Real-Time Image Mosaicing and Stabilization in Unmanned Aerial Vehicle Surveillance

by

Tolga Büyükyazı

A Thesis Submitted to the
Graduate School of Science and Engineering
in Partial Fulfillment of the Requirements for
the Degree of

Master of Science
in
Mechanical Engineering

Koc University

September 2013
Koc University
Graduate School of Sciences and Engineering

This is to certify that I have examined this copy of a master's thesis by

Tolga Büyükyazı

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Committee Members:

_____

Ismail Lazoglu, Ph. D. (Advisor)

_____

Arif Karabeyoglu, Ph. D.

_____

Alper Erdogan, Ph. D.

Date: _____

ABSTRACT

Using mini Unmanned Aerial Vehicles (UAVs) equipped with camera for aerial surveillance is an application gaining popularity worldwide. However severe vibrations and fast movements coupled with size and weight constraints on the equipment that can be used in these vehicles presents a limit for effectiveness of UAV surveillance. Although there are several studies investigating image stabilization and mosaicing to provide a solution to this problem, most of them are at experimental phase requiring movement constraints or additional hardware installed on UAV. In order to provide a hardware independent solution that will work actual operational conditions, a novel real-time aerial image stabilization and mosaicing system is developed. System is developed for Baykar mini IHA which is the main UAV used by Turkish Military on rural operations and designed to enhance surveillance capabilities instead of being restricted to experimental work. Methods developed are integrated into Mobile Ground Control Station software and deployed to various regiments. In order to achieve required standards, factors affecting the performance of real-time operation in real-world conditions were analyzed in detail. Classifications of scenery encountered during flights and differences between infrared and day light images were investigated. A survey on current state of art registration algorithms is conducted and selected algorithms are tested in both in-door experiments and flight tests. Necessary optimizations and modifications are done in order to achieve a robust, accurate, real-time mosaicing and stabilization algorithm. Comparisons of several different approaches are done by using a novel mosaic quality measurement method employing printed high resolution images for “Ground Data” and 5 axis CNC for positioning. Resultant methods are able to increase effectiveness of mini UAV surveillance beyond its current limitations and can be applied to any basic UAV configuration having a Ground Control Station computer.

ÖZET

Kamera taşıyan mini İnsansız Hava Araçları (İHA) ile yapılan hava gözlemlerinin dünya çapında yaygınlığı artmaktadır. Buna karşı yüksek titreşimler ve hızlı hareketler, bu araçlarda kullanılabilecek cihazlar üzerindeki büyüklük ve ağırlık sınırlamaları ile birleşerek etkili İHA gözlemleri için bir sınır oluşturmaktadır. Bu probleme çözüm getirmek için görüntü stabilizasyonu ve mozaiklemesi üzerine çeşitli çalışmalar olmasına karşı, bu çalışmaların çoğu deneysel aşamada olup hareket sınırlamaları ya da IHA üzerine ek donanım yerleştirilmesini gerektirmektedir. Donanımdan bağımsız, gerçek operasyon koşullarında çalışacak bir çözüm üretmek amacıyla, yeni bir gerçek zamanlı hava görüntüsü stabilizasyonu ve mozaiklemesi sistemi geliştirilmiştir. Sistem Türk ordusunun kırsal operasyonlarda kullandığı ana İHA olan Baykar mini İHA için geliştirilmiş ve deneyseller çalışma ile sınırlı kalmayıp gerçek İHA gözlem görevlerini geliştirecek şekilde tasarlanmıştır. Geliştirilen metotlar Mobil Yer Kontrol İstasyonu yazılımı ile birleştirilip çeşitli birliklere dağıtılmıştır. Gerekli standartları yakalamak amacıyla, gerçek dünya koşullarında gerçek zamanlı çalışmayı etkileyen faktörler incelenmiştir. Uçuş esnasında karşılaşılan görüntülerin sınıflandırılması ve kızıl ötesi ile gün ışığı görüntüler arasındaki farklar araştırılmıştır. En ileri resim eşleştirme algoritmalarını kapsayan bir inceleme yapılmış ve seçilen algoritmalar kapalı alan deneyleri ve uçuş testleri ile denenmiştir. Dayanıklı, isabetli bir gerçek zamanlı mozaikleme ve stabilizasyon algoritmasına ulaşabilmek için gerekli olan optimizasyonlar ve modifikasyonlar yapılmıştır. Çeşitli yaklaşımlar arasında karşılaştırmalar, "Yer Verisi" olarak basılmış yüksek çözünürlüklü görüntüleri ve konumlandırma aracı olarak 5 eksenli CNC kullanan yeni bir mozaik kalitesi ölçüm metodu yararlanılarak yapılmıştır. Sonuç olarak geliştirilen metotlar mini İHA gözlemi etkinliğini şu an ki sınırlarının ötesine arttırabilmekte ve Yer Kontrol İstasyonu bulunan temel İHA yapılandırmasına uygulanabilmektedir.

ACKNOWLEDGEMENTS

Author would like to thank thesis advisor Prof. Dr. Ismail Lazoglu for his guidance throughout this study, committee members Prof. Dr. Arif Karabeyoglu and Prof. Dr. Alper Erdogan for sparing their time for evaluation of this thesis, Özdemir Bayraktar, CEO of Baykar Makina for providing resources that enabled this study, Selçuk Bayraktar, Head of Research and Development for his guidance and support, Haluk Bayraktar and Ahmet Bayraktar for their support of the project and Baykar crew; Dursun Seymenoğlu, Semra Buzlu, Mehmet Ali Güney, Alper Bodur, Mehmet Suat Kay, Erdoğan Akhan, Caner Abanoz, Dursun Çimen, Taha Cergiz, Erhan Işık, Yavuz Çilingir and Engin Gülşen for their assistance during development and testing of the system. Also special thanks to Kristof Richmond for sharing his experience and knowledge of creating real-time mosaics. The authors would also like to thank to Gökhan Erünlü and Zeynep Seçil Yüksek for their support and helpful discussions.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

x', y'	image coordinates
x, y, z	object coordinates
λ	ratio of image to original object
f'	image forming distance at a pin-hole camera
n_1, n_2	refraction indexes of medium and refracting surface
α_1, α_2	refraction angles
d_1, d_2	intersection distances
R	diameter of refracting surface
f	focal distance
z, z'	distance of object and mage from thin lens
n_l	refracting index of thin lens
u	image displacement vector
x_i	ith position vector
I_0, I_1	base and target image
E_{SSD}	Sum of Squared Differences metric
E_{SAD}	Sum of Absolute Differences metric
w_0, w_1	windowing functions for base and target image
E_{WSSD}	windowed SSD
A	area of window
RMS	Root Mean Square
E_{CC}	cross-corelation metric
E_{NCC}	normalized cross-corelation metric
$\overline{I_0}$, $\overline{I_1}$	average intensity values of base and target image

N	number of pixels
$E_{WCC}(u)$	windowed cross-corelation metric
$E_{PC}(u)$	phase correlation metric
$C_x(x)$, $C_y(x)$	partial derivatives of Gaussian function
$G_{\sigma_d}(x)$, $G_{\sigma_i}(x)$	Gaussian based derivative and integration filters
$B(x)$	first order Gaussian derivative operator
$A(x)$	gradient image
$\lambda_{0,1}(x)$	Eigenvalues
k_H	Harris Detector coefficent
$L(x, y, \sigma)$	space-scale function
$D(x, y, \sigma)$	Difference of Gaussian function
k	scaling factor
$H(x, y, \sigma)$	Hessian Matrix
$S_{p \to x}$	FAST segmentation test
d, s, b	darker, similar, brighter results
I_p, $I_{p \to x}$	intensities of central pixel and tested pixel
$\tau(p; x, y)$	BRIEF test
$p(x)$, $p(y)$	pixel intensity at x and y locations
$f_{nd}(p)$	BRIEF descriptor
n_d	number of dimensions
x', y'	transformed corrdinates
ε	orientation factor
θ	rotation angle
t_x, t_y	translation in x and y directions
H_E	Euclidean transform

R	rotation matrix
s	scaling factor
H_S	similarity transform
H_A	affine transform
ϕ	rotational component in affine transformation decomposition
D	diagonal matrix in affine decomposition
H_P	perspective transform
v	perpective term
H_{Per}	perpective terms matrix
r_i	residuals at ith RANSAC trial
$\tilde{x}_i$, $\hat{x}'_i$	transformed and matched keypoints
P	probability of success
kr	random samples
p^{kr}	probability of all random samples being inliers
S	number of trials
e_{out}	outlier percentage
n_{out}	number of outliers
n_{total}	total number of matches
e_{avg}	average pixel error
$m(x, y)$	mosaic image pixel
$f(x, y)$	transformed frame pixel
E_{drift}	drifting error matrix
F_{final}	total final transformation matrix
α	rotation angle
d_x, d_y	translation components
d_{xorgin}, d_{yorgin}	translation component of orgin at starting point

d_{xfinal}, d_{yfinal}	translation component of orgin at the end of path
C_{orgin}, C_{final}	center point at the start and end of path
e_{drift}	drifting error
$n_{difference}$	difference between number of found points and ideal points
n_{found}	number of found points
n_{ideal}	number of ideal points
n_{max}	maksimum boundary of number of found points
n_{dist}	number of step points
k_{hes}	ratio of number of step points to difference points
h_{update}	Hessian threshold update value
h_{step}	Hessian threshold update step
$h_{current}$	current Hessian threshold
n_{min}	minimum boundary of number of found points
t	unit time
$F(t)$	frame to frame transformation
$R(t)$	resultant total transformation
$T(t)$	mosaic translation transformation
$M(t)$	total transformation
α'	angular difference between filtered angle of previous frame and normal angle of current frame
α''	filtered angle
k	damping coefficent
$K(t)$	filtered rotating mosaic transformation
$d(t)$	distance between frame center and mosaic center
$r(t, d(t))$	filtering function

k_s, c	filtering scalar coefficents
n, m	filtering power coefficents
$d_{x,y}$, $d'_{x,y}$	orginal and filtered frame center to mosaic center coefficent
$S(t)$	filtered stabilization transform

Chapter 1

INTRODUCTION

Increasing costs of conventional aircrafts coupled with advances in autonomous vehicle technologies resulted in a rising interest toward extended usage of Unmanned Aerial Vehicles (UAVs) for tasks that were traditionally conducted by manned vehicles. One of the popular approaches is using mini UAVs equipped with a camera for surveillance missions. Mini UAVs provide cost efficient, fast, flexible low altitude aerial surveillance and can be deployed easily by using a mobile Ground Control Station with fewer crew requirements.

1.2 Statement of the Problem

While employing UAVs for surveillance is a popular idea, it also creates additional considerations due to characteristics inherent in these vehicles. Especially for mini UAVs size and weight constraints create several effects that severely distorts UAV footage even making aerial surveillance impractical. Some of the most prominent of these effects can be listed as follows;

- Because of the vibrations of UAVs, it is hard to stabilize camera view in one location and at low altitudes with a substantial amount of zoom these vibration effects become predominant making camera footage unusable. Although there are mechanical stabilization gimbals for larger aircrafts, such systems have a

high cost and can not be installed to mini UAVs due to size and weight constraints.

- Small size of the mini UAVs make them very receptive to changes in air currents. Sudden wind flows create sharp image movements, further distorting the footage.
- Also in an actual surveillance mission, it is often required to zoom in an area to view the details of objects of interest. Especially in low altitudes, this situation creates a reduced field of view due to optical constraints, decreasing environmental awareness of the operator.
- Fast movements and turns of the mini UAV with a reduced field of view make it difficult for user to follow, easily causing disorientation.
- All of the image movements described above may result in motion blur if appropriate shutter adjustments are not made. On the other hand decreasing shutter time decreases the amount of light received by camera sensor which may decrease signal to noise ratio in low light conditions.

There are also other distortion factors inherent in mini UAV surveillance footage such as interference effects due to communications, quality of light weight, small optics and imaging devices etc. These effects will be discussed briefly but are not addressed in this study.

Although it is a very tempting idea to use a portable mini UAV for surveillance missions instead of deploying a full scale manned aircraft, distortion effects described above severely limits the application areas and effectiveness of mini UAV surveillance. Several studies were conducted in order to address these issues by using image mosaics [1,2]. Specialized hardware can provide real-time processing speeds but on the other hand it may not be suitable for installation in very small sized mini UAVs [2]. Constraining the camera orientation and UAV movements can result on efficient mapping of the area but

many actual surveillance missions require flexibility for better examination of the objects of interests [1]. There is a need for robust, real time, low cost aerial mosaicing and stabilization systems that does not require any additional hardware and is able to capture the unstructured camera and UAV motions.

In this thesis, a novel aerial mosaicing and image stabilization system, achieving real time and near real time processing speeds by optimized algorithms, working on standard ground station without requiring any additional hardware in mini UAVs is presented. System described here is able to work on a wide variety of illumination and terrain conditions with both day light and infrared cameras and were tested on real world working situations by using an actual on-service UAV.

1.2 Related Works

A good example mosaicing and stabilization of aerial images acquired by using a mini UAV can be found at [1]. Study presented in [1] is part of a larger ongoing project for use of UAVs in wilderness search and rescue called WISAR [3] which employs several mosaicing and stabilization approaches. In [1] authors describe an aerial image stabilization and mosaicing application having three different modes namely, stabilization, mosaic and stabilized mosaics. A Harris Corner detector based feature matching algorithm was used to establish frame to frame alignments. In mosaicing mode, frame correspondences are utilized to construct local mosaics and any frame that is out of the viewer screen is deleted. In stabilization mode a full view of the transformed frame used with a spline fitted to frame centers in order to provide camera movement following and filter out the vibrations. A combined mod called stabilized mosaic mode is employed to provide spline smoothing principle to mosaic images. A survey study to measure effectiveness and benefits of these algorithms is also included. Although idea of using aerial mosaics in order to handle same

distortions present in UAV surveillance remains same, apart from mechanics of classical mosaic mode, displaying modes and methods used in registration of frames are different compared to study presented in this thesis. Moreover, while [1] focuses on mapping with a forward-moving UAV and a downward pointing camera, in this study, capturing unstructured image motions and avoiding any restriction on UAV and camera movement was considered as a requirement for effective UAV surveillance. Study at [1] shows that aerial image mosaics greatly enhance human perception in search and rescue missions. Dynamic mosaic concept, employed at [1], which is updating mosaics with every new coming frame in order to present most recent information, can be found [4] and was also employed in this study.

In [2] an integrated aerial surveillance system which uses a video processing hardware installed on UAV is presented. System uses a video processing card and consists of front-end and back-end processing. In front end processing, captured images are processed by using this card on the plane in real-time and relevant information sent to ground control station for further processing. Back end processing is conducted on Ground Control Station offline. Several features like mosaicing, motion tracking and video compression were integrated into one processor [2]. On the other hand, one of the main aims in this study was to develop methods that will not require any specialized hardware which may not be suitable to use in mini UAVs because of the size and weight restrictions.

There are several other studies presenting different approaches or utilizing different systems. A good study on real-time mosaicing using autonomous vehicles can be found at [5] where author uses an Autonomous Underwater Vehicle (AUV) to obtain mosaic images of sea floor. Author describes an efficient system can create image mosaics as maps while navigating into unknown areas. System uses data fusion of several sensors and also uses computer vision data as navigation information [5]. [6] provides a comprehensive study on construction of mosaics by IMU and GPS information. A low flying UAV was employed

for image acquisition and offline bundle adjustment utilized in order to obtain satisfactory results. While these two studies employ additional sensor data, study presented here aimed to be only depended on processing of camera frames in order to increase hardware independence. In [7] authors provide a complete study on combining underwater mosaics with 3D data in order to construct maps with depth information. On the other hand main aim of constructing mosaics in this study was to real-time enhancement of surveillance during missions. A real-time aerial image processing example can be found at [8] where authors process 1024 by 768 pixel images at 30 Hz by employing a GPU which is claimed to be suitable for integration onto UAVs. On the other hand for the UAV used in this study and many other small sized UAVs, volume of the vehicle body is so small and weight of the vehicle is so critical that such kind of integration is not possible.

Stabilization of the video frames is also a well-studied problem. One characteristic problem with classical stabilization methods is that transformed images are cropped resulting in information loss and a distracting black background view. Several studies were conducted to address this issue. A notable study is presented in [9] where authors describe a background preserving stabilization method. On the other hand main challenge of stabilization mode in this study was to develop methods that will filter vibrations and unwanted sudden movements while still following desired camera motions. For background preservation, a Hybrid mode was developed utilizing mosaicing principle and any method that would add additional computational burden was intentionally avoided for real-time processing.

More general surveys on constructing mosaic images from a sequence of frames can be found at [10], [11] and [12]. In [10] authors describe basics of image mosaicing and describe various methods of image indexing in order to cover different information. In his study Anandan [11] provides a good overview of mosaicing concept and presents taxonomy to classify different types of mosaics. He describes two categories of video

mosaics namely static mosaics and dynamic mosaics. In static mosaics, frames in a video are processed to determine frame by frame relations. Later all frames are blended into a single mosaicing image with a blending method of choice. This method provides an efficient representation of a video sequence but dynamic events are generally lost. He proposes a second approach called dynamic mosaicing where frames in a video frame are combined in to a sequence of mosaics instead of single one. Coordinate system for representation can be chosen either as fixed as static mosaics or as the coordinate system for the most recent frame. He also describes temporal mosaic pyramids constructed from static and dynamic mosaic sequences where coarse level represents the integration of all frame and finest represents single frames. For sequence alignment, Anandan [11] describes three approaches namely frame to frame alignment, frame to mosaic alignment and mosaic to frame alignment. Frame to frame alignment requires only one pass from frame sequence but has an error accumulation problem. This approach can be further refined by establishing transformation directly between individual frames and mosaic image. Mosaic to frame alignment on the other hand preserves the coordinate system of the individual frames.

A very detailed and wide range survey on image registration and stitching can be found at [12]. Survey covers different motion models, two general approaches on image registration and different methods used in this approaches, global mosaics and image blending. Author also provides discussions on several advantages and disadvantages of different registration approaches compared to one other [12].

Although image mosaics are widely investigated by many authors, there is a lack of unified metrics and methods to examine the quality of the results. Several authors addressed this issue by proposing evaluation criteria and methodologies [13,14,15]. In [13] authors us a “Virtual Camera” to generate realistic artificial images from a base image and use base image as “Ground Truth” information for evaluation of mosaicing algorithms.

Average intensity differences, average geometric differences between control points and sum of missing and redundant pixels are used as error metrics. [14] also employs generation of realistic artificial images principle and measure mosaic quality based on coverage and difference between base and mosaic images. [15] proposes a wider range of metrics namely, entropy, clarity, registration error, peak signal-to-noise ratio and structural similarity, and an assessment methodology for evaluation. While all the methods described above utilizes creation of artificial images, this approach only employed at the early stages of development. Later on methods using a CNC for simulate UAV motions and printed aerial images to provide "Ground Truth" data were developed for better simulation of surveillance conditions.

Speed and accuracy of a mosaicing algorithm mostly depends on its frame registration component. A good survey and comparison of various image registration algorithms can be found at [16]. Also detailed descriptions of the algorithms that were investigated in this study can be found at their respective references [17,18,19,20,21].

1.3 Contributions

This study presents a real time real world application that is designed to be used in practical aerial surveillance tasks in critical missions. In order to achieve this several key advancements needed to be made. These contributions are listed in the following subsections.

1.3.1 A Practical Real-Time Real-World System That Enables Video Mosaicing, Stabilization by Using an Ordinary CPU

Image mosaics and image stabilization is a hot research area having a wide variety of implementations [12]. There are ongoing studies for using aerial image mosaics with UAVs [1,3]. Also there are several video processor cards that are designed for such kind of purposes [2]. On the other hand these systems require additional hardware integrated to

UAVs which bring in additional cost and weight and makes it impractical for use in mini UAVs. Other studies in using image mosaics in aerial surveillance have attained successful results but are still in experimental phase [1]. In this study a system that is designed and integrated into actual on-service UAVs is presented. System presented here is able to work in several modes namely mosaicing, stabilization, hybrid and rotating mosaics, and can be used with both thermal and visible spectrum cameras. It is tested in actual work environments that both urban and rural images were used and in different levels of light and also night missions. Developed algorithms are able to work with an ordinary CPU making system hardware independent and are optimized to achieve real-time processing speed without a significant loss of accuracy. Furthermore since developed software works on ground station CPU's, it does not require any additional hardware to UAVs, This aspect both reduces the weight of the UAV and makes developed system practical to use with the existing systems. Also since it does not require any additional hardware, it is also possible to use it with existing ground stations. This performance was achieved by using heavily optimized algorithms designed to use with UAV aerial images and imaging conditions.

