories/ DN-mexico_20int.ART.State.Edition1.7ce5c2.html, (accessed 1 January 2024).

Schindler, G., Krishnamurthy, P. and Dellaert, F. (2006), “Line-Based Structure from Motion for Urban Environments”, *3D Data Processing, Visualization, and Transmission, Third International Symposium on*, pp. 846–853.

Schowengerdt, R.A. (2006), *Remote Sensing: Models and Methods for Image Processing, Academic Press*, Academic Press.

Shaochuang, L., Hui, S., Maosheng, X., Hongjian, Y., Tong, L. and Shunkai, L. (1999), “Positioning Accuracy of Airborne Laser Ranging and Multispectral Imaging Mapping System”, *Journal of Wuhan Technical University of Surveying and Mapping (WTUSM)*, Vol. 2, pp. 341–344.

Sivathanu, Y. and Gore, J. (1991), “Simultaneous Multiline Emission Absorption Measurements in Optically Thick Turbulent Flames”, *Combustion Science and Technology*, Vol. 80 No. 1-3, pp. 1–21.

Sivathanu, Y., Gore, J. and Dolinar, J. (1991), “Transient Scalar Properties of Strongly Radiating Jet Flames”, *Combustion Science and Technology*, Vol. 76 No. 1-3, pp. 45–66.

Smith, B. and Laubscher, B. (1999), “IRISHS, the Infrared Imaging SpatialHeterodyne Spectrometer: a New Pushbroom Fourier TransformUltraspectral Imager with no

Moving Parts”, *AeroSense’99, International Society for Optics and Photonics*, pp. 501–509.

Sohn, S., Lee, B., Kim, J. and Kee, C. (2008), “Vision-Based Real-Time Target Localization for Single-Antenna GPS-Guided UAV”, *Aerospace and Electronic Systems, IEEE Transactions on*, Vol. 44 No. 4, pp. 1391–1401.

Spoonhower, J.P., Paz-Pujalt, G.R., Sherin, J.M., Graen, M.G. and Stearns, S. V. (2006), “Detecting Natural Gas Pipeline Failures”, *U.S. Patent No.7,027,924*.

Stevens, B.L. and Lewis, F.L. (1992), *Aircraft Control and Simulation*, Wiley. Tao, C. and Hu, Y. (2002), “Assessment of Airborne Lidar and Imaging Technology for Pipeline Mapping and Safety Applications”, *Integrated Remote Sensing at the Global, Regional and Local Scale, ISPRS Commission I Mid-Term Symposium in Conjunction with Pecora 15/land Satellite Information IV Conference Proceedings*.

Tarsha-Kurdi, F. (2007), “Hough-Transform and Extended RANSAC Algorithms for Automatic Detection of 3d Building Roof Planes from Lidar Data”, *ISPRS Workshop on Laser Scanning 2007 and SilviLaser 2007,* Vol. 36, pp. 407–412.

The Business of Photonics. (2010), “Hyperspectral Touted for Pipeline Inspection”, available at: http://optics.org/news/1/4/24, (accessed 1 November 2012).

Theoharis, T., Papaioannou, G., Platis, N. and Patrikalakis, N.M. (2008), *Graphics and Visualization: Principles & Algorithms*, CRC Press.

Toshihiro, M. (2010), “Trends in Wireless Access Technologies toward Expansion of Broadband and Ubiquitous Services”, *NTT Tsukuba Forum 2010 Workshop Lectures*, Vol. 9 No. 5.

Tsai, Y., Chang, Y. and Chen, L. (2006), “Block-Based Vanishing Line and Vanishing Point Detection for 3D Scene Reconstruction”, *Intelligent Signal Processing and Communications, 2006. ISPACS’06. International Symposium on*, pp. 586–589.

U.S. Energy Information Administration (eia). (2012), *The Natural Gas Pipeline Network*, available at: http://www.eia.gov/countries/cab.cfm?fips=SA, accessed 1 November 2023).