1.3.2 An Optimized Image Registration Algorithm to Achieve Real-Time Image Registration without Significant Loss in Accuracy

Image registration is a relatively well developed field of Computer Vision [12]. On the other hand since registration algorithms are generally at the base of more complex applications, better and faster algorithms are always a hot research topic. For the application presented here a registration method that will work well with aerial images acquired from a fast moving, sudden turning mini UAV was required. In this research several existing algorithms were tested. Because of the movements of UAV a detector robust to scale variations and rotation is required. Unfortunately, those types of detectors are computationally expensive to be used in real-time applications. SIFT is such kind of an algorithm [19]. A modified version of SIFT called SURF were developed showing great

speed increases without loss of accuracy [20]. On the other hand original version of SURF still proved to be quite slow for the application presented in this study. So SURF was optimized in order to produce a "Modified Algorithm" that can meet the speed requirements without much loss of accuracy in the working range of a general UAV imagery. With additional optimizations in other parts of the application, real-time processing speeds were obtained without degrading mosaic quality.

1.3.3 Aerial Image Mosaics and Image Stabilization by Using Infrared Images

Since many of the actual surveillance in critical missions are conducted with thermal cameras at night, study presented in this thesis was required to cover such conditions. Surprisingly, in literature review research papers on image mosaics and stabilization using infrared images were not encountered. This may be due to the fact that thermal images gradient nature is much more different than regular visible light images, algorithm developed with visible light images are working poorly in infrared images. With several modifications developed algorithm configured to enable working with infrared images

1.3.4 A Novel Mosaic Quality Measurement Method that Utilizing Real World Data

Although there is a developed literature on image mosaics for a wide variety of applications, there is a lack of literature on measuring mosaic quality. There is not a unified metrics or standards for measuring quality and most the quality data is based on subjective examination and method specific metrics. Several authors addressed this problem by using artificially created images to obtain accurate "Ground Truth". On the other hand using artificial images does not provide the real-world working conditions and produce additional concerns. In order to address this problem, a novel mosaic quality measurement method that is utilizing a 5 axis CNC and printed aerial photos for positioning and ground truth

data is developed. Path following tests utilizing proposed method showing interesting results on the performance of mosaicing algorithm using different state of art registration methods and “Modified Algorithm” is provided.

1.4 Outline

Chapter 2 provides necessary background on imaging and image registration. A basic classification of the aerial images that is used is explained. Fundamentals of image forming and their application to imaging using an UAV, aberrations and points of concern, CCD arrays and video stream formats are described briefly to form foundations of the concepts that are presented at later chapters. A survey on current state of image registration algorithms and their working principles is provided.

Chapter 3 begins with a brief review of the geometric transformations and a discussion on their ability to satisfy mini, UAV surveillance requirements. Algorithms used in geometric transform estimation and different methods for image blending are described. A brief overview of the image mosaics concept and a novel methodology to measure mosaic quality are presented.

Developed system and its working principles are presented at Chapter 4. System and test setups, image acquisition and preprocessing phases, modifications done at the image registration part and reasons for the modifications, adaptive algorithm, transform estimation and mosaic construction methods, operating modes are described in detail. A study on working with infrared images and their difference to day light images also presented.

Chapter 5 provides the results of various flight and in-door tests. Practical benefits of using mosaics in UAV surveillance, detailed comparisons of several state of art image registration methods and modified registration algorithm, images of the flight tests, notable

conditions and errors that were encountered during flights and tests with infrared mosaics presented. A short conclusion on the study and future work also provided.

Chapter 2

VISION FOR UAV SURVAILLANCE

2.1 Aerial Images

Success of the image registration algorithms depend on the nature of the image structure such as distribution of the gradients and characteristics of the image motion between successive frames. Image registration algorithms which have a long history of development usually tested and developed for more generalized image registration tasks. For this study, it was crucial to find and develop methods that work adequately with the scenery encountered in the operational range of mini UAV.

Two major classifications are done for scenery based on observations during flight tests. It was observed that scenery encountered by the UAV ranges from images with high intensity variations such as urban areas to relatively low intensity variation images such as grasslands in rural areas. Taking this observation into account, aerial images throughout this study classified into two main categories as high variation urban images and low variation rural images. It should be noted that not all cases fit into these categories but this classification is useful for understanding of algorithm performance.

One important aspect that was observed in the flight tests was the performance of the registration methods working with rural terrain images. Methods that were developed by assuming sharp and rich gradient nature for the target image performed poorly when encountered with low smooth gradient rural images. On the other hand operational range of

the tested UAV mostly covers rural border areas so developed methods are required to be working on these conditions.

Two example aerial images can be seen at Figure 2.1. As it can be seen form Figure 4 (1) scenery like urban areas provide a more favorable gradient structure for feature detection and extraction having apparent intensity variations throughout the scene. On the other hand in Figure 2.1 (2) high intensity variation areas although present, relatively rare and overall intensity distribution is less fluctuating. When two dimensional Fourier transforms of the scenery is compared, it can be seen that urban image has stronger components in higher space-scale levels while strong frequencies in low variation image are more clustered around the center. In flight tests it was observed that such kind of scenery produces fewer number of apparent feature points compared to urban scenery causing image registration methods to perform poorly. Detectors employing image pyramids formed by scaling an image with increasing scaling factors are more suited to extract feature from high level of the pyramid using slow varying gradient structure creating opportunity to detect additional points but these methods generally result in additional computational burden.

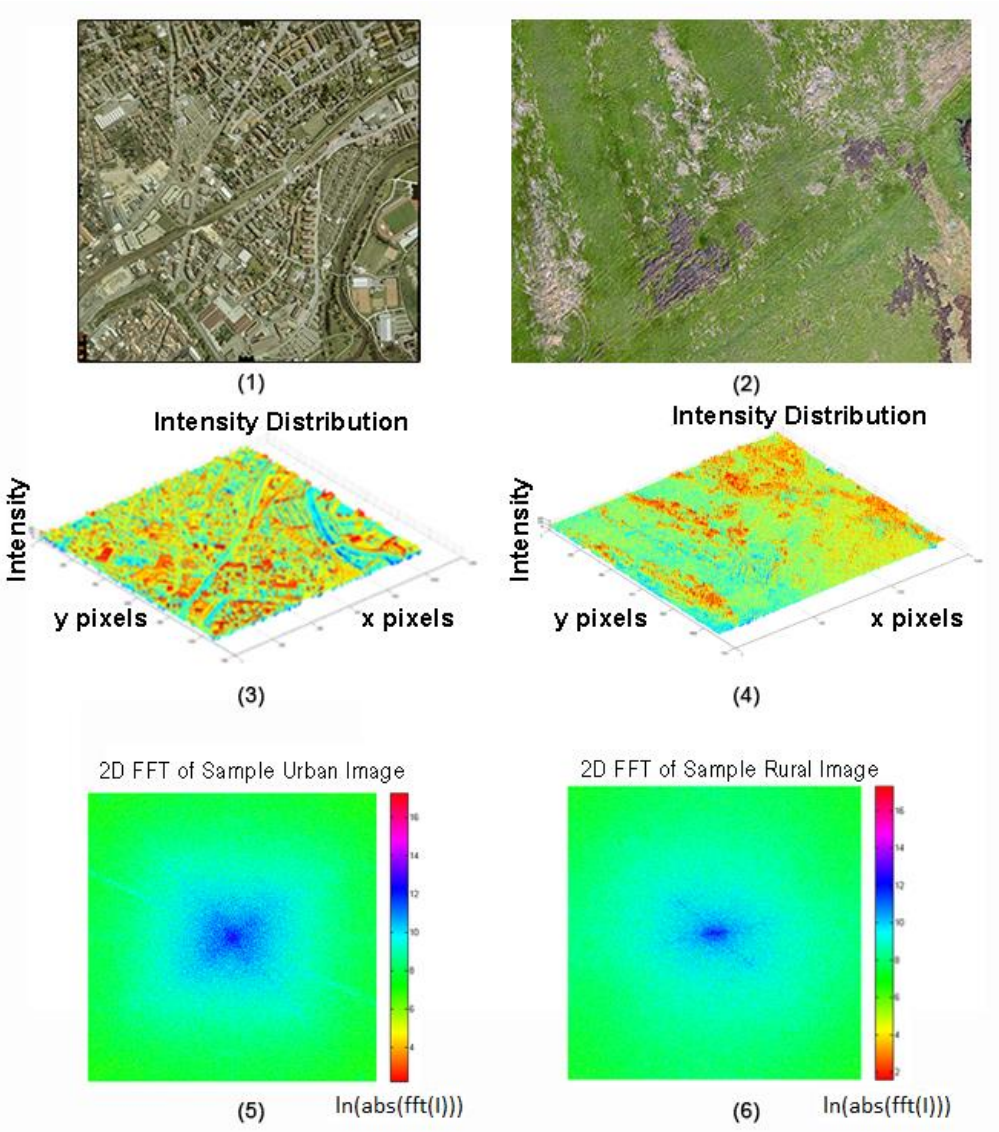


Figure 2.1: Two samples of high intensity variation and low intensity variation images. (1) ,Amberg scan retrieved from Esri ArcGIS raster data, showing a typical urban image while (2) ,Lemon Fair Map retrieved from MapKitter, is a sample scenery frequently encountered in rural area surveillance. (3) and (4) shows 3D intensity variation graphs and (5) and (6) are the graphs of 2D FFT of two images showing stronger components in higher frequencies. Printed versions of samples such as these were used in in-door tests and observed to effect algorithm performance.

In order to build an aerial imaging and mosaicing system, understanding of the imaging process is essential. In this section, several basic concepts of imaging were summarized in brief and their effects on aerial imaging using small UAVs were described.

2.2 Imaging by using UAVs

In order to have a better understanding of mini UAV surveillance conditions, a basic understanding of imaging models is required. Most basic and classical model of imaging is pin-hole camera model. In this model light from an object goes through the pinhole and creates the upside down image of the object on an image plane behind the hole. In idealized model every point in the image is created by exactly one light ray coming from one point in the object and passing through the pin hole. In reality since the size off the pin hole is finite, every image point is actually produced by a cone of light rays coming from a finite area on object. This model generally provides a good approximation to imaging.

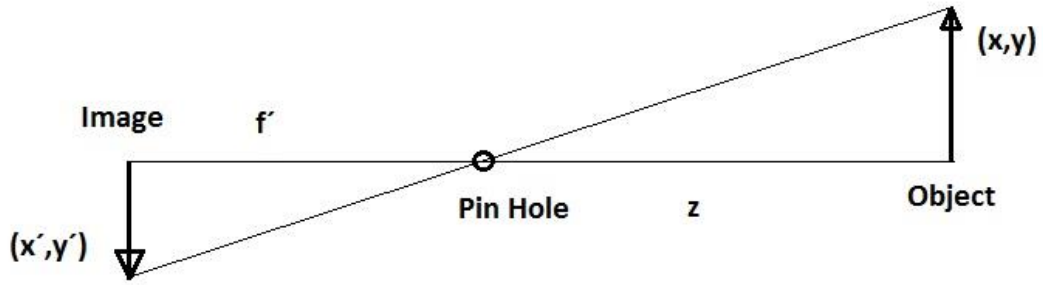


Figure 2.1: Pin-hole camera model

This model is called pinhole perspective projection model. Perspective projection creates inverted images. An uninverted equivalent of the actual image called "virtual image" can be considered lying in front of the pinhole at the same distance as image plane. An apparent effect of the perspective projection can be seen by considering several objects in front of the pinhole. Size of the images of the objects will be affected by their distance to the pinhole. If two lines parallel to each other are considered, this effect can be seen more obviously since they will intersect in the "horizon" line.

Positions of points in an image can be calculated by the following equations;

$$\begin{cases} x' = \lambda x \\ y' = \lambda y \\ f' = \lambda z \end{cases} \Leftrightarrow \lambda = \frac{x'}{x} = \frac{y'}{y} = \frac{f'}{z} \tag{2.1}$$

$$\begin{cases} x' = f' \frac{x}{z} \\ y' = f' \frac{y}{z} \end{cases} \tag{2.2}$$

where x' and y' represents the image coordinates and f' represent the image plane distance from pinhole.

There are other models for imaging such as affine projection models using different approximations but are not covered in this study since this basic model is enough for understanding the effects of UAV motions on surveillance footage at most of the time. A good reference can be found at [22].

Another concept that needs to be briefly discussed is the effects of lenses. There is a limit for shirking the size of the pinhole in order to increase sharpness of images because of the diffraction effects. Also shrinking the hole reduces the amount of light while increasing its size give brighter but blurred images. Real cameras employ lenses in order to gather enough light and focus then into a small point so that image can be both sharp and bright. Lenses uses refraction of light to bend the light rays and equations governing this phenomenon is as follows;

$$n_1\alpha_1 \approx n_2\alpha_2 \Leftrightarrow \frac{n_1}{d_1} + \frac{n_2}{d_2} = \frac{n_2 - n_1}{R} \tag{2.3}$$

where n_1 and n_2 are refraction indexes, α_1 and α_2 are angles from original and refracted ray, d_1 and d_2 are intersection points to the optical axis of original and refracted rays and R is the diameter of the refracting surface.

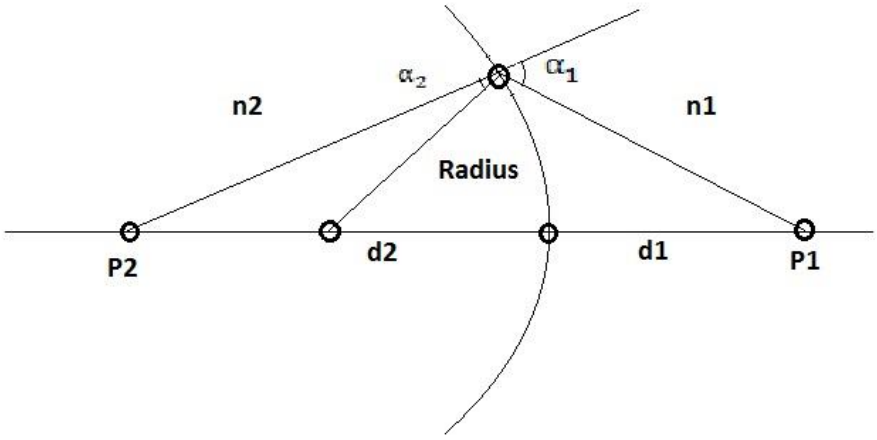


Figure 2.3: A diagram showing refraction equations.

If spherical surfaces adjacent to each other with an infinitesimal thickness between them surrounded by vacuum are considered, "thin lens" model is obtained. Thin lens model is a very useful approximation in order to drive imaging equations. Equations are as shown below;

$$\frac{1}{z'} - \frac{1}{z} = \frac{1}{f} \tag{2.4}$$

$$f = \frac{R}{2(n_l - 1)} \tag{2.5}$$

where f is the focal distance of the lens and point lying at f distance in optic axis of lens called focal point. n_l represents the refraction index of the lens and R is the radius of the lens surface curvature. z' and z represent the distance of image and object from the lens respectively. Any light ray coming parallel to this axis is refracted to pass from this point.

An image will only be sharp if image plane is positioned at a distance satisfied the above equation. In practice however, image will likely to retain sharpness at a range of distances that are referred as depth of field.

This simplified equations area very useful and simple to use and provides a good approximation. Unfortunately real lenses are subjected to more complex laws. A more realistic approximation is the thick lens model governed by same refraction equations except for an offset. When lens thickness is considered, refraction of the light traveling through the lens should be taken into account. In that situation on the light rays through the optical axis remain undeflected. On the other hand for an understanding of the imaging conditions governing UAV surveillance, simplified equations are generally sufficient.

An important point in aerial imaging is the UAV coordinates and movements and gimbals which altogether determine the viewing point of the camera with respect to terrain. Most of the literature about image registration generally deals with fixed point cameras or stationary cameras with rotational degrees of freedom capture unstructured scenes. For small UAV aerial mosaics one would be dealing with unstructured sudden movements and camera gimbals affecting the viewing point with respect to train to be imaged.

Gimbals are in use for rotational degrees of freedom from a long time. In modern aerial imagery camera gimbals occupies an important role since surveillance of ground objects are general mode from far away distances that small misalignment in rotation or any kind of vibration can have a very obvious effect on the images of object. Because of this factor aerial imaging gimbals are designed to be extra accurate and stable and also can cost between 500.000 dollars to several million dollars [2].

Figure 2.4: FLIR Star SAFIRE 380 HD Gimbal taken from user's manual.

If the structure of a camera gimbals are examined it is seen that it general consist a frame fixed at an aircraft having two degrees of freedom. Other gimbals with less or mode degrees of freedom can also be found. These two degrees of freedom determines the cameras orientation with respect to aircraft and hence determines the viewing angle. On the other hand viewing point and orientation of the camera is also determined by coordinates and orientation of the UAV. Since the resultant coordinate with respect to scene will be a combination of all of these.

Since an UAV is a moving object, camera position changes with respect to time. Also any sudden movements or distortion on the UAV or the gimbal system affects the viewing scene. By considering these movements, terrain structure, different surveillance paths and orientation and environmental effects, for an aerial mosaicing system to work without constraining UAV movements and operating conditions, developed methods required to be able to handle unstructured image motions having rotational, translational and scaling degrees of freedom.

2.3 Points of Concerns, Aberrations and Camera Calibration

Although approximated models described in Section 2.1 are good for calculations, real lenses are subject to different aberrations that distort images. There are many kinds of aberrations that degrade image and detailed discussion of this is beyond the scope of this study. However some of them need to be summarized because of their handling way in aerial imagery and mosaics.

There are five types of primary aberrations caused by difference between first and third order optics. These are coma, astigmatism, field curvature, distortion and spherical aberration. Difference between the par-axial image of an object and intersection point of the ray coming from same object is called longitudinal spherical aberration of the ray. Difference between the optical axis and the interception of the light ray coming from the object at the image plane of the par-axial image is called transverse spherical aberration of the ray. So as a result rays coming from a point at object pass through a spherical area instead of a single point and this area is called circle of least confusion which results in blurring of the image. Other aberrations also blur the image apart from distortion which changes the total shape of the image because of different areas of the lens have different focal points.

Also these aberrations considered so far assumes single wavelength but in actually cases lenses have slightly different responses to different wavelengths. This situation creates a phenomenon called chromatic aberrations. Intercepting of refracted rays of different wavelengths at different points of the optical axis is called longitudinal chromatic aberration and forming of different circles of confusion by rays of different wavelength at the same image plane is called transverse chromatic aberration.

There are several ways of dealing with aberrations. In practice compounding several lenses in a well calculated order can minimize aberrations. This is the case in the professional cameras with large camera objectives. Use of several lenses causes phenomena called vignetting which decreases image brightness. Also such kind of lens system general requires a large space and increase weight.

Another common aberration correction is used for correcting distortions is use of image processing. It is a well applied practice in a camera calibration and response of the camera to a chess board is first captured. Since a chessboard is composed of equal squares, any distortions can be easily determined. Then a transform matrix which is function of distance of a pixel from the center of the image is estimated and all pixels of the image are corrected with the estimated matrix.

For the system discussed here a camera that should be light weight and small in size in order to be able effectively install it onto a small UAV is needed. Any larger optical apparatus will not be suitable, so aberration minimization by using lens system is impractical. Calibration of distortions by using software is a viable alternative. On the other hand this requires additional transformation through all pixels of image which brings in additional computation burden which is not very desirable for a real time application.

In experiments it was observed that distortions become significant when objects nearby are being imaged. On the other hand as a distinctive property of aerial imaging, in the operational range of the system described in this study, objects of interest are generally located far away and distortion affects are not significant. So it was chosen to trim a small portion around frame edges and discard any kind of calibration transformation bringing additional computational cost. One should note that for a mosaicing application, severe distortion effects can ruin the image registration and compounding parts resulting in total malfunction of the algorithm.