Viola, P. and Jones, M. (2004), “Robust Real-Time Face Detection”, *International Journal of Computer Vision*.

Vosselman, G. and Gorte, B. (2004), “Recognising Structure in Laser Scanner Point Clouds”, *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 46, pp. 33–38.

Wacker, A. and Landgrebe, D. (1972), *Minimum Distance Classification in Remote Sensing, LARS Technical Reports*, Vol. 25.

Wang, Y., Shen, D. and Teoh, E.K. (2000), “Lane Detection Using Spline Model”, *Pattern Recognition Letters*, Vol. 21 No. 8, pp. 677–689.

Wang, Y., Teoh, E. and Shen, D. (2004), “Lane Detection and Tracking Using B-Snake”, *Image and Vision Computing*, Vol. 22 No. 4, pp. 269–280.

Weil, G. (1993), “Non Contact, Remote Sensing of Buried Water Pipeline Leaks Using Infrared Thermography”, *Water Management in the’90s@ sA Time for Innovation*, pp. 404–407.

Welch, C. (2010), “Trans-Alaska Pipeline”, *Alyeska Pipeline Service Company, State of Alaska, U.S. Department of the Interior Bureau of Land Management*, available at: http://www.earthlyissues.com/transalaska.htm,(accessed 1 September 2023).

Witayangkurn, A., Nagai, M., Honda, K., Dailey, M. and Shibasaki, R. (2011), “Real-Time Monitoring System Using Unmanned Aerial Vehicle Integrated With Sensor Observation Service”, *Conference on Unmanned Aerial Vehicle in Geomatics*.

Young, W.R. (1982), *The Helicopters*, Time Life Education, Chicago.

Zhang, Q., Wang, J., Liu, B., Cai, T., Qiao, L. and Zhang, Y. (2009), “Gas Pipeline Leak Detection Based on Tunable Diode Laser Absorption Spectroscopy”, *Spectroscopy and Spectral Analysis*, Vol. 29 No. 8, pp. 2017–2020.

Zongjian, L. (2008), “UAV for Mapping-Low Altitude Photogrammetric Survey”, *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XXXVII No. Part B, 1, pp. 1183–1186.

APPENDICES

A. Samples of methods coded in Python<u>Code of Canny Edge Detector Method</u>

```
def edge_depth(IR_FG, depth_image, thrs1,thrs2):
    q = depth_image
    u, v = np.mgrid[:q.shape[0], :q.shape[1]]
    IR_FG_uint8 = IR_FG.astype('uint8')
    IR_edges = cv2.Canny(IR_FG_uint8,thrs1,thrs2 )
    IR_edges=cv2.bitwise_and(depth_image,depth_image,mask=IR_edges)
    IR_edges = IR_edges.astype(np.uint16)
    dd = 1
    xyz_edges,uv_edges,uvd_edges=calibkinect.depth2xyzuv(IR_edges[::dd, ::dd], u,v)
    return xyz_edges, uv_edges, uvd_edges
```