2.4 CCD Arrays, Image Grapping and User Interface

In order to have a better understanding of the conditions affecting aerial imagery, a brief discussion of capturing of images is required. In practice several methods were used for image capturing such as photographic films, vidicon vacuum tubes, and charged coupled devices (CCD). Since camera used in this study has a CCD array attention should be focused to these devices.

CCD sensor is composed of a rectangular grid of electron collection constructs over a silicon wafer produced by growing a layer of silicondiosid then depositing a conductive gate structure on this layer. When an image is formed on the array, photons striking the girt electron hole pairs are generated and electrons are captured by the potential well formed by corresponding gate. After a period of time T electrons generated in each site are collected. Image is read from CCD row by row and a video signal is produced. Frequency of this signal can be standard 30 Hz or user defined.

CCD cameras have various organizations of the sampling arrays. A color CCD camera may have a 2x2 mosaic of red green blue sensitive sensors made by filter coating organized as two green one red one blue being in every mosaic (Bayer pattern) or can use a beam splinter to form image on three different CCDs with different color counting. After this point, image information can be outputted with several ways. RGB output is separate digitization of the individual color channels where a composite video signal format namely NTSC for America PAL and SECAM in Europe and Asia and component video is a format where color and brightness information is separated. In this application composite PAL output at 25 Hz rates were used. After sampling, voltage then transformed into analog video signal. By using a frame grapper this analog image can be transfer into a digital one again.

In real cameras there are several effects degrading the image captured. Blooming is the overflow of the charge stored at a site because of the brightness of the object and can be corrected by illumination control. There also other factor of noise such as thermal and quantum effects, fabrication defects and quantization. Also noise added by camera electronics and frame grabber discretization effects can be included. All these noise sources can be model by using random statistical distributions quite straightforward and can be found at [22].

For a working UAV imaging application, another essential part of the system is the user interface, device interface and communications between them. Images are captured at a camera installed on the plane. Later these images are sent to ground station via wireless communication links as a video signal. This video signal is captured by ground communications unit and sent to a frame grabber to be converted into digital signal and supplied to user interface for processing and display. This user interface also holds UAV user controls. During this process another important factor affects the quality of the video signal and consecutively computer vision algorithms performance. Any kind of signal degeneration due to weather, other source interfaces or communication error creates corrupts a portion or the whole signal creating unwanted defects at the images. These defects can seriously reduce performance of any algorithm.

All these sources has important effects on both general image processing and special for the application studied in this thesis and will be investigated further at later chapters.

2.5 Image Registration Methods Survey

Image registration is one of the oldest fields of Computer Vision and throughout its development, different algorithms and approaches were developed to for more accurate and faster alignment of images. Image registration algorithms generally lies at the core of other

more complex algorithms and in several areas like video compression, image stabilization, constructing video previews and image mosaics [12]. In order to develop an efficient aerial image mosaicing system, variety of image registration algorithms and approaches were investigated.

2.5.1 Two Main Stream Approaches in Image Registration

According to [12] there are two main stream approaches for image registration namely, direct methods and feature based methods:

- Direct methods use pixel to pixel matching and use information at all or a specified portion of the image pixels for motion estimation.
- Feature based methods finds apparent point or regions on the images which are general called as salient point or corner. Then extracts some form of descriptor to describe these points, and uses them for matching two images.

In this study, in order to find the method best suited to proposed application, a large variety of image registration approaches were investigated. This process first started with the Direct Methods and later continued with Feature Based Methods. Preliminary studies are discussed in section 4.3 but first a brief summary of image registration methods are presented in order to provide basic understanding of the concept.

2.5.2 Direct Matching Methods

Direct Methods employs a search algorithm on a part of or whole of image to align two images and look at their pixel difference by using a suitable error metric. In order to speed up the process, pyramidal approach and Fourier transforms can also be used. Also sub-pixel accuracies can be obtained by using incremental methods.

2.5.3 Basic Direct Methods

One of the simplest and straight forward way of aligning images is called Block Matching or Template matching. In Template matching, new image is shifted in translational direction on a template image pixel by pixel and then an error metric is calculated to find the differences between overlapping pixels. For Block matching, same approach is used by using pixel Blocks. These Blocks can be of any size and can be formed by dividing the image into equal blocks or choose by any other approach.

A basic way to calculate the agreement of pixels is to look at the pixel differences by using Sum of Squared Differences (SSD) or Sum of Absolute Differences (SAD). Equations of these approaches are given below;

$$E_{SSD}(u,v) = \sum_i [I_1(x_i + u, y_i + v) - I_0(x_i, y_i)]^2 = \sum_i e_i^2 \qquad (2.6)$$

$$E_{SAD}(u,v) = \sum_i |I_1(x_i + u, y_i + v) - I_0(x_i, y_i)| = \sum_i |e_i| \qquad (2.7)$$

By finding the minimum of the SSD a least squares solution to this problem is obtainable. This approach can be used on different color channels or into intensity images first calculated from color images. In practice a real displacement between two consecutive images does not need to be represented by integer pixel displacements. If this displacement is fractional, an interpolation between the pixels may be required. Also as it is seen from the equations, individual pixel pairs with large difference can alter the final outcome in a mostly matching alignment. In order to prevent this, other error metrics can be applied such as SAD. SAD provides an error metric that grows less quickly so less effected from radical pixel pairs but is not suitable for gradient-descent approaches. Other functions like Median

of Absolute Differences or some function that increases less with rising values can also be used [12].

Equations described above gives every pixel of image same importance. However it is possible to emphasize specific pixels for the application at hand. For example, in a video sequence taken from a moving camera, pixels on the edges of the first image may not be present at the second. Also there can be camera distortions or other effects so that matching with the central pixels may seem more reliable. In such kind of cases a weighting function varying with spatial coordinates of the pixel can be applied to SSD formula as shown below;

$$E_{WSSD}(u,v) = \sum_i w_0(x_i, y_i) w_1(x_i + u, y_i + v)[I_1(x_i + u, y_i + v) - I_0(x_i, y_i)]^2 \quad (2.8)$$

One drawback of using this formula as it is can be seen when matching image pairs with relatively smaller overlap area. In such kind of large displacement conditions, results will be biased toward lesser overlap alignments since not matching pixels does not contribute to total error where even matching right pixel can make small contributions due to round of errors. To overcome this effect, one way is to divide the result with the total overlapping area weights and calculate a mean pixel error. Also a root mean squared intensity error is also possible to be used.

$$A = \sum_i w_0(x_i, y_i) w_1(x_i + u, y_i + v) \quad (2.9)$$

$$RMS = \sqrt{\frac{E_{WSSD}}{A}} \quad (2.10)$$

Intensity difference between pixels may not be only due to the texture differences in the scene. Because of the lighting variances of the area and cameras response to the amount of

light exposed same pixels can have a different illumination level. These effects can also be added into equations by scalar multipliers and constants but it is out of the scope of this study.

Another very common practice for aligning two images is using cross correlation. Cross correlation tends to align same wave shape of two different signal and produces the peak value when to wave shapes are most identical.

$$E_{CC}(u,v) = \sum_i I_0(x_i, y_i) I_1(x_i + u, y_i + v) \tag{2.11}$$

Since cross correlation works based on shape of the signal rather than value of it, it works better in exposure differences [12]. On the other hand if there is a very bright area in the image, maximum product may be in that region so instead of using above formula, normalized cross correlation is more commonly used [12].

$$E_{NCC}(u,v) = \frac{\sum_i [I_0(x_i,y_i) - \overline{I_0}][I_1(x_i+u,y_i+v) - \overline{I_1}]}{\sqrt{\sum_i [I_0(x_i,y_i) - \overline{I_0}]^2 [I_1(x_i+u,y_i+v) - \overline{I_1}]^2}} \tag{2.12}$$

$$\overline{I_0} = \frac{1}{N} \sum_i I_0(x_i, y_i) \tag{2.13}$$

$$\overline{I_1} = \frac{1}{N} \sum_i I_1(x_i + u, y_i + v) \tag{2.14}$$

Note that in this formula if the variance of the region is zero, result becomes undefined. It is also reported that its performance drops for low contrast noisy regions [12].

2.5.4 Search Patterns

Second phase of direct methods is composed of the search methods for comparing respective image matches. Most straightforward and simplest way is to do a full search comparing image patches over a range of displacements. It is sometimes called exhaustive search and its computational cost increases drastically with the search regions size.

In order to speed up this process, an approach is to use hierarchical search algorithms. In this approach, an image pyramid is constructed by down-sampling the image in several layers. Then a full search is applied in the coarser layer to cover a greater distance with less number of shifts. When best match is found, a local search is applied at the next courser level of the image pyramid. This process is repeated until final search is done at the original image. One should not that if image is consisting of high frequency fine detailed features, constructing image pyramids will result in loss of important information and hinder the working of this approach. On the other hand it was reported that in practice, although in not guaranteed to converge to best match, work general fine with much faster speeds [12].

2.5.5 Fourier Based Approaches

One class of fundamental methods that are needed to be investigated for real-time processing speeds in UAV surveillance is the Fourier based approaches. While coarse to fine search using image pyramids can be effective in relatively small search regions, in order to capture large displacements, more pyramids levels needs to be used at the end degrading important features resulting in loss of information. In such conditions another approach alternative is to use of Fourier transforms. Translational motions of a feature in different images can be modeled as two same signal varying with a phase difference in two images. In that case Fourier transform of the shifted signal can be written as follows;

$$F\{I_1(x_i + u, y_i + v)\} = F\{I_1(x, y)\}e^{-2\pi ju.f} = I_1(f)e^{-2\pi ju.f} \quad (2.15)$$

Also convolution in spatial domain corresponds to multiplication in the Fourier domain enable us to write the cross correlation equation as follows;

$$F\{E_{CC}(u, v)\} = F\{\sum_i I_0(x_i, y_i)I_1(x_i + u, y_i + v)\} \quad (2.16)$$

$$F\{E_{CC}(u, v)\} = F\{I_0(u, v) \overline{*} I_1(u, v)\} = I_0(f) \overline{*} I_1(f) \quad (2.17)$$

By using this approach, in order to compute cross correlation of the two images, one can take Fourier transforms of both images, multiplying the first one with the complex conjugate of the second one and take the inverse transform of the result.

One drawback of the Fourier convolution theorem is that it needs to be applied to all pixels in the images using a circular shit for pixels out of image borders. While this can be tolerated for small displacements and similar image sizes, it is not acceptable when overlapping region is a small percentage of the images [12].

In order to cope with this situation cross correlation can be applied to a windowed portion of the images. If a patch from the image is taken and any point outside this patch is considered to be zero, following equation is obtained.

$$E_{WCC}(u, v) = \sum_i w_0(x_i, y_i)I_0(x_i, y_i)w_1(x_i + u, y_i + v)I_1(x_i + u, y_i + v) \quad (2.18)$$

$$E_{WCC}(u, v) = [w_0(x, y)I_0(x, y) \overline{*} w_1(x, y)I_1(x, y)] \quad (2.19)$$

A variant of cross correlation with Fourier transform approach is known as phase correlation. In that case product in every frequency is divided by magnitude of the Fourier transforms. In ideal conditions with no noise and perfect shift inverse transform of the result of following equation results in a single peak located at the phase difference of two images.

$$F\{E_{PC}(u,v)\} = \frac{I_0(f)I_1^*(f)}{\|I_0(f)\|\|I_1(f)\|} \tag{2.20}$$

$$F\{I_1(x+u)\} = I_1(f)e^{-2\pi ju.f} = I_0(f) \tag{2.21}$$

$$F\{E_{PC}(u,v)\} = e^{-2\pi ju.f} \tag{2.22}$$

Phase correlation reported to outperform classical correlation in low frequency noise conditions but decreases performance at low signal to noise conditions. Several other studies are conduct on effecting ways of using correlation and Fourier transform for image registration reported in [12].

2.5.6 Limitations of Direct Methods

Block matching, cross correlation and Fourier based alignments are generally used to estimate translational motion. However in order to capture aerial imagery from surveillance UAV rotation and scaling should be taken into account. There are several methods to use Fourier based algorithms in rotation and scale such as converting image coordinates to log polar coordinates and an overview is given in [12]. For this study such kind of procedure brings additional computational cost so not considered for real time [12].

Techniques described in this section generally used by searching through pixel values. In order to attain sub pixel accuracy inter pixel steps by using interpolation can be used. On the other hand this would bring additional computational cost which is not desirable for real-time processing.

Another well-known approach to perform motion estimation in sub pixel accuracy is known as Lucas Kanade Algorithm. It uses image gradients for calculating SSD function by using a first order Taylor series approximation. Although in this study Lukas Kanade algorithm was not tested, idea of using image gradients is important for later feature matching methods.

2.5.7 Feature Matching Methods

Second main approach in image registration can be referred as feature matching methods [12]. Performance of direct methods depends of the distributions of the image intensities, namely having regions distinctive and apparent gradient distributions. Unlike direct methods, feature matching methods does not try to match each pixel in two images. There are based on finding these apparent points or regions in an image called features, and matching these features to estimate a geometric transformation.

2.5.8 Harris Corner Detector

Accuracy of the image registration is dependent on the presence of strong gradients hence eigenvalues of the Hessian matrix can be considered as most critical in determining performance of the image matching. In [23] Shi and Tomasi proposed a method that is based on calculating these eigenvalues and determining these points and used patches around these points to track them in an image sequence.

While Shi Tomasi uses a square patch with equal weight, Harris and Stephens [17] proposed using Gaussian filters as presmoothing methods. This enables the evaluation of Hessian and Eigenvalue images by using a series of filters and algebraic operations. Equations of this approach are presented below;

$$C_x(x,y) = \frac{\partial}{\partial x} G_{\sigma_d}(x,y) * I(x,y) \tag{2.23}$$

$$C_y(x,y) = \frac{\partial}{\partial y} G_{\sigma_d}(x,y) * I(x,y) \tag{2.24}$$

$$B(x,y) = \begin{bmatrix} C_x^2(x,y) & C_x(x,y)C_y(x,y) \\ C_x(x,y)G_y(x,y) & C_y^2(x,y) \end{bmatrix} \tag{2.25}$$

$$A(x,y) = G_{\sigma_i}(x,y) * I(x,y) \tag{2.26}$$

$$\lambda_{0,1}(x,y) = \frac{a_{00}+a_{11} \mp \sqrt{(a_{00}-a_{11})^2+a_{01}a_{10}}}{2} \tag{2.27}$$

In these equations $G_x(x,y)$ represents the derivative filter where $G_{\sigma_i}(x)$ represents the integration filter. Förnster uses these equations to calculate the minimum eigenvalue which is deterministic in finding good feature points [12].

Since the actual calculation of the eigenvalue is computationally expensive, Harris proposes another approach. Instead of calculating the actual eigenvalues, Harris uses the following formula with $k_H = 0.06$ as a score to determine the quality of the potential key points.

$$\det(A) - \alpha \mathrm{tr}(A)^2 = \lambda_0\lambda_1 - k_H(\lambda_0 + \lambda_1)^2 \tag{2.28}$$

Other authors also studied usage of different metrics but Harris detector proven to be superior in various comparison studies [24]. One of the metrics that they use is called repeatability. Repeatability is ability to find same point in two consecutive images. This way comparing feature points in two images became meaningful. Another metric that they use is called information content which is amount of apparent intensity variance around a key point. As the result of their comparison, they found that improved Harris corner detector outperform other according to their metric scales [24].

2.5.9 Scale and Rotation invariance

Rotations due to unstructured camera and UAV movements and scaling due to zooming present critical factors affecting UAV surveillance. Since such effect is present in a wide range of practical applications, another major field for feature detection and description is the scale and rotation invariant methods. While classical methods work well on translation motion, a scale and rotation invariant detector would be better at capturing image motions from a free moving camera. One way to achieve scale invariance is for scale-space maxima of Difference of Gaussian (DoG) [19] or Harris corner [17] detectors computed at an image pyramid where the sub-sampling between adjacent levels is less than a factor of two also referred as a sub-octave pyramid. For computation of the pyramid, a factor of $\sqrt{2}$ can be used [19]. Other kinds of scale, and rotation invariant feature detectors and descriptors were proposed and a good survey can be found at [25]. A complete survey of every detector is beyond the scope of this study but SIFT and SURF detectors are discussed at 2.10 and 2.12.

In order to align images with different orientation, rotation invariant feature detectors also were investigated. If pixel sampling of an image is considered, when a feature fitting

in a square in initial position is rotated, it will fit in a totally different shape. If square patches are compared or using square patches for descriptor calculation, a different result from the area of the same feature in the initial image will be obtained. If this rotation degree is not too much, corresponding error will not be too great so making a correct match would be still possible but if this rotation is large, such as 45 degree, comparison would be done on to different patch intensity distributions even if they are located at the same keypoint. In order to compensate for that one can think of calculating several descriptors by incrementally varying rotational degree but this will be a cumbersome method to achieve desired accuracy. Another simple way to apply is to calculate a dominant orientation by looking at the gradients of the patch and calculating the descriptor. Brown [12] uses average gradient orientation direction and Lowe [19] uses histogram of local orientation of the gradients for this purpose.

Apart from keypoints there are other kind of feature used for image registration such as; line segments, affine invariant regions, maximum stable regions calculated using watershed detection, salient regions calculated using patch entropy [12] but these methods were not investigated in this study because of the proven performance and wide usage of keypoint based methods.

2.5.10 Scale Invariant Feature Transform: SIFT

Scale Invariant Feature Transform (SIFT) aims to detect distinctive scale invariant feature by using Difference of Gaussians (DOG) function. Author state that under variety of assumptions only possible scale- space kernel can be the Gaussian function [19]. Then defines space-scale of an image as a function $L(x, y, \sigma)$ as;

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{2.29}$$

which is result of convolution of an image $I(x, y)$ with the Gaussian function

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \tag{2.30}$$

Author then defines Difference of Gaussian function $D(x, y, \sigma)$ as;

$$D(x, y, \sigma) = \big(G(x, y, k\sigma) - G(x, y, \sigma)\big) * I(x, y) \tag{2.31}$$

which is equivalent to difference between two nearby scales separated with a constant scale multiplier$k$.

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \tag{2.32}$$

Later on this difference is used to determine space extreme and used to detect stable keypoints.

Author refers to earlier studies to site that maxima and minima of Laplacian of Gaussian $\sigma\nabla^2 G$ provides the most stable image features and Difference of Gaussian provides a good approximation to Laplacian of Gaussian [19].

$$\sigma\nabla^2 G = \frac{\partial G}{\partial \sigma} \approx \frac{G(x,y,k\sigma) - G(x,y,\sigma)}{k\sigma - \sigma} \tag{2.33}$$

This equation then can be reformed as;

$$G(x, y, k\sigma) - G(x, y, \sigma) = (k-1)\sigma^2\nabla^2 G \tag{2.34}$$

which show that Difference of Gaussian function having scales differing by a constant factor incorporates normalization by σ^2 required for scale invariance. Later on Difference of Gaussian function is used to find local extreme at a range of scale levels to find scale invariant apparent keypoint locations [19].

In order to achieve rotational invariance, a histogram of gradients around the locations of keypoints is constructed and a principle orientation is assigned. Later on coordinates of the descriptor and gradient orientations are rotation according to keypoint orientation for descriptor representation. Resultant descriptor can be considered as an image feature defined with smaller features around its location which provides enhanced distinctiveness [19].

2.5.11 Methods Designed for Speed

While accuracy is important for many feature matching scenarios, real-time applications also have strict speed requirements. Since for real-time mini UAV surveillance computation speed is also very crucial, various optimized detection description methods were also investigated.

2.5.12 Speeded-Up Robust Features: SURF

Speeded-Up Robust Features (SURF) is based on similar approaches with SIFT algorithm where authors primary focus was increasing speed of computation through several optimizations. This is achieved by approximating Gaussian derivatives with box

filters and using integral images for fast computation of box filter response through space-scale range [20].

Integral images are formed by summation of all pixel values through a rectangular region formed by origin and a location as;

$$I_{\Sigma}(x,y) = \sum_{i=0}^{i\leq x}\sum_{j=0}^{j\leq y} I(i,j) \tag{2.35}$$

and enable fast computation of box type convolution filters. After an integral image is formed, sum of intensities over an upright rectangular area of can be calculated by only using two subtractions and one addition, independent of size [20].

Hessian matrix at a point (x,y) of a point is as show below;

$$H(x,y,\sigma) = \begin{bmatrix} L_{xx}(x,y,\sigma) & L_{xy}(x,y,\sigma) \\ L_{xy}(x,y,\sigma) & L_{yy}(x,y,\sigma) \end{bmatrix} \tag{2.36}$$

where $L_{xx}(x,y,\sigma)$, $L_{xy}(x,y,\sigma)$ and $L_{yy}(x,y,\sigma)$ are the second order Gaussian derivatives with respective directions. Since SIFT is an successful approximation to Laplacian of Gaussians authors make a further approximation by using box filters instead of discretized Gaussian kernels. Integrating this approximation with integral images for calculation of box filter responses, authors were able to develop a detector algorithm with a relatively low computational cost [20].

SURF descriptor also utilizes an approach similar to SIFT. For the orientation assignment Haar wavelet responses of sampled points around the keypoint calculated by using box filters. Later on wavelet responses and absolute values of wavelet responses at vertical and horizontal directions summed in order to form a 4D descriptor for each sub-

region. Descriptors of each sub-region then combined to for 64D surf feature descriptor. An "Upright" version exists for increased speed and distinctiveness [20].

2.5.13 Fast Area Segmentation Test: FAST

FAST detector was developed by keeping speed in mind and archives higher computational speed by utilizing a simple segmentation test [18]. Every pixel in the image was compared 16 surrounding pixel that forms a discretized circle and flagged as darker, brighter or similar as it is shown in equation below;

$$S_{p\to x} = \begin{cases} d, & & I_{p\to x} \leq I_p - t & (darker) \\ s, & I_p - t < I_{p\to x} < I_p + t & (similar) \\ b, & I_p + t \leq I_{p\to x} & (brighter) \end{cases} \tag{2.37}$$

In order to classify a pixel as a corner point, a predetermined number of surrounding pixel should be darker or brighter than the central pixel through a continuous arc. In the original paper, several numbers were tested and 9 and 12 found to be optimum for performance [18].

To further increase the performance of the detector, authors employed machine learning and developed a comparison routine that compares surrounding pixels at an order to detect failure without going over all circumferential pixels. Also non-maximal suppression which is selecting the most apparent corner point among neighboring corner points was not possible in the original version because of lack of matching scores. A modified version which calculates the sum of absolute differences between the central pixel and the pixel in contiguous arc was developed by the authors in order to provide a score for apparent pixel selection [18].

2.5.14 Binary Robust Independent Elementary Features: BRIEF

Binary Robust Independent Elementary Features (BRIEF) utilizes a test to form binary descriptors that has fast matching speed by calculating hamming distances. In their paper authors state that binarization of the various descriptors employed in several studies since hamming distances between binary vectors can be calculated very fast on modern CPUs. On the other hand these techniques require first computation of the original descriptor making them inefficient to use. Authors overcome this situation by directly computing binary descriptors by using smoothed image patches and pair-wise intensity comparisons [21].

Test τ is defined as shown at equation 2.38.

$$\tau(p; x, y) = \begin{cases} 1 \; if \; p(x) < p(y) \\ 0 \qquad otherwise \end{cases} \tag{2.38}$$

Where $p(x)$ is the pixel intensity at smoothed patch p at location vector x. A set of (x, y)location pairs defines a test which produces a BRIEF descriptor bit string as show at equation 2.39.

$$f_{nd}(p) = \sum_{1 \le i \le n_d}^{n} 2^{i-1} \tau(p; x_i, y_i) \tag{2.39}$$

Where n_d is the number of bits in the descriptor and selected to be 128, 256 or 512 by the authors.

5.5.15 Feature Tracking and Feature Matching

After finding apparent features in an image there is several ways to utilize this information in order to perform image registration. One of these approaches is called Feature Tracking. In this method, first keypoints in an image is detected. Later patches around these keypoints are tracked by using direct matching methods through a sequence of images. As it is seen in the direct methods section, performance of these methods largely depend on quality of the image gradients in the matching region. So this approach can be thought of as an enchantment to direct methods by finding points that they will perform well such as Good Features to Track [23].

Because of employment of direct methods in Feature Tracking algorithms are subjected to similar constraints in many ways. If an image sequence such as video frame is considered and assume that displacement between consecutive frames is small, feature tracking can perform well. As displacement increases, required search region also increases. Also in an image sequence initial features detected can be lost in later frames because going out of the scene. In such cases, if number of tracked features decreases to a certain level, performing feature detection in order to update feature points is required. Also if initial features are tracked for a long image sequences, shapes of these feature can change because of the changes in the viewing angle. In that case calculating a new descriptor from the new image would be helpful.

While feature tracking is employed in many applications, its advantages and disadvantages for using in aerial imaging by mini UAV will be discussed later. Another approach after feature detection for motions estimation is called Feature Matching. This approach is more suitable for unknown image motions and larger displacements than feature tracking. In this method, first features in two images are detected. Then they are compared to each other by using a matching algorithm. After calculating point

correspondences, a suitable method is used to estimate a geometric transformation. This process is described in more detail in following sections.

Most straight forward way to finding point correspondences is to compare every feature in the reference frame keypoint set to every feature in the target frame keypoint set one by one ad that relating every one keypoint tin the first set to one keypoint in the second based on their score metric. This method enables to cover all the possible correspondences but its computational cost increases quadraticly with the number of feature included. This means that it will perform relatively faster when small feature sets are used its computation speed.

Several other algorithms were proposed in order to make faster keypoint matching possible. Some of these methods can be summarized as follows. Beis and Lowe uses a modified search ordering for a k-d tree algorithm called Best Bin First Match. Shakhnarovich uses an extend version of the locality sensitive hashing called parameter-sensitive hashing utilizing unions of independently computed hashing functions. Brown hash the first three Haar wavelets from an 8 8 image patch. Detailed descriptions of these matching methods is beyond the scope of this study and a matching strategy utilizing exhaustive matching were developed because of the reported instability of the other methods [12].

Chapter 3

AERIAL MOSAICS

3.1 Geometric Transformations

In order to efficiently capture unstructured image motions due to camera and UAV movements, an understanding of the geometric transformation and their limitations is required. In this section, a brief overview of the geometric transformations that were investigated in this study is presented.

3.1.1 Euclidean Transform

Isometric transforms preserves Euclidean distance and can be written as equation below.

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} \varepsilon\cos\theta & -\sin\theta & t_x \\ \varepsilon\sin\theta & \cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{3.1}$$

In this equation $\varepsilon = \pm 1$ and if $\varepsilon = 1$ it is orientation-preserving and called Euclidean Transformation [26]. Euclidean transformations are most basic and specialized of projective transformations. This transformation models rigid body motion of an object in 2-D plane. It can be written as follows;

$$x' = H_E x = \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} x \tag{3.2}$$

where R represents the 2x2 rotation matrix and t represents the translation vector in x and y directions. It has 3 degrees of freedom.

3.1.2 Similarity Transform

Similarity transformation is an isometry subjected to isotropic scaling. In other words, while having all degrees of freedom of Euclidean transforms it has an additional scaling degree of freedom [26]. Transformation matrix can be written as follows;

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} s\cos\theta & -s\sin\theta & t_x \\ s\sin\theta & s\cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{3.3}$$

In this equation s represents the scaling factor. Similarity transform matrix can also be written as shown below;

$$x' = H_S x = \begin{bmatrix} sR & t \\ 0^T & 1 \end{bmatrix} x \tag{3.4}$$

It has 4 degrees of freedom and preserves “shape” of the transformed object. It is composed of rotation and translation in two dimensions and scaling.

3.1.3 Affine Transform

Affine transforms are composed of a non-singular linear transformation followed by a translation [26]. Affine transform matrix can be written as shown below;

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{3.5}$$

It can be written as shown below;

$$x' = H_A x = \begin{bmatrix} A & t \\ 0^T & 1 \end{bmatrix} x \tag{3.6}$$

where A represents the 2x2 non-singular matrix. It has 6 degrees of freedom. By using Singular Value Decomposition A can be written as follows;

$$A = R(\theta)R(-\phi)DR(\phi) \tag{3.7}$$

In this equation θ and ϕ represents rotation angles and D represents the diagonal matrix found by SVD. D can be written as follows;

$$D = \begin{bmatrix} \lambda_0 & 0 \\ 0 & \lambda_1 \end{bmatrix} \tag{3.8}$$

where λ_0 and λ_1 are factors of non-isotropic scaling.

Operation performed by matrix A can be summarize as a rotation by ϕ degree, a scaling by λ_1 and λ_2 at rotated coordinate system, a rotation back by $-\phi$ degree and a final rotation

of θ degree. Affine transforms preserve parallel lines, ratios of length of parallel lines and ratio of areas.

3.1.4 Perspective Transform

A projective transformation also knows as perspective transformation is a general non-singular linear transformation in homogenous coordinates [26]. It is matrix representation is as follows;

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{3.9}$$

and can be written as below;

$$x' = H_P x = \begin{bmatrix} A & t \\ v^T & v \end{bmatrix} x \tag{3.10}$$

In this equation v represents the perspective term. It can be computed from from 4 point correspondences. It has 8 degrees of freedom. If the term v is 0 it is not possible to scale the matrix where v is unity. Ratio of ration or cross ration of lengths of lie is invariant.

In projective transformations area deformation depends of the location of the area unlike affine transformations. Because of the perspective term they can map infinite points into a coordinate.

A projective transformation can be decomposed as follows.

$$H_p = H_S H_A H_{Per} = \begin{bmatrix} sR & t \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} K & 0 \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} I & 0 \\ v^T & v \end{bmatrix} = \begin{bmatrix} A & t \\ v^T & v \end{bmatrix} \tag{3.11}$$

$$H_p = H_{Per} H_A H_S = \begin{bmatrix} I & 0 \\ v^T & 1 \end{bmatrix} \begin{bmatrix} K & 0 \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} sR & t \\ 0^T & 1 \end{bmatrix} \quad (3.12)$$

3.2 Effect of Different Transformations in Aerial Image Mosaics

When combining aerial images it is crucial to choose a suitable transform to create mosaics effectively. Factors that affect images to be combined are movements of the UAV, direction and movements of the camera, structure of the terrain to be mapped. Since real-time and near real-time processing speeds are required for this study, computational burden is also a points of concern as well as capturing unstructured image motions.

At the most general case, perspective transform is best to represent image movements, and effects. It can represent the perspective affects due to movement of the UAV and camera so theoretically give more accurate mapping in unstructured movements. Also it can model zooming and changes of the imaging height.

On the other hand because perspective transform is linear in homogenous coordinates, calculations needed to be done by using this coordinate system. This creates additional computational burden when result coordinates are transferred back into Cartesian domain in order to map one image onto another. Also because of the flexibility of the perspective transform, any errors in transformation matrix create more apparent distortions in resultant mosaic. This affect was clearly observed at experiments in this study and was also reported by several authors. Also if the case of imaging from a camera fixed in a single point is considered, rotating in up down or sideways directions, a perspective transform will be far better to capture this movement. At same conditions a Euclidean transform will create error in every image registration. Yet especially if the image scene is getting closer to horizon, a perspective transform distort images increasingly in order to capture the movement that will result in newly acquired images being distorted and skewed drastically making it impractical to use. So in such kind of extreme situations, advantages of perspective

transform can turn out to be a disadvantage and such kind of conditions frequently arise in aerial imagery with unstructured camera and UAV motions.

An alternative is Euclidean transform with just 3 degrees of freedom, namely one rotational and 2 translational motions. If UAV is flying parallel to a 2d planar terrain, with a downward looking camera, general projective transformation representing the full image motion will reduce to an Euclidean transform. Any further deviation because of camera angles or UAV movements will produce slight errors. Also since Euclidean transformation can provide mapping with Cartesian coordinates, it does not have additional computational burden that perspective transformation brings. When camera rotates from side to side or up down at a fixed point, Euclidean transformation will generate registration error and will not be able to cover the whole motion but also will not distorted the newly acquired frames. It will also not be able to cover any camera zoom or UAV elevation changes.

Affine transform having 6 degrees of freedom can provide mapping in Cartesian coordinates with same computational cost with Euclidean transform. 6 degrees of freedom may seem like it would be better in covering camera motions but in an actual imaging conditions, images will go under perspective transformation and skewing of affine transformation will not be enough. Yet having more degrees of freedom, it will be subjected to greater image distortions and in a mosaicing application an estimation error in transformation matrix can create such distortions that propagates throughout the rest of the frame sequence. On the other hand it can model image zooming or changes in the imaging height unlike Euclidean transform.

Another alternative for image mosaicing is similarity transform. Similarity transform is like Euclidean transform, having an additional variable for scaling effects. This variable enables them to model image zooming. Mapping with a similarity transform have the same cost like affine and Euclidean. It has 4 degrees of freedom for two translational one rotational and one scaling degrees so it does not have the distorting skewing affect that can

be found in affine or perspective transform estimation error. On the other hand en error in scaling propagates throughout the rest of the frame sequences mapping every new frame to the mosaic in a smaller scale.

In this study, several transformation approaches were tested in order to find the best suited ones for aerial mosaicing using small UAVs. Because of the nature of the vehicle, transformation should be flexible enough to capture unstructured image motions due to fast turning and moving UAV yet should not distorted the images too much. It should be able to work in real time and it should be able to allow users to explore terrain at different camera zooms and heights. As a result Similarity transform was selected because of its optimum properties for capturing unstructured motion without skewing effects and relatively fast computation time.

3.3 Geometric Transform Estimation

After establishing feature correspondences between reference and target set by using a matching method of chose, a transformation matrix is estimated to relate two images. In practice in a correspondence set is composed of a number of false matches next as well as correct matches. If simply average values of displacement were calculated by using this set, incorrect matches can result in deviations from actual values. So in order to perform a successful image registration, methods that work with such set compositions are utilized. Two of the most commonly used methods for this purpose are called LMS and RANSAC [26].

3.3.1 LMS

Least Median of Squares method starts with selecting a number of keypoint matches. A geometric transformation is calculated using these correspondences. Then errors between the target set and transformed initial set keypoints is calculated and this process is repeated until median of this error is minimum. Since a mismatch having a large deviation from the correct result can greatly affect the outcome of a set composed of mostly correct matches this method was avoided and a more suitable method called RANSAC

3.3.2 RANSAC

Another important method is called Random Sample Consensus (RANSAC) [27]. Like in LMS method, again a set of keypoint correspondences selected for initial calculation the geometric transform. Then keypoint in the reference image is transformed to find their new locations in the target image. Residuals between these transformed keypoints and matched keypoints in the target image is calculated as shown in the below equation.

$$r_i = \tilde{x}_i'(x_i; p) - \hat{x}'_i \tag{3.13}$$

By setting an error bound usually at a few pixel, residuals which are smaller than this error bound is counted and those matches are selected as inliers. This process is repeated a number of times and the transformation with the largest set of inliers is selected or repeated until a desired percentage of keypoint matches is counted as inlier.

Since RANSAC utilizes a random selection process, its convergence to a successful estimation should be calculated in statistical manner. Sufficient number of trials S can be

calculated by using the below equations where P is the probability of success kr is the random samples and p^k is the probability that all random samples are inliers in a selection.

$$1 - P = \left(1 - p^{kr}\right)^S \tag{3.14}$$

$$S = \frac{\log(1-P)}{\log\left(1-p^{kr}\right)} \tag{3.15}$$

It can be easily seen that number of trials required depends on both the probability margin that is selected and percentage of inliers to outliers in a given matching set.

3.4 Image Blending

Image blending is the process of combining images after registration part is completed. When combining images several defects can occur, such as visible seams (due to exposure differences), blurring (due to misregistration) and ghosting (due to moving objects). General approach in academia in this area of research is to finding efficient methods and algorithms to overcome such kind of defects. Some of the most common methods for seam selection and blending can be listed as below;

• Averaging: Taking an average of the new and old pixel values. Simple averaging may not work well under registration errors and scene movements and presence of moving objects.

• Median: Median filtering used to remove rapidly moving objects.

• Feathered average: Averaging with weighting changing due to a distance map, giving more emphasis over the pixel near the center and at edges less. Reasonably effective over exposure differences but blurring and ghosting is still present.

• P norm: A form on feathering that employs rising of distance map values to some large power. It is considered as a reasonable trade of between exposure differences and blur.

• Vornoi: A version of p norm blending where $p \rightarrow \infty$. Result is assignment of each pixel to the nearest frame center in the image sequence. It is reported to have very hard edges with noticeable seams.

• Weighted ROD vertex cover with feathering: An algorithm that takes underlying image structure into account. Algorithm first takes into account all the image sequence in order to determine Regions of Differences (RODs). Then areas of disagreements are determined mostly created by moving objects. These regions are removed in the final image resulting in a composite image where moving objects are only included from one frame at a time. Algorithm has a tendency to remove objects at the edges of the frame by including where they are at the center.

• Graph cut seams with Poisson blending: Is an algorithm where user roughly selects region to be included than regions to be included by using statistical calculations.

• Pyramid blending: This method utilizes Laplacian image pyramids instead of single level blending.

A more detailed discussion and description of blending methods can be found in reference [12]

While these methods were developed in order to create better looking resultant images, aim in this study was to provide user with most useful knowledge of the view in real time. Since even the simplest blending methods includes operations than in all of the image pixels which bring additional computational burden, a simpler direct approach of writing newly acquired pixels on the old pixels at resultant image mosaic was selected. With this method it is possible to do blending without much additional computational cost and

provide user with the latest real time view of the scene which is crucial for surveillance applications.

3.5 Mosaicing Overview

In the study presented here, mosaicing approach was chosen in order to full fill requirements of surveillance with unstructured UAV and camera motion. In classical mosaicing mode, images are integrated in a local mosaic in order to avoid excessive error accumulation. No bundle adjustment is done since processing needed to be done at real time speeds and presented to user. Image warping algorithm is optimized in a way that warping speed became independent from the size of the resultant mosaic. This enabled construction of local mosaics with number of frames and sizes that are usually seen at global mosaics but since not bundle adjustment or any kind of error update is used, accumulation of the error causes significant error as the number of frames integrated increases. In order to provide critical most update view to the user and avoid ghosting affects due to blending, most recent frame is transformed and written on to the existing pixels on mosaic. Instead of preserving full frames in memory, only mosaic image and feature descriptors are preserved. If newly acquired image is transformed outside the boundaries of the mosaic, mosaic image is shifted in order to fully enclose the most recent frame and pixels shifted out of boundaries are erased. All mosaic sequences is written into a video file making system a dynamic mosaicing application with all of the temporal and recent scene information is preserved for later view. If a frame to frame registration is failed, unmatched frame is transformed into the location of the last successfully matched frame but not integrated into mosaic providing user with the most recent surveillance information even in the case of registration failures. Later on construction of the mosaic continues with the next successfully matched frame. Classical mosaicing mode takes the first frames coordinates as reference coordinates and providing mosaic images with the

alignment of the start of surveillance which is beneficial for some cases. Since movements of the UAV can cause disorientation of the user, a second mode is constructed dividing rotation component of the most recent frame transformation among mosaic image and the transformed frame according to a smoothing function. This enables a smooth following of the camera alignment prevent user from losing the orientation of the UAV. Details of these mods and mosaicing component are presented at the system description part in detail.

3.6 Measuring Mosaic Quality

Although there is a developed literature on image mosaics for a wide variety of applications, most of the evaluation methods based of subjective examination and method specific metrics. Main problem with objective evaluation of the mosaic quality is obtaining accurate "Ground Truth" data [13]. If the mosaic outputs of the proposed algorithms are compared to a reference mosaic, accuracy of the comparison is bounded with the accuracy of the reference method. If marking of reference scenery is used, accuracy is affected by the accuracy of other measurement methods. Several authors employed artificially created realistic camera frames by using a base image in order to obtain accurate ground truth information [13,14]. A "Virtual Camera" is developed in order to construct video frames with different viewing positions by using reference image and later on reference image is used as ground truth information [13,14,15].

In order to quantify comparison results, several metrics are proposed by various authors [13,14,15]. Some of these metrics are Average intensity differences, average geometric difference between control points, number of misplaced pixels [13], base image coverage of mosaic coupled with Sum of Squared Differences [14], entropy, clarity, registration error, peak signal-to-noise ratio and structural similarity [15]. Details of these methods and techniques can be found in respective papers [13,14,15].