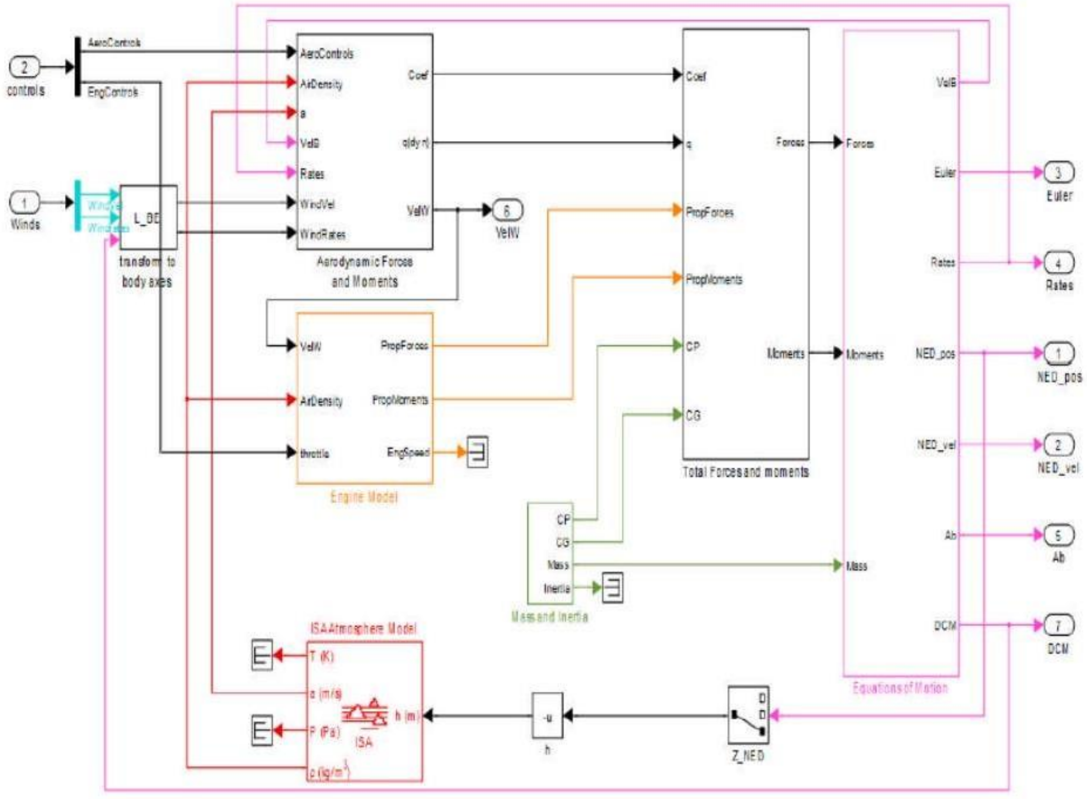
<u>**Code of Hough Transform Method**</u>

```
def Line(depth_image, edges,PointsInLine, minLineLength,maxLineGap, linesNo):
    plinesd = cv2.HoughLinesP(edges.astype('uint8'), 1, np.pi/180, PointsInLine,
np.array([]), minLineLength,maxLineGap)[0]
    print plinesd
    q = edges
    dd = 1
    u, v = np.mgrid[:q.shape[0]:dd, :q.shape[1]:dd]
    HIGHT, WIDTH = edges.shape
    imgHL1 = np.zeros((HIGHT, WIDTH))
    imgHL2 = np.zeros((HIGHT, WIDTH))
    HT = []
    for ee in range(len(plinesd)):#[:linesNo]:
        HTp = plinesd
        P1Y = HTp[:,0]
        P1X = HTp[:,1]
        P2Y = HTp[:,2]
```

```
        P2X = HTp[:,3]
        imgHL1[P1X,P1Y]= depth_image[P1X,P1Y]
        imgHL2[P1X,P1Y]= depth_image[P2X,P2Y]
        xyz_HT_P1, uv_HT_P1, uvd_HT_P1= calibkinect.depth2xyzuv(imgHL1[::1, ::1],
u, v) xyz_HT_P2, uv_HT_P2 , uvd_HT_P2= calibkinect.depth2xyzuv(imgHL2[::1, ::1],
        HT_data=       np.hstack((xyz_HT_P1,uv_HT_P1,       uvd_HT_P1,xyz_HT_P2,
u, v)

uv_HT_P2,    uvd_HT_P2))
    HT  =  np.array(HT_data)
    return HT, plinesd
```

B. Simulink Model of 6DOF dynamics of Piper Cub



C. PID controller's gains and saturation limits of SAS's of the Piper Cub UAV.

Type	PID Gains			Saturation Limit (radian)
	P	I	D	
Pitch	0.1	0	0.05	-0.13 to +0.15
Roll	1.5	0	0	-0.157 to +0.157
	1.2	0	0	
Yaw	2	0	0	-0.35 to +0.35