Since the main scope of this study is developing and real-time real-world image stabilization system for mini UAV flights, evaluation of the methods were not considered independent from the process. Although artificial frames based of real images were used in early simulations, this approach was intentionally avoided for later comparison tests. Instead approaches that enable testing during the operation of the UAV and setups simulating real world conditions including camera and optical effects were developed.

First approach for algorithm evaluation was the use of inlier and outlier information as a metric for image registration accuracy. In this approach number of outliers n_{out} was divided by total number of matches n_{total} in order to determine outlier percentage e_{out} (3.16). This was an indirect and algorithm specific method to measure image registration quality and since only difference between the compared methods was in their detector and descriptor component, it was a viable method for the system described in this study. It was based on the assumption that quality of the resultant mosaic and more generally performance of the all methods including stabilization was primarily based on the quality of the image registration part. Advantage of this approach was that it enabled online accuracy measurements without using any test set up or predefined test sets and can easily be used in actually flight tests without bring in any additional computation or restrictions.

$$e_{out} = 100\frac{n_{out}}{n_{total}} \tag{3.16}$$

While the first approach provides an indirect measurement, a more direct measurement of the mosaic quality was achieved by calculating average pixel differences e_{avg} at the overlapping area of the newest frame and mosaic before blending by using (3.17). Since every new frame was processed and blended with the mosaic image in order to provide user with most recent information, a portion of the scenery present in the newest frame $f(x, y)$ is generally also present in the mosaic $m(x, y)$. Apart from noise, pixelization effects, and

illumination changes, pixel values present in mosaic and newest frame should represent same underlying intensity distribution. In this approach difference between these pixel values were considered as an indicator of registration error. It should be noted that this error metric also represents noise effects, any illumination and perspective changes and pixelization errors. On the other hand since these effects are not altered by changing registration methods, average pixel difference was considered as a good indicator of relative algorithm accuracy with a fixed offset present in all tests. Also presence of these effects was considered to provide a more realistic test environment of the developed methods. One important aspect that needs to be taken account was the effect of drifting. While minor registration errors results in slight shifts in the same image gradient providing meaningful difference values, accumulated error may result in comparison of unrelated pixel which turns out to be misleading.

$$e_{avg} = \frac{1}{N}\sum_{(x,y)}\big(m(x,y) - f(x,y)\big)^2 \tag{3.17}$$

Third approach is developed in order to measure drifting due to registration errors. In ideal registration when camera frames follow a path and come back to the starting point of the first frame, total transformation matrix should be equal to identity matrix. Any difference between the identity matrix and the final total transformation matrix is considered due to drifting registration errors (3.19).

$$E_{drift} = F_{final} - I \tag{3.18}$$

$$E_{drift} = \begin{bmatrix} s\cos\alpha - 1 & -s\sin\alpha & d_x \\ s\sin\alpha & s\cos\alpha - 1 & d_y \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} s\cos\alpha & -s\sin\alpha & d_x \\ s\sin\alpha & s\cos\alpha & d_y \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.19}$$

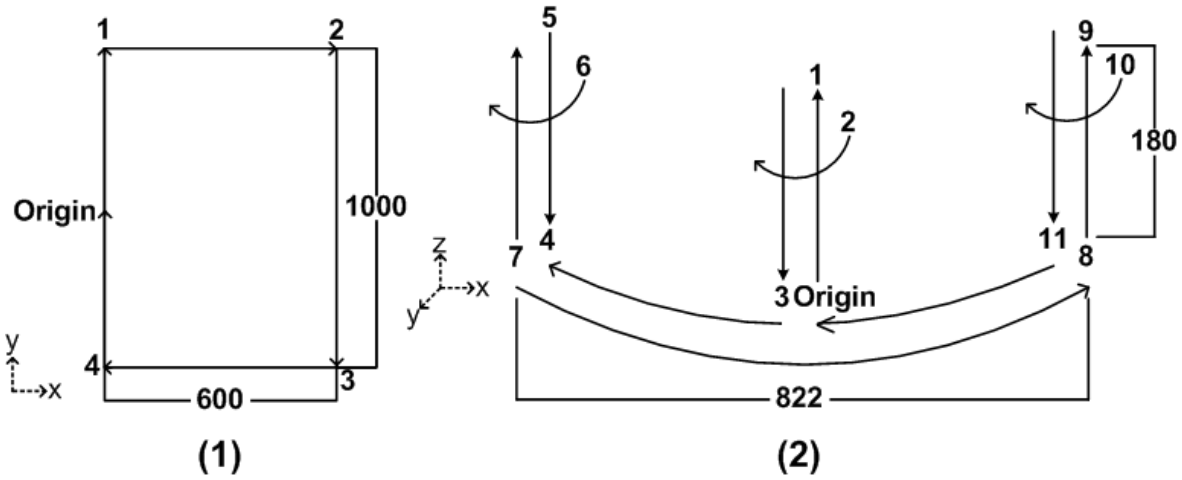


Figure 3.1: Schematics of translation path (1) and rotation – scale path (2). Dimensions given are in millimeters. Starting from origin, CNC head navigates through all points to complete a cycle and going back to starting point. Upward arrows in (2) showing upward movements where head is lifted 180 mm, rotated 120 degrees and moved downwards to original height which produces scaling and rotation effects.

$$C_{orgin} = \begin{bmatrix} d_{xorgin} \\ d_{yorgin} \\ 1 \end{bmatrix} \tag{3.20}$$

$$C_{final} = \begin{bmatrix} d_{xfinal} \\ d_{yfinal} \\ 1 \end{bmatrix} \tag{3.21}$$

Several camera paths shown in Figure 3.1 having pure translations and combinations of translation and rotations were tested by using a 5 axis CNC and printed high resolution aerial images in order to simulate UAV motion to provide truth data. Camera followed these paths at a fixed amount of time so effects of missing frames due to computation were also included. Difference between the original central point C_{orgin} (3.20) and final central

point C_{final} (3.21) were calculated by using (3.22) and (3.23) in order to provide a drifting value e_{drift} in terms of number of pixels. Details and results of this method can be found at Section 14.1.

$$C_{final} = F_{final} * C_{orgin} \tag{3.22}$$

$$e_{drift} = \sqrt{\left(d_{xfinal} - d_{xorgin}\right)^2 + \left(d_{yfinal} - d_{yorgin}\right)^2} \tag{3.23}$$

3.7 Annotations

Another way of enhancing use of mosaics for aerial surveillance is utilizing annotations. Although annotations are more suited for using with global mosaics [1,3], a local mosaic version of annotation were implemented at classical mosaic and rotating mosaic modes.

When user double clicks on a point in the screen, screen coordinates are transformed into mosaic coordinates and a total shift transform that represents the shifting of the mosaic image from the beginning of the process for annotation is produced. Every shift that is applied to the mosaic image after placement of the annotation is also applied to this transforms. If an additional point is clicked, same process is applied to the new point. This way several points with different total shift transformation can be hold. If annotation point ventures out of the mosaic boundaries it is still held in the memory so when viewer screen turns back to annotation point location, it can be seen. On the other hand because of the accumulation of the errors, drifts from actual point location were also observed for large displacements.

Chapter 4

SYSTEM DESCRIPTION

In this chapter working principles of the system developed for real-time mosaicing and stabilization of UAV images is described. Theoretical background of the concepts discussed in this chapter is given at Chapter 2 and Chapter 3. Results of the flight tests and in-door experiments are presented at Chapter 5.

4.1 System Overview

In order to overcome size and weight constraints of the mini UAVs, system presented here is designed to be as much hardware independent as possible. For this purpose, basic UAV system configuration consisting of one UAV and one Ground Control PC was considered as default setup. Any additional hardware requirement for computation such as a GPU or on-board video processor card was intentionally avoided. All processing was done on software and real-time processing speeds and operational level accuracy are aimed to be achieved by optimizations done on algorithm side. To further increase the flexibility, processing software was designed to be single threaded so that tested performance was also independent from CPU architecture. It should be noted that developed algorithms can easily be expended to multithreaded applications for a speed gain in more specialized PC configurations. Schematic of the system overview is shown in Figure 4.1 where an additional processing station is optional.

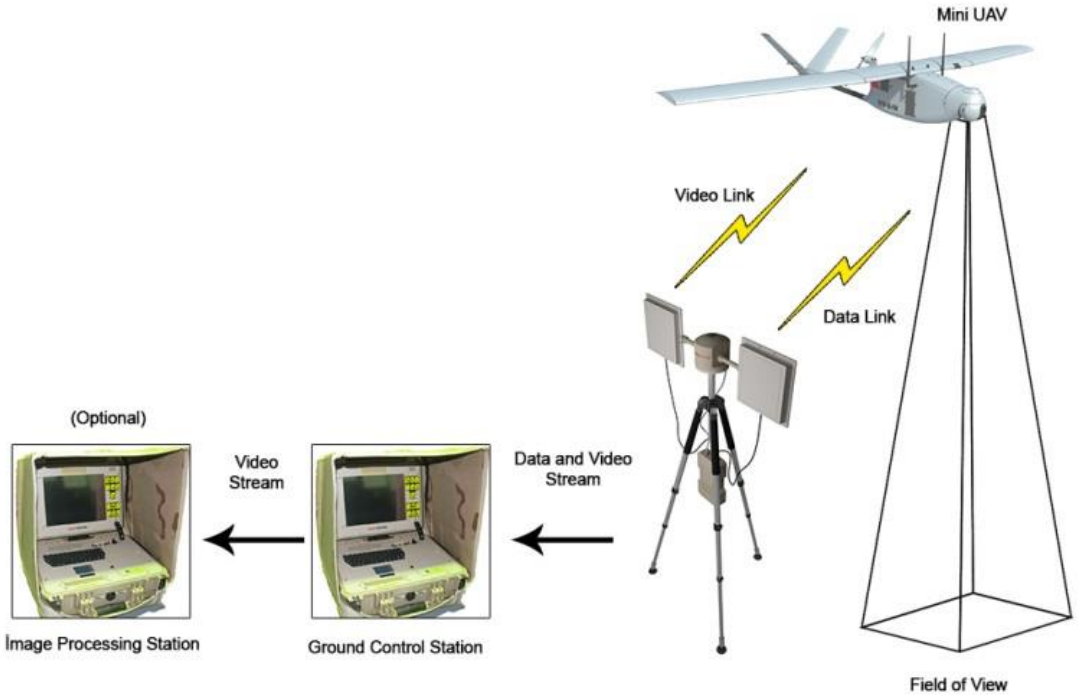


Figure 4.1: Schematic of the system showing connection between Bayraktar mini UAV, tracking antenna and Ground Control Station. It should be noted that second station is optional and does not change processing performance.

Mini UAV used in this study is Bayraktar mini IHA which is widely used in various surveillance missions worldwide such as border patrol at Turkey and Qatar (Figure 4.2). It has a wingspan of 2 meters, length of 1.2 meters, and weights 4.8 kg. Its standard operational speed is 60 km/hour at an altitude of 1000 meters and has a maximum altitude of 4000 meters. Its range is 15 km and can carry day light and infrared cameras as primary payload. UAV is controlled from a mobile Ground Control Station via a tracking antenna.

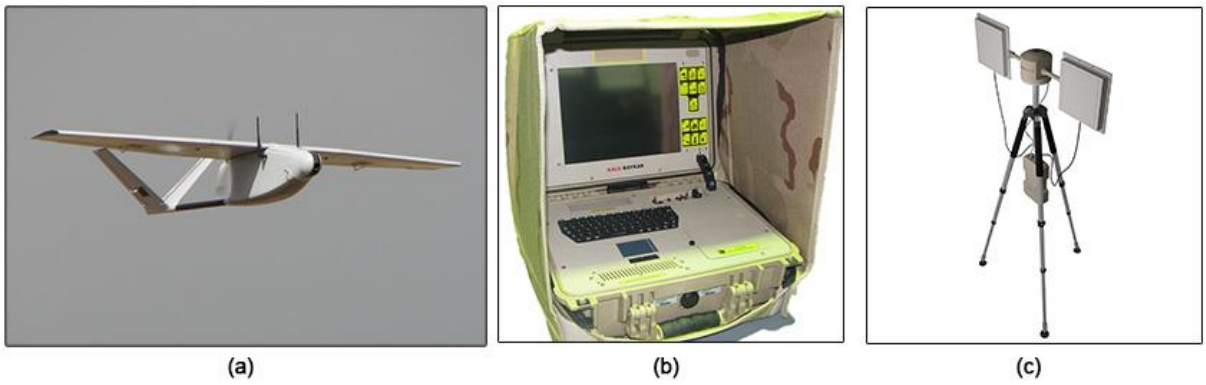


Figure 4.2: Test platforms: (a) Bayraktar mini UAV, (b) Ground Control Station, (c) tracking antenna.

Ground Control Station weights 11 kg and designed to be easily deployable to mountain areas. It has Intel i5 CPU having two 2.67 GHz cores and 4 GB RAM. All image processing is done by using a single core of CPU.

Also several in-door tests were conducted on an Intel i7 3.40 GHz desktop PC receiving frames directly from camera. Camera was mounted on the head of a 5 axis CNC having 3x6 meter base area (Figure 4.3). UAV movements were simulated by precision 3D positioning of the CNC head. Aerial scenery was provided by hardcopies of high resolution aerial images printed in various dimensions ranging from 1x1 meter to 2x3 meter. UAV Camera used in this study provides 576 x 720 pixel frames at 25 Hz.

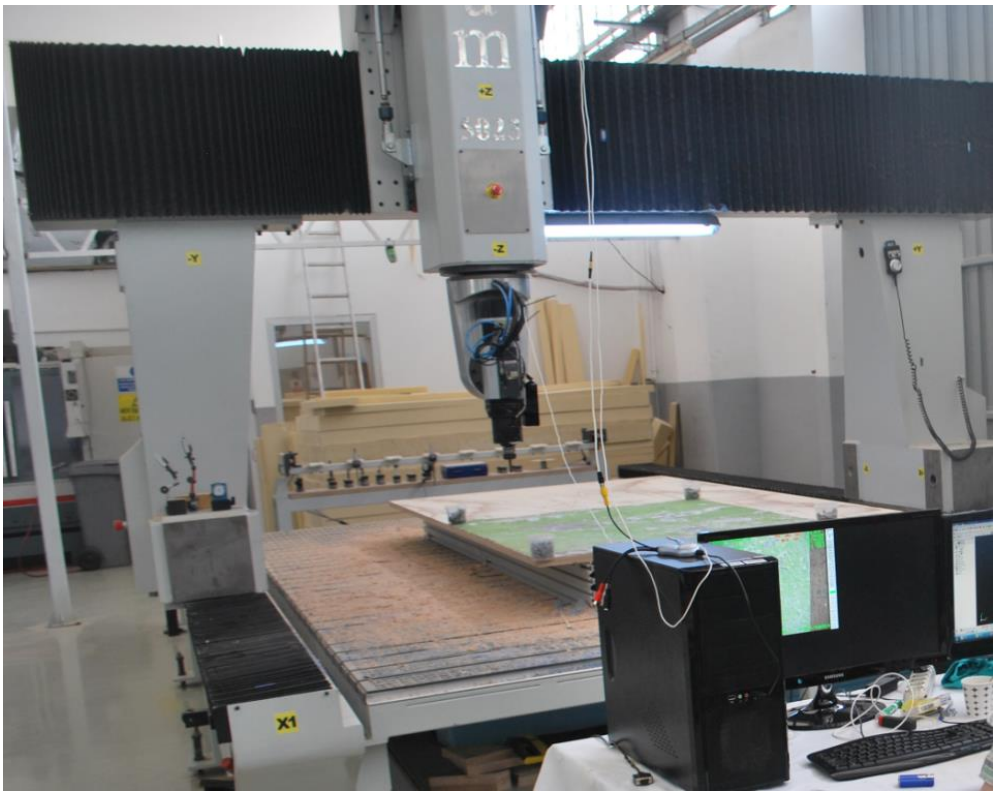


Figure 4.3: 5 axis CNC test set up used in in-door path following test.

4.2 Image Acquisition and Preprocessing

First step in the process is acquisition of the image by on-board camera on UAV. Quality of the acquired images directly affects the performance of any computer vision algorithms so several factors is needed to be considered for successful processing. Two of the most notable factors affect the quality of the images acquired by the Ground Control Station prior to processing. Motion blur caused by the shaky and fast motions of the UAV can severely distort images making them useless for processing and observation. In order to prevent this effect, shutter speed of the on-board camera adjusted to retain image sharpness.

Developed algorithms are designed to be robust enough to handle any decrease in signal to noise ratio caused by reduction in the amount of received light by camera sensor.

Distortions due to communication interference also have a prominent effect on the performance of the processing of the images. In order to deal with this situation, algorithms designed to discard any frames that failed to match and continue with the next frame. If distortions persist for a longer time, algorithm resets itself until a successful matching is obtained.

Apart from a deinterlacing filter no additional filtering is applied in order to avoid computational burden. Any noise affects that are present in the frames are aimed to be handled by robust registration algorithms. In order to make speed and accuracy adjustable, parametric selection of Region of Interest and scaling down factors were added as controls. Field tests showed that actual image motion can be very large in order of 100 pixels resulting in a sharp decrease in the number of matching points fall into ROI which adversely effects success of the image registration. So in most of the actual tests a ROI was not applied. On the other hand as form of achieving good speed increase with an insignificant loss in accuracy for surveillance operations, scaling factor used as a control parameter. In actually tests, generally a scaling factor between 0.75 and 0.5 was employed.

4.3 Direct versus Feature Based Methods for Mini UAV Surveillance

Various different image registration approaches were considered in order to find the most suitable for low altitude fast UAV surveillance. Image registration methods were classified into two main stream groups, namely direct methods and feature based methods at [12]. Direct methods use pixel to pixel matching while feature based methods finds apparent point or regions on the images which are general called as salient point or corner. Later on these points can be tracked similar to direct methods or vectors defining the

features called “descriptors” can be extracted and used for matching. More detailed discussion of image registration approaches is beyond the scope of this study and interested user can find a good overview at [12].

During preliminary tests, in order to find best method suited for mini UAV surveillance, several different approaches were considered and an algorithm utilizing comparing intensity differences in various sized patches by using Sum of Square Differences (SAD), referred in this text as “Block Matching”, were developed [12]. Intensity comparison over a patch was chosen because of its common usage and several search strategies employing different block sizes and numbers such as one to five large blocks concentrated in center, uniformly distributed many small blocks etc. were tested. Results showed that Block Matching variations were not suitable for the task and attention was shifted to feature based methods. Several notable outcomes of the preliminary studies are listed below;

- In order to capture large image motions, search range of the Block Matching algorithm needs to be extended which in return decreases the computation speed very quickly. Although this may be acceptable for small vibrations, during the test it was observed that Block matching had poor performance with mini UAV surveillance footage experiencing severe vibrations and an increase in the search range decreased the computation speed beyond limits of near real-time processing.
- For Block Matching to perform accurately, adequate intensity variation had to be present in the patch. This was not an issue for processing high intensity variation images like urban scenery but during the tests it was observed that accuracy decreased drastically while processing low intensity variation scenery like rural grasslands. Increasing the number of blocks or block size to capture high intensity variation areas also had an adverse effect on computation speed.

- Block matching with single block can be used to estimate two dimensional translational motions. By using many blocks it is possible to estimate slight rotations but in practice it was seen that this method generally failed to cover quick turns due to UAV motion and sudden winds.
- An approach involving first detection than tracking of the interest points can be considered more viable but since this method would also have the weaknesses of direct methods, efforts on finding an appropriate image registration method shifted towards feature based methods for further studies.
- Since the displacement is calculated by matching features it is independent from the actual value of the motion as long as enough matching points still reside in the consecutive frames.
- Even in rural images general there are image regions with relatively more prominent gradient structure and feature matching methods automatically utilizes such kind of areas without any additional adjustment.
- Number of degrees of freedom of the estimated matrix can be adjusted without changing the image registration part of the algorithm.

4.4 Image Registration for UAV Surveillance

Image registration is a well investigated board research that forms the basis of a wide variety of applications. In the image registration component of this study was to find and optimize suitable registration methods so that resultant mosaicing algorithm will satisfy the requirements of mini UAV surveillance missions. Main requirements of the mini UAV surveillance missions considered in this study can be listed as follows.

- Real-time and near real time processing speeds by using a regular PC CPU is required.

- Complex and fast unstructured motion of the UAV and camera should be followed.
- Algorithm needs to perform reasonably well in different lighting and terrain conditions where gradient structures and intensity distribution changes. Especially good performance in the actually operating environments of test UAV is important.
- Registration components should be robust enough to tolerate slight motion blur and other image distortions.
- Overall slight registration errors can be tolerated. At the end, resultant system should enhance the environmental understanding of the operator, providing practical benefits.

After surveying existing state of art algorithms, Harris corner detector [17], FAST detector [18], BRIEF descriptor [21], SIFT detector and descriptor [19] and SURF detector and descriptor [20] were chosen for further investigation. Detailed descriptions of these algorithms are beyond the scope of this paper, but for completeness of the paper, brief descriptions are given. Harris corner detector determines salient points by calculating a score based on trace and determinant of the gradient matrix and is widely used in many applications because of its relatively fast computation with superior performance [17,24]. FAST compares a central pixel to its surrounding circle of pixels and is noticeable with its computation speed [18]. SIFT employs a Hessian based detector and a descriptor that examines smaller feature like structures in a larger patch and are known for its robust performance under scaling and rotation [19]. Also it is relatively slow compared to other registration algorithms examined in this study. SURF is also based on Hessian based feature detection and smaller local features based description and uses integral images and box filters for fast computation [20]. BRIEF utilizes intensity comparison to directly form a binary descriptor for speed increases [21]. Interested reader can find additional information

on respective references and a good survey and comparison in [12,16]. Results of several tests are presented at the results and discussion section.

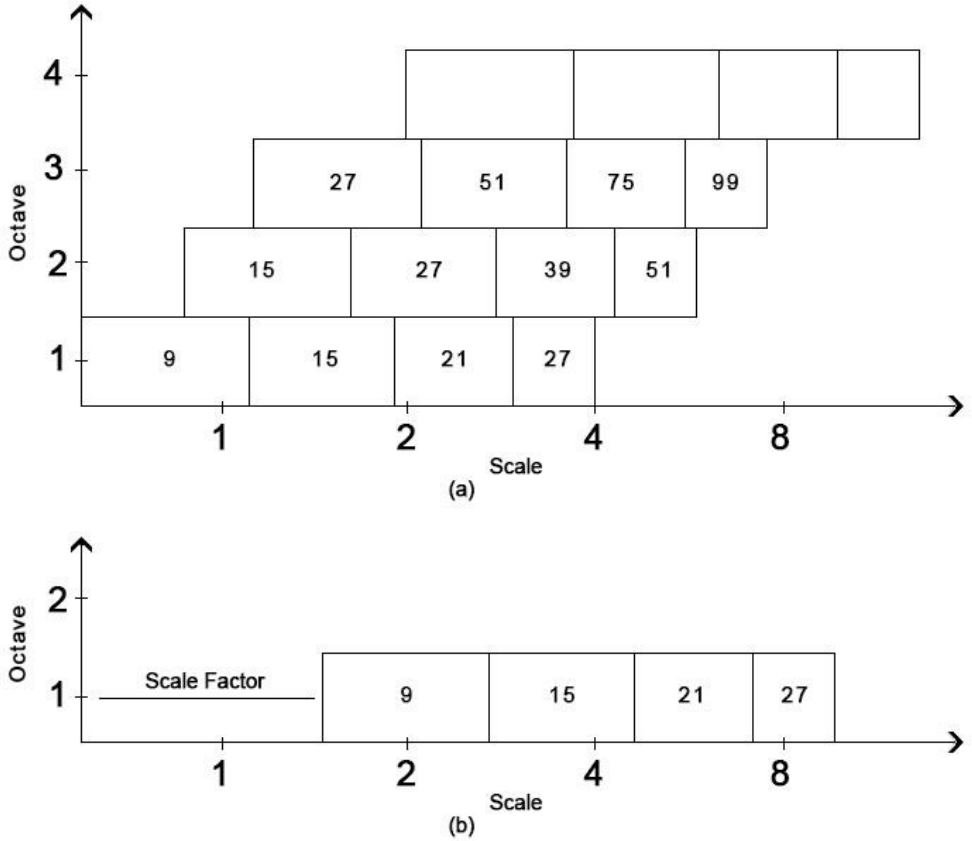


Figure 4.4: Octaves and layers of original SURF (a) (Bay et al. 2008) and Modified Algorithm (b). Numbers in the boxes indicate the sizes of the box filters where original SURF operate through all scale-pyramid levels and Modified Algorithm only performs a quick search through a few levels. Effect of the scaling factor is also presented. Several layer combinations are possible and in experiments processing only two layers is generally employed.

As a result of the performance tests and examination of the studies done by other authors, SURF was selected to be used in the further studies. On the other hand speed of the SURF in its original version was not enough for real-time processing in this test set up. So modifications aimed to gain speed increase as much as possible while losing from accuracy as little as possible were applied.

SURF is based on Hessian based feature detection, smaller local features based description and uses integral images and box filters for fast computation [20]. One key aspect of the SURF algorithm is that it was designed as a generalized detector and descriptor to perform reasonably well in a wide range of images and image motion models. On the other hand in order to develop an image registration algorithm for mini UAV surveillance, a modified detector and descriptor couple that performs well under typical image intensity structures and image motions encountered in an actual surveillance mission is adequate. SURF algorithm searches trough all space scale in order to detect keypoints and is able to match features if there is a great scale difference between base and target images. By investigating the histogram of feature detected distribution on different scales in original study [20] a range with a concentration of feature points were determined. In Modified Algorithm a quick search of few layers of this range was performed so that adequate number of feature points can be extracted from various scenery encountered in UAV flights as shown in Figure 4.4. By using this approach Modified Algorithm performed with an accuracy level comparable to original SURF algorithm at a fraction of its computation time. In order to handle rotations, upright version of the SURF descriptor which is able to cover rotation as much as 15% with a great speed increase [20] was found adequate.

4.5 Fixed versus Adaptive Algorithm

Keypoints were detected by computing a score based on Hessian matrix [20] and value of threshold to determine whether response of the filters at a certain location indicates a valid feature point or not affects the number and quality of features detected. This threshold is referred as "Hessian Threshold" throughout the text. In order to meet speed and accuracy requirements of real time UAV surveillance, two different approaches to determine Hessian threshold were tested. In fixed threshold approach, several tests were conducted to find optimum performing Hessian threshold throughout operational environments. In adaptive threshold approach, an adaptive algorithm was employed to adjust the threshold level in order to meet the gradient nature of the image.

Main motivation behind the fixed threshold was to determine a value that produces robust yet few features so there are enough points for accurate transform estimation even at the low gradient scenery that encountered in the operational but matching component is significantly faster because of reduced number of points. After a series of in-door and flight tests, a threshold of 900 was selected. Although chosen threshold works fine for most of the scenery, it was observed that registration fails at several specific conditions like zooming to grassland where gradients are extremely low. On the other hand main operational area of the test UAV is rural mountain ranges so registering such kind of images can not be neglected. In order to further increase robustness and make the system operational at all possible environments, an adaptive threshold approach was investigated.

In flight tests and in-door experiments, it is observed that algorithm performs fast and accurate when there is about 80 - 120 detected points. So instead of holding Hessian threshold value fixed an adaptive algorithm that holds the number of features found around acceptable levels was developed. In order to create an algorithm that response fast enough to rapid image gradient changes due to the fast movements of UAV yet does not changes its parameters in single corrupted frames, several adaptive approaches such as using

moving averages, fixed step sizes and adaptive step sizes were tested. It was observed that moving averages although robust against single corrupted frames, fails to respond on time at quick scenery changes. On the other hand response time of the single fixed step algorithm can be adjusted by increasing step size and can have a relatively fast response. Yet a large step size create fluctuations at Hessian threshold and a fixed step results a different change in the number of detected points according to the original threshold value to be adjusted.

On the other hand, an adaptive algorithm adjusting its step size according to the current threshold value and difference between target and found number of points was able to create fast responses, relatively robust against corrupted single frames and was able to settle to an optimum value. Because of these characteristics an adaptive step size algorithm was selected for implementation in the final version of the program.

Adaptive step algorithm first starts by calculating the difference $n_{difference}$ between number of points found n_{found} and ideal number of points n_{ideal} (4.1).

$$n_{difference} = n_{found} - n_{ideal} \tag{4.1}$$

If the difference is smaller than a certain valuen_{max}, no adjustment is made in order to prevent Hessian threshold changes for small key point number fluctuations (4.2).

$$n_{difference} > n_{max} \tag{4.2}$$

If this distance is greater than a defined limit, it is compare to a step size adjustment value n_{dist} to calculate step size adjustment factor k_{hes} by using (4.3).

$$k_{hes} = n_{difference}/n_{dist} \tag{4.3}$$

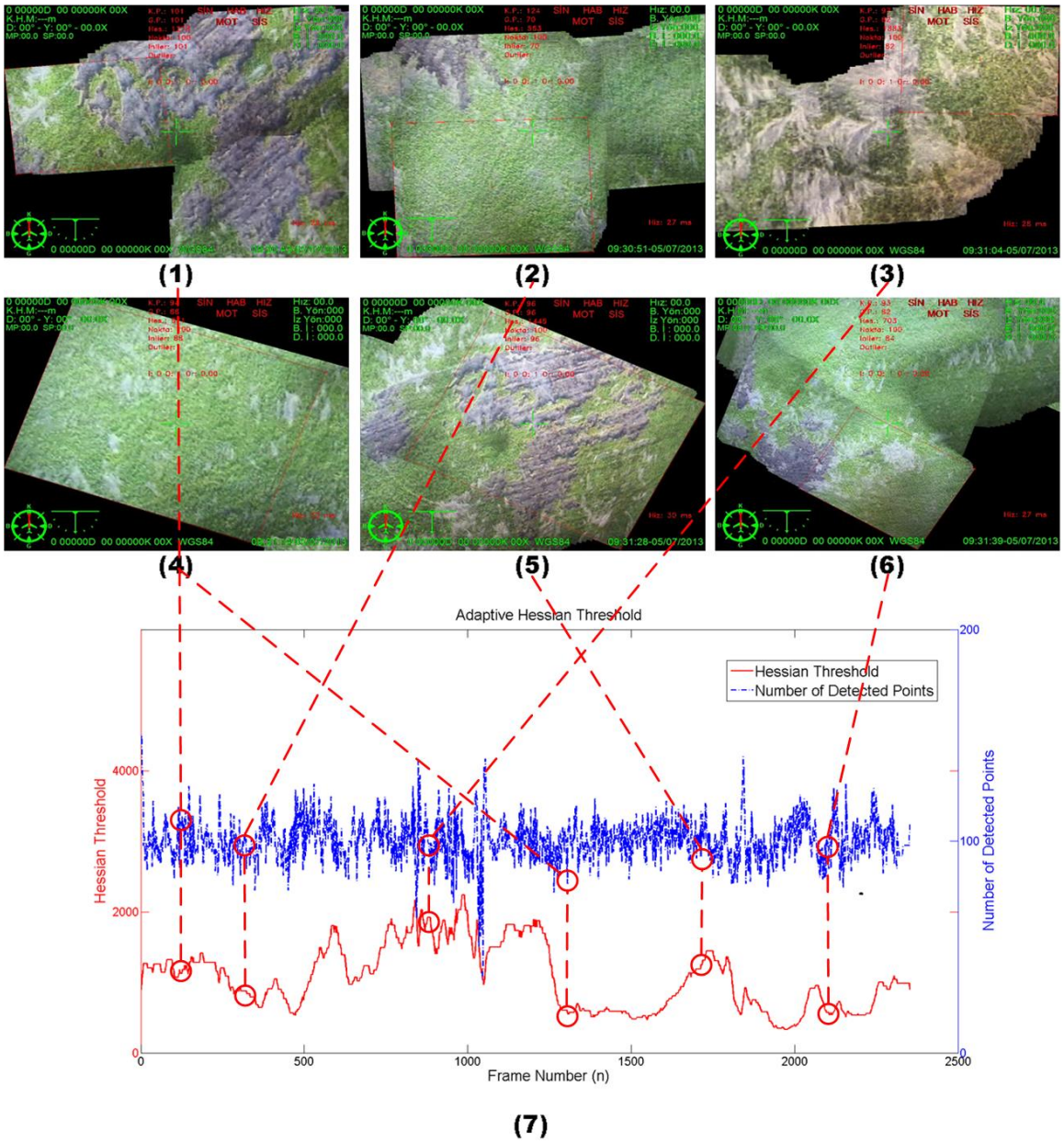


Figure 4.5: Changes in the number of detected points and Hessian threshold as a response to changes in image intensity distribution. Test was conducted by following a path on printed hardcopy of image shown at Figure 2.1 (2). Total imaging time was 68 seconds and 2353 frames were processed. Frames positions from (1) to (6) are marked on graph (7)

to show Hessian threshold and number of detected points. Fluctuations created by changes in intensity nature and response of the adaptive algorithm to adjust Hessian threshold in order to keep the number of detected points around 100 can be seen at (7).

If k_{hes} is greater than or equal to one, maximum step size is used. If it is smaller than one, this fraction is used to adjust the step size so that smaller step changes are used when number of found points is closer to the ideal number of points.

$$k_{hes} = \begin{cases} k_{hes} > 1 \, , k_{hes} = 1 \\ k_{hes} \leq 1 \, , \; k_{hes} = \; k_{hes} \end{cases} \tag{4.4}$$

One last factor is employed to determine the ratio of current Hessian threshold value compared to the maximum allowable Hessian threshold value. A percentage factor utilizing ration of current and maximum Hessian threshold values is calculated and used to adjust the step size as shown in (4.5).

$$h_{update} = h_{current} \left(1 + \left(k_{hes} h_{step}\right)\right) \tag{4.5}$$

Similar equations are employed for negative adjustments by utilizing a minimum boundary n_{min} (4.6) and a reducing equation (4.7).

$$n_{difference} < n_{min} \tag{4.6}$$

$$h_{update} = h_{current} \left(1 - \left(k_{hes} h_{step}\right)\right) \tag{4.7}$$

With this final adjustment, if the overall Hessian threshold value is small, smaller step adjustments used while if the threshold value is large, larger steps are employed, resulting in an adaptive adjustment on number of detected points throughout the Hessian threshold range. Response of the Adaptive algorithm to changing intensity nature can be seen at Figure 4.5.

4.6 Transform Estimation and Constructing Mosaics

Frame to frame correspondences were established by an exhaustive matching algorithm calculating Euclidean distances. For optimization purposes, detector parameters were adjusted and a limit to the maximum number of features was applied in order to decrease the number of detected features.

In a typical feature matching process, resultant matched pair set contains both outliers and inliers. These outliers usually refined by an estimation algorithm such as RANSAC but success of these algorithms is also affected by the percentage of outliers in a matching set [27]. In order to provide RANSAC with a better set of matches, a double-matching method followed by a refinement process was employed. As a result of the matching process, point to point correspondences between every feature in the base frame and two alternative features in the target frame were established. Later maximum and minimum scores of resultant set of correspondences were determined and used as the range of the scores. A user defined percentage of the scores was selected and used as a threshold value. Score of every matching pair in the set was compared to this threshold and only pairs having lower score than the threshold are used to produce a refined set and supplied to RANSAC [27]. Since at least half of the unrefined double match set is outliers, for a successful RANSAC estimation threshold percentage should be lower than 50%. In practice it was seen that a threshold of 10% leaves to few matching pairs resulting in the degeneration of the

estimated matrix. On the other hand a threshold of 40% or 50% still preserves lots of mismatches and estimation start to produce registration errors. Algorithm was observed to work robustly between 20% and 30%.

For modeling image motions, homography, similarity and Euclidean transformations were tested. Homography has enough degrees of freedom to cover unstructured image motions with perspective effects but conversion to homogenous coordinates brings in additional computational burden which reduces computation speed. Also errors in homography has a tendency to skew and distort images in a way that retains in the rest of the sequence and produces undesired results. This effect is also reported by other authors [1] who preferred Euclidean transformation for mapping. While Euclidean transformation is useful when mapping a relatively flat terrain with a UAV flying at a fixed height, operational conditions of UAV used in this study requires additional degrees of freedom. Similarity transforms on the other hand is faster to compute and has two translational, one rotation and one scale degree of freedom. These degrees of freedom capture most of the critical image motion providing a better surveillance and also prevent excessive image distortion due to errors in image registration. Because of good balance between number of degrees of freedom to cover image motions and speed of computation, similarity transform was chosen.

For blending of frames several blending approaches are available [12]. Algorithms surveyed in [12] generally focus on preventing blending artifacts such as “ghosts” to construct better looking results. On the other hand for surveillance purposes, presenting user with the most recent information is thought to be critical and replacing old pixels in the mosaic image with the newer ones in the frame without any other operation was chosen. Speed of the transformation process is independent from the size of the resultant mosaic.

4.7 Operating Modes

Several operating modes were developed in order to provide user with better video enchantment in different situations.

4.7.1 Classical Mosaicing Mode

Mosaicing mode is used to provide user with a larger view of the scenery and natural image stabilization (Figure 4.7). At the first pass of the algorithm, acquired frame is placed at the center of the mosaic image and frame to frame transformation matrix $F(0)$ is assigned to identity matrix I (4.8). For the consecutive passes, frame to frame transformation matrix $F(t)$ is estimated and multiplied with the total transformation matrix of previous frame $R(t-1)$ as shown in (4.9). Resultant total transformation matrix $R(t)$ defines the position of the newest frame with respect to the reference frame.

$$F(0) = I \tag{4.8}$$

$$R(t) = F(t) * R(t-1) \tag{4.9}$$

After calculation of the total transformation matrix, four corners of the frame are transformed in order to determine the position of the new frame with respect to mosaic image. If frame is transformed outside the boundaries of mosaic, distance values $dx(t)$ and $dy(t)$ in x and y directions is calculated as shown in Figure 4.6.

Translation matrix $T(t)$ is formed as shown in (4.10) to slide the mosaic and frame respectively in order to fully enclose the newest frame and used to calculate the final transformation matrix $M(t)$ (4.11). Any old pixel values pushed outside of mosaic image are erased from memory.

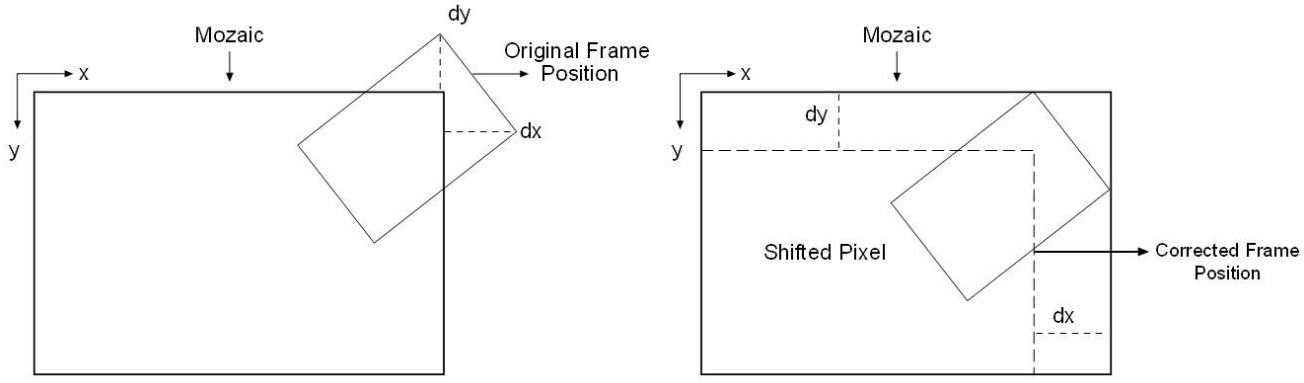


Figure 4.6: Mosaic schematic. dx and dy in the orginal transformation position (left) indicate the distance between the outside corners of the frame and mosaic boundaries. Final position (right) shows shifting of the mosaic pixels in order to enclose the newest frame.

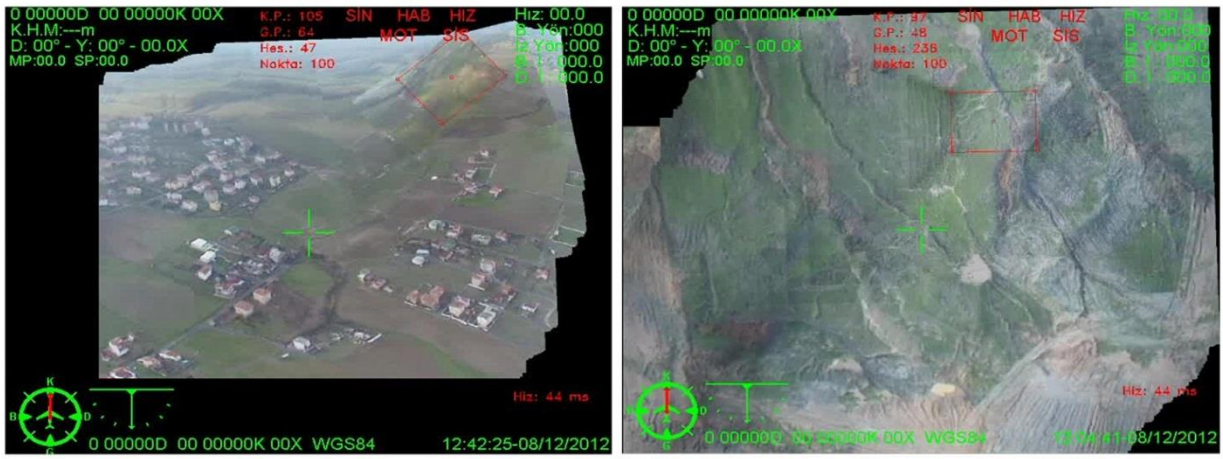


Figure 4.7: Two mosaic images constructed in flight tests. (1) shows a sequence captured while free navigation and (2) shows mapping with downward looking camera.

$$T(t) = \begin{bmatrix} 1 & 0 & dx(t) \\ 0 & 1 & dy(t) \\ 0 & 0 & 1 \end{bmatrix} \tag{4.10}$$

$$M(t) = T(t) * R(t) \tag{4.11}$$

After formation of the new mosaic image feature points of the new frame are preserved for the second pass and the next frame is acquired. No other information is held on the memory and old frame is erased.

This method provides a panning effect with UAV and camera movements. A different version of this mode which holds a large mosaic at the memory and provides a portion of it to user were also tested but discarded because of the error accumulation effects were apparent when revisiting a previously constructed region.

Transformation of the image is done by using a patch of the mosaic where only affected pixels are considered. Since computation cost is independent from the size of the unaffected area, it is possible to create very large mosaic images without a significant loss in speed. On the other hand it was observed that without using any additional global correction method, error accumulation effects becomes apparent with increasing mosaic size making it unpractical to use. Figure 4.7 shows two example mosaics created in real-time with an average processing speed of 44 ms per frame.

Figure 4.8 shows the working of the developed system in surveillance of a local fire. Examples are taken from a 7500 frame sequence where operator observes the fire site with an UAV moving toward the location. At Figure 4.8 (1) operator makes a wide view sweep of the area observing the fire sight and its surroundings. Later operator zooms in order to conduct a closer investigation of the area and current frame show with the red rectangular in the Figure 4.8 (2) shrinks with respect to mosaic accurately following zooming behavior. Because of the vibrations and strong air currents, operator slightly misses the source of the

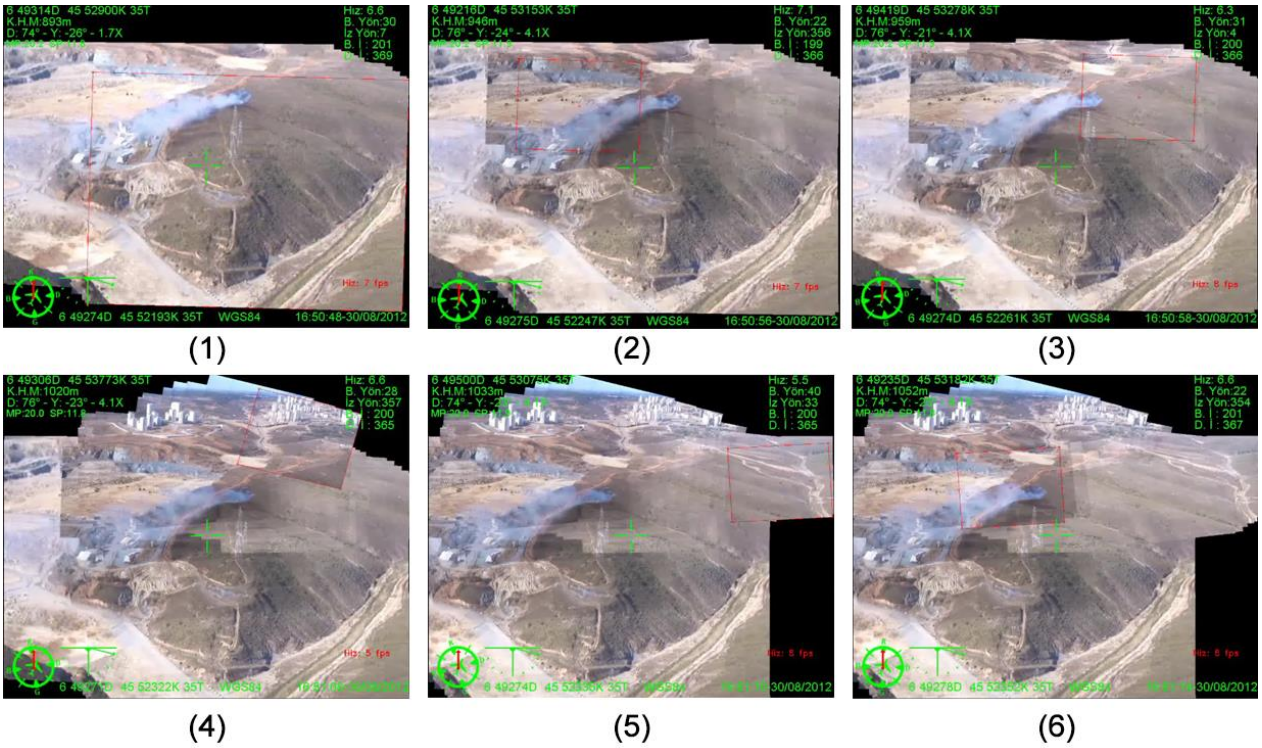


Figure 4.8: 6 frames of a mosaic sequence while operator observes a fire site. Sequence demonstrates all degrees of freedom of the developed system including scaling at an actual surveillance operation.

fire when zooming in Figure 4.8 (2) and searches toward right at Figure 4.8 (3). In order to increase environmental awareness, operator first searches upward revealing two city blocks at Figure 4.8 (4) and toward right revealing a road at Figure 4.8 (5). In order to observe the most recent situation at the fire site, operator navigates back to fire at Figure 4.8 (6). It should be noted that although slight registration errors due to error accumulation and viewing from a moving camera point can be seen when previously captured areas are revisited, developed system accurately capture all image motions including zooming, rotation and translation. Without using a real time mosaicing system, operator could easily lose the track of the fire location during the motions presented in this sequence. On the

other hand by using the methods presented here, even if the current view is far away from the actual objects of interest, operator is able to track its way back to starting location and even with the presence of excessive vibration and sudden wind currents, quality of the surveillance is not degraded.

4.7.2 Rotating Mosaic Mode

Rotating mosaics mode is developed in order to align the mosaic image to UAV orientation while filtering out small vibrations in rotational degree of freedom (Figure 4.9). In rotating mosaics mode, steps described in previous mosaicing mode is repeated and equations (4.8) – (4.11) is used in order to calculate corrected frame transformation matrix $M(t)$. Since similarity transforms are used for modeling image motion resultant transform $M(t)$ is in the structure of (4.12) making it possible to calculate frame rotation component.

$$M(t) = \begin{bmatrix} s\cos\alpha & -s\sin\alpha & d_x \\ s\sin\alpha & s\cos\alpha & d_y \\ 0 & 0 & 1 \end{bmatrix} \tag{4.12}$$

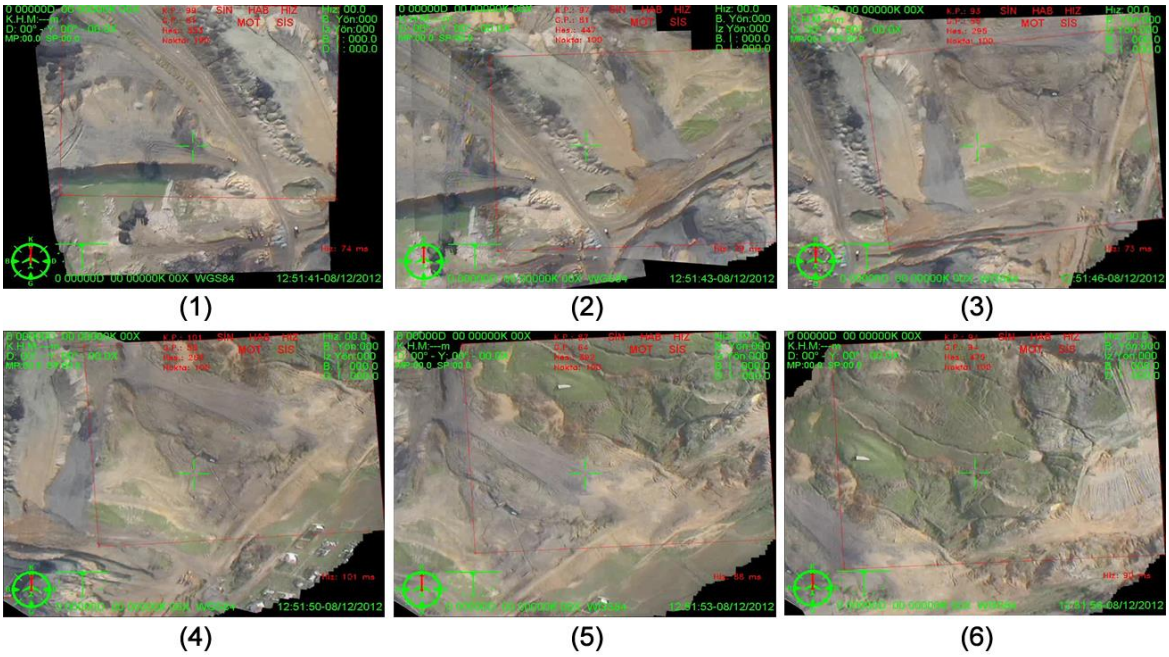


Figure 4.9: Six snapshots of Rotating Mosaic sequence. Rotation effect of the background mosaic is clearly visible from (1) to (6).

Angular difference $\alpha'(t)$ is calculated by using (4.13) where refined rotation $\alpha''(t-1)$ of the previous frame subtracted from current rotation degree $\alpha(t)$ of the transformation matrix $M(t)$.

$$\alpha'(t) = \alpha(t) - \alpha''(t-1) \tag{4.13}$$

Later on current refined rotation $\alpha''(t)$ is calculated by using (4.14) where k is the damping coefficient for smoothing which is typically 0.1.

$$\alpha''(t) = k.\alpha'(t) + \alpha''(t-1) \tag{4.14}$$

A refined transformation matrix $K(t)$ is constructed by using refined rotation $\alpha''(t)$ and scale s, translation t_x and $t_y$parameters of the frame transformation matrix $M(t)$ (4.15). This matrix defines a new position to the frame where rotation component is of the original transformation is split between frame and mosaic and used to determine a display area. Schematic of the method is shown at Figure 4.10.

$$K(t) = \begin{bmatrix} s\cos\alpha'' & -s\sin\alpha'' & d_x \\ s\sin\alpha'' & s\cos\alpha'' & d_y \\ 0 & 0 & 1 \end{bmatrix} \tag{4.15}$$

An alternative way for working with the rotation component α is to apply total rotation of $M(t)$ to display window which produces a mosaic view holding the plane orientation upright

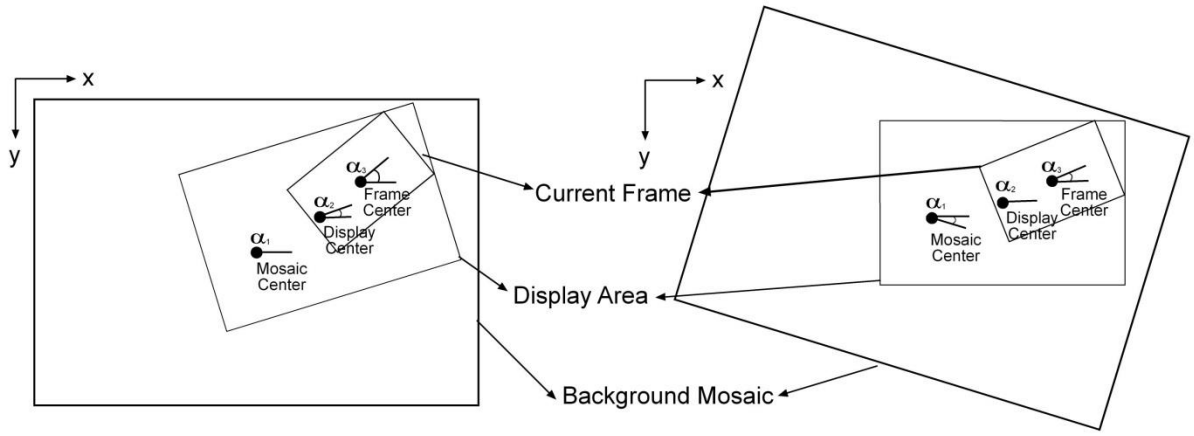


Figure 4.10: Schematic of Rotating Mosaics. Schematic of the original position (left) shows the angles of background mosaic, display area and current frame as how they are held in memory. Schematic of display version (right) shows relative angles as the way they are provided to user.

while mosaic image around rotates according to UAV rotations. Drawback of this approach is rotational vibrations are directly reflected to mosaic giving an undesired vibrating view. On the other hand using a damping factor of $k = 0.1$ is found to produce a view where mosaic image follows camera alignment with a smooth movement preventing disorientation of the operator.

4.7.3 Stabilization Mode

Stabilization mode is developed for providing user with vibration smoothed camera frames while following the general camera motion (Figure 4.11). Although image stabilization is a well-developed area with implementations on commercial products, fast unstructured motion of the UAV coupled with severe vibrations creates a problem with different nature. Stabilization during aerial surveillance usually can be achieved by use of mechanical gimbal systems and electronic stabilization (Kumar et al. 2001). One other hand high performance gimbals are very expensive, ranging from $300 000 to $500 000 (Kumar et al. 2001) and may not be installed on mini UAVs due to size and weight constraints. Contrary to this approach, in this study, developed system was aimed to require no additional specialized hardware installment on UAV and solve stabilization problem algorithmically. One important aspect of UAV aerial video footage stabilization is differentiating between actual camera movements and undesired vibration and jumps at real time. A sharp sudden image movement can be both due to distortion effects such as sudden air currents and amplified vibrations as well as desired actions such as UAV movements and adjustment of the camera gimbal by UAV operator. Although this distinction can easily be made by offline processing of the complete image sequence, since the next frame information is not available in real time processing, a method that can handle both cases needed to be developed. While (Kumar et al. 2001) addresses this problem by decomposing image movement into high and low temporal frequency components and damping the high

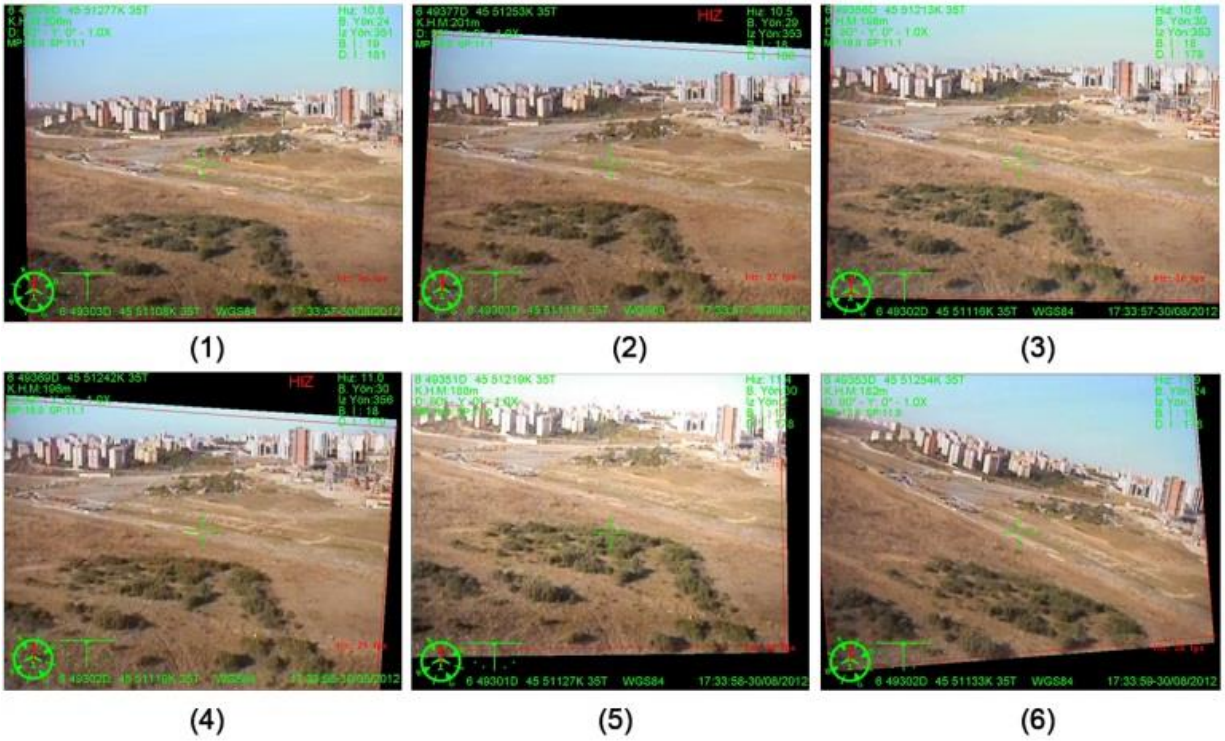


Figure 4.11: Six frames of a stabilization sequence. Sequence was captured during landing approach of mini UAV where severe vibration effects are visible.

frequency component at the absence of any additional information about the nature of motion, in this study a novel and practical approach utilizing position of the transformed image for estimating filtering coefficients is developed. Another noticeable issue about video stabilization is estimation of the missing pixels which is addressed at [9] by a method named “Motion Inpainting”. While authors at [9] utilized offline processing and complete video sequence to estimate missing pixels, in this study this problem is addressed by a use of mosaics which is described in detail at Section 12.4.

In order to find an appropriate approach, methods reducing the scale, translational and rotational component of the resultant total frame to frame transformation matrix by a fixed amount and by a function response are investigated. When the motion parameters are

reduced with a small fixed percentage, vibrations were still present in the resultant display. Increasing the reduction percentage decreased the vibrations but also created a lag at following large image motions. In order to create a stabilization method with a different response to different conditions, a filtering function with a non-linear response to amount of image motion was developed. Schematic of the method presented in Figure 4.12.

Resultant total frame transformation $R(t)$ is calculated by using (4.16) where $S(t-1)$ is the previous filtered stabilization transformation and $F(t)$ is the frame to frame transformation where $S(0) = I$.

$$R(t) = F(t) * S(t-1) \tag{4.16}$$

Since both $F(t)$ and $S(t-1)$ are similarity transforms, resultant total frame to frame transformation $R(t)$ is also a similarity transform in the form of (4.17) enabling calculation of scale s, rotation α and translation t_x and t_y parameters.

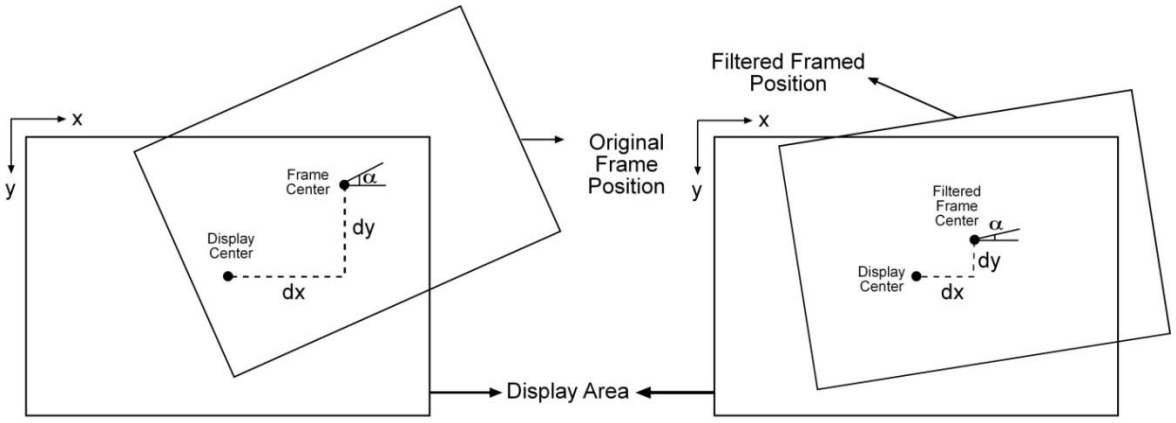


Figure 4.12: Schematic of Stabilization mode where original position (left) and filtered position (right) shows the difference between center positions and frame angles after using filtering function.

$$R(t) = \begin{bmatrix} s\cos\alpha & -s\sin\alpha & d_x \\ s\sin\alpha & s\cos\alpha & d_y \\ 0 & 0 & 1 \end{bmatrix} \tag{4.17}$$

Later, translation of the center point of newest frame $d(t)$ is calculated by using (4.18) where $dx(t)$ and $dy(t)$ are distance between the center point of the transformed frame and display in x and y directions respectively.

$$d(t) = \sqrt{dx(t)^2 + dy(t)^2} \tag{4.18}$$

Calculated translation value $d(t)$ is supplied in to the filtering function (4.19) where $k_s = 0.9$, $c = 2$, $n = 9$ and $m = 4$ in order to calculate reduction coefficient r. Reduction coefficients for scale and rotation parameters also been calculated in a similar fashion. Functions having different parameters can be seen at Figure 4.13.

$$r\big(t, d(t)\big) = \frac{k_s}{\left(1+(c.10^{-n}.d(t)^m)\right)} \tag{4.19}$$

Filtered translationd'_x, d'_y, rotation α', and scale s' components are calculated according to (4.20) and later used to construct filtered stabilization transformation $S(t)$ as seen on (4.21).

$$d'_{x,y} = r\big(t, d(t)\big).\, d_{x,y} \tag{4.20}$$

$$S(t) = \begin{bmatrix} s'\cos\alpha' & -s'\sin\alpha' & d'_x \\ s'\sin\alpha' & s'\cos\alpha' & d'_y \\ 0 & 0 & 1 \end{bmatrix} \tag{4.21}$$

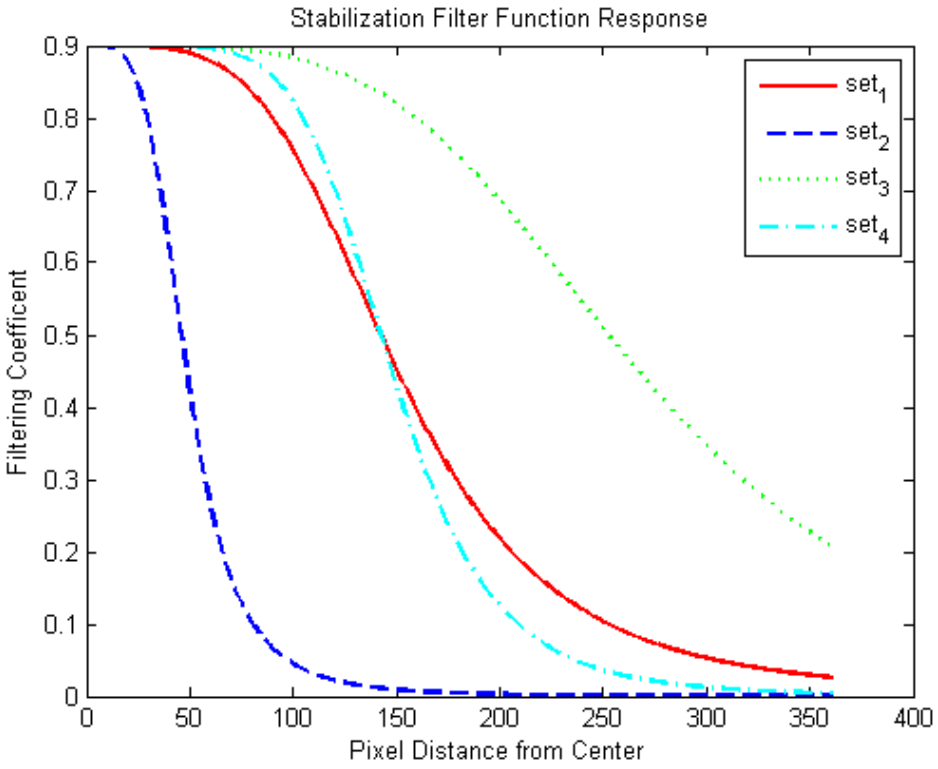


Figure 4.13: Stabilization function response where parameter sets for individual functions are: $\text{set}_1 \text{k}_s = 0.9$, $c = 2$, $n = 9$ and $m = 4$, set_2 $\text{k}_s = 0.9$, $c = 2$, $n = 7$ and $m = 4$, set_3 $\text{k}_s = 0.9$, $c = 2$, $n = 10$ and $m = 4$, set_4 $\text{k}_s = 0.9$, $c = 1$, $n = 13$ and $m = 6$.

Function (4.19) is designed to provide low reduction at the central region of the display area while gradually increase the value when the frame moves near the edges and its coefficients are determined by experimental search through its parameter space. Response of the filtering function with different coefficients with respect to location of the central point of the new frame can be found at Figure 4.13. Underlying principle of this method depends of the image motion behavior observed in UAV flights. Regular low amplitude image vibration motions are observed to not to exceed image boundaries even if they are

amplified by zooming in actual UAV flights. Sudden wind flows on the other hand created large image motions which also tend to come back to its starting position relatively quickly, in less than one second, because of the vehicle stabilization algorithms present in low level flight controls. On the other hand controlled motions such has UAV and gimbal movements has a tendency to stop at a different positions then starting point or continue movement. So a filtering function that produces low reduction near the center of the display area and gradually increase its reduction percentage near the edges of the display area provides filtering of the regular vibrations, robustness against sudden wind currents and still able to follow the controlled camera movements.

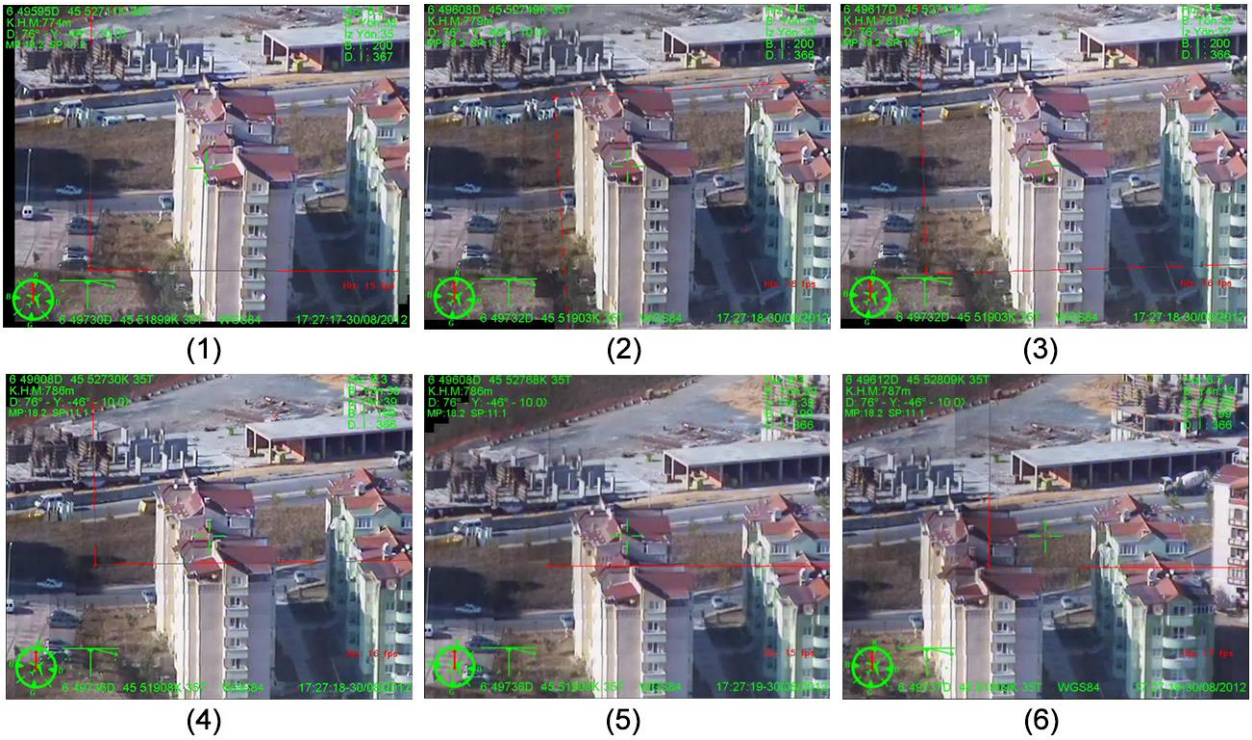


Figure 4.14: Six frames taken from a hybrid mode sequence. Sequence is taken when UAV used to examine buildings when flying forward to them with a camera at maximum zoom.

4.7.4 Hybrid Mode

Although stabilization mode provides good vibration filtering with adequate camera following, it was observed that black background due to the cropping of the frame was distracting for some users. In order to prevent this effect and construct a background by using previous frames information, a stabilization mode based on mosaicing principle was developed (Figure 4.14). Hybrid mode uses a fixed size background mosaic in order to record frame information. A frame sized portion of the mosaic is extracted and provided to user (Figure 4.15). With this approach frame sized still looking images can be constructed and it is observed to be very useful at surveillance of distant fixed objects with zoomed camera view. On the other hand, in fast moving scenery and with scale affects, hybrid mode starts to behave much like classical mosaicing mode.

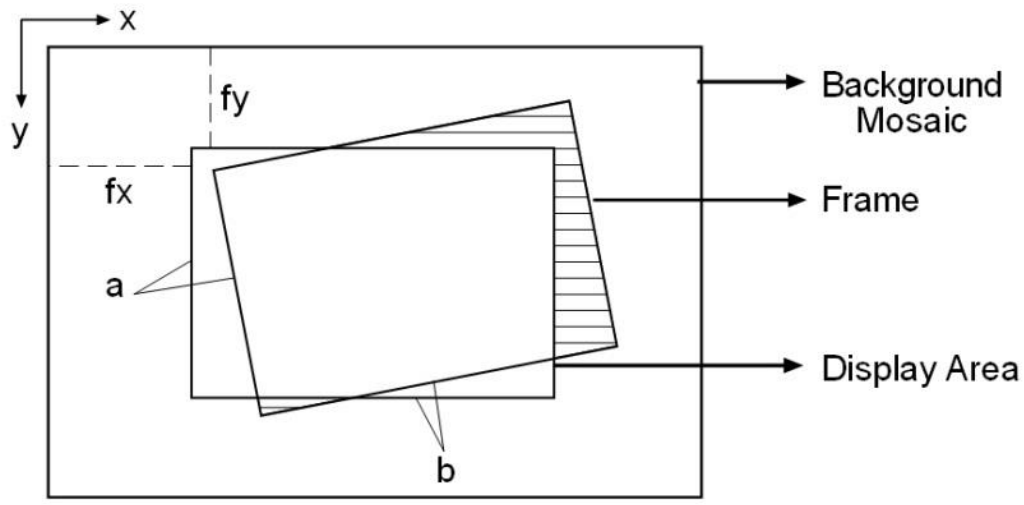


Figure 4.15: Schematic of Hybrid mode. Lined area indicates the portion of the current frame not visible to user. Sides of the display area and frame a and b are equal and fx and fy indicate the distance between the display window and background mosaic.

4.8 Infrared Mosaics Basics

While most of the tests in this study were conducted by using day light cameras, in actual surveillance missions infrared cameras present great importance because of their unique advantages. An object having a temperature difference with its surrounding is easy to be noticed in an infrared image, while it can be hard to distinguish from its surroundings in a day light aerial image. This factor creates a tendency toward extended usage of infrared cameras even in day time for border patrolling of rural areas to detect human presence so a practical image stabilization and mosaicing system needs to cover IR images as well. On the other hand, surprisingly there is a lack of literature on aerial mosaicing and stabilization of infrared images by using computer vision. This may be due to the fact that nature of the IR images is different than daylight images which traditionally most of the conventional detectors and descriptors were developed and tested on. In order to develop an applicable system, type of IR images that are encountered in surveillance mission needs to be investigated.

As it can be seen from the intensity graphs at Figure 4.16, while day light images have relatively rapid and apparent intensity variations, IR images combine smooth and scarce intensity variations with sharp increases at hot areas and uniform areas in between. When 2D FFT graphs of the same scene captured by both day light and IR camera compared, difference at frequency distributions is similar to the difference at frequency distributions of low and high variation images described before with addition of artifacts produced by sharp bright areas. This makes day light images more suitable for feature detectors. On the other hand a feature detector utilizing space-scale should be able to capture features at an IR image.

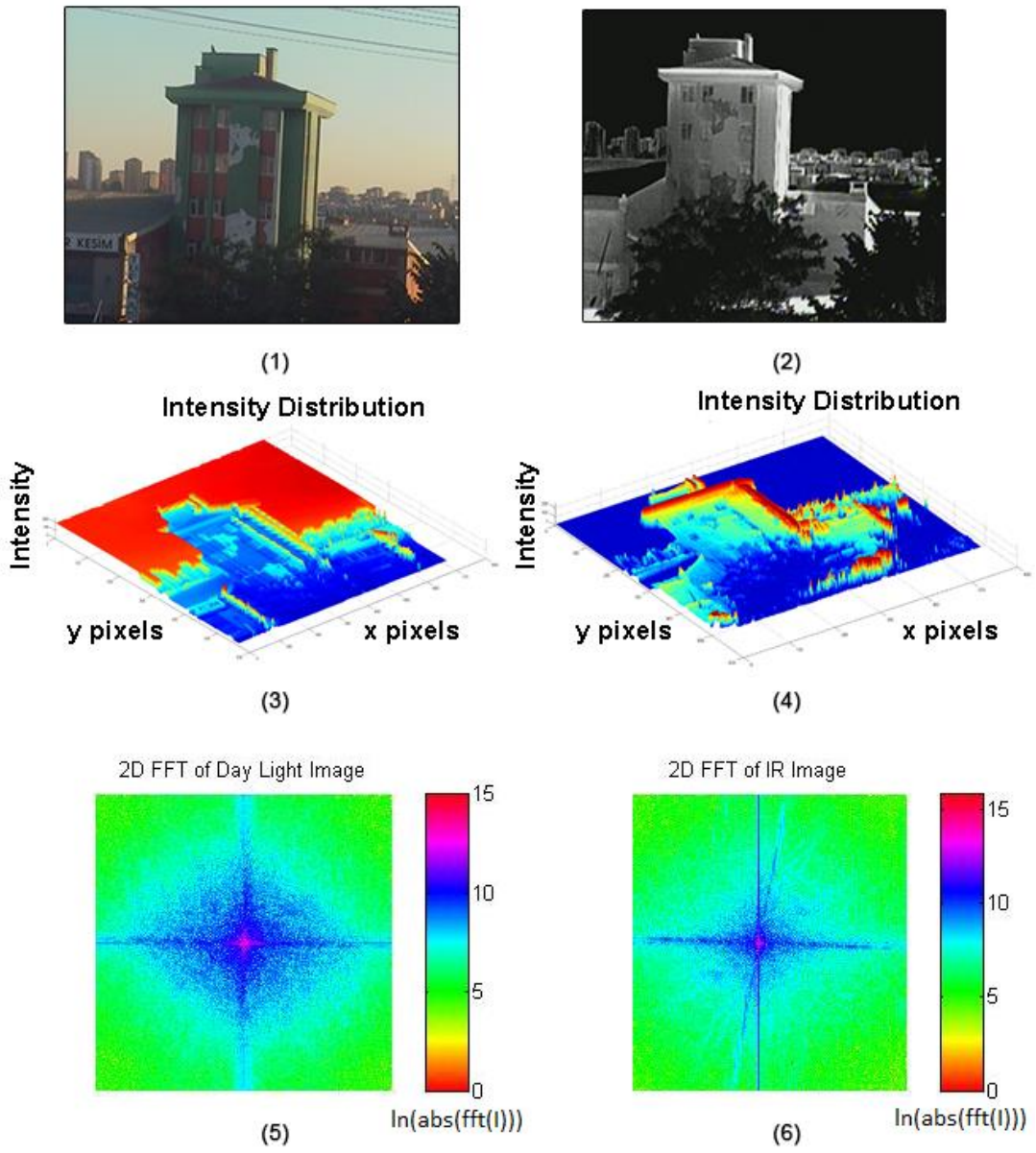


Figure 4.16: Comparison between day light and infrared images. Same scenery was captured by both day light (1) and infrared (2) cameras. Intensity graphs (3) and (4) and 2D FFT of the images provide an example to the difference of intensity distributions between two image types.

When the system developed by using day light images tested with IR camera, it was observed that image registration fails most of the time making overall working as the program in effective. When details of the matching process are investigated, it was seen that number of feature points found dropped sharply in comparison to the day light images. Lowering the hessian threshold values increases number of features found but matching process still stays too fragile. This situation is due to the fact that decreases Hessian threshold causes detection of less distinctive features which in turn increases the number of outliers.

Approach that was taken in order to successfully was to preprocess the infrared images in order to create a gradient distribution comparable to day light images. This is achieved by either down scaling the image so that gradient changed appears more rapidly at the same number or pixels or increasing the box filter sizes so that more apparent gradient change is captured in filter kernel. Both approaches result in the same effect and downsizing before construction of the integral image is chosen because speed increases it provides. This process can be viewed as shifting the range of the image pyramid that was processed in order to find most optimum number of feature. Resultant images still produces fewer number of feature points compared to day light images with modified algorithm but a scaling factor of 0.5 is found to produce adequate results in flight tests. One notable point was that multi-scale algorithms covering all levels of the image pyramid such as SIFT and original SURF had a better performance in terms of detecting and matching features.

Chapter 5

RESULTS AND DISCUSSION

5.1 Introduction

Methods developed for real-time mosaicing and stabilization in UAV surveillance and various existing state of art image registration algorithms were tested in both in-door experimental setups and flight tests. Some of the results are presented below.

5.2 Practical Uses

Main focus of this study was to create a stabilization and mosaicing system that would enhance the real surveillance missions. Because of this factor, throughout the development of this system, advices from actual field operators were used extensively. All of the modes and functionality of the program was developed to meet a real surveillance mission needs. In this section some of the key benefits of the system that makes a great difference in UAV surveillance are listed.

5.2.1 Vibration Smoothing at Fast Movements

Under mild wind conditions looking from wider camera views, vibration effects on UAV video footage seems to be minimal and does not require additional stabilization. On the other hand if the view is zoomed, small vibrations present at the vehicle results in large

image movements. It was observed that is such situations, unprocessed video footage becomes very hard to comprehend while stabilized video footage with camera following function is able to provide user far more understandable view. Figure 4.11 is a sequence of such kind of a situation at UAV landing phase.

5.2.2 Scene Fixing

One of the primary tasks that occur in surveillance missions frequently is zooming and fixing the camera to a distance object for examination. Drawback of using mini UAVs in such kind of task is the increased image vibrations. Stabilization mode can be utilized is such kind of situations but cropping affects and shaky movements in the black background is found to be confusing by some of the users. On the other hand hybrid mode is observed to provide user with a still looking view because of the background construction by using mosaicing principle. Figure 4.14 is a sequence of UAV images where hybrid mode is effectively utilized for stabilization.

5.2.3 Detection of Moving Object that are Seen in the Video for a very Short Time

Mosaicing modes of the program also were tested under some real surveillance scenarios. One of the key benefits of the aerial image mosaics is extending duration of the objects visible to the user. During the shaky movements of the image, moving objects can appear for a very short time in user screen and then disappear making detection by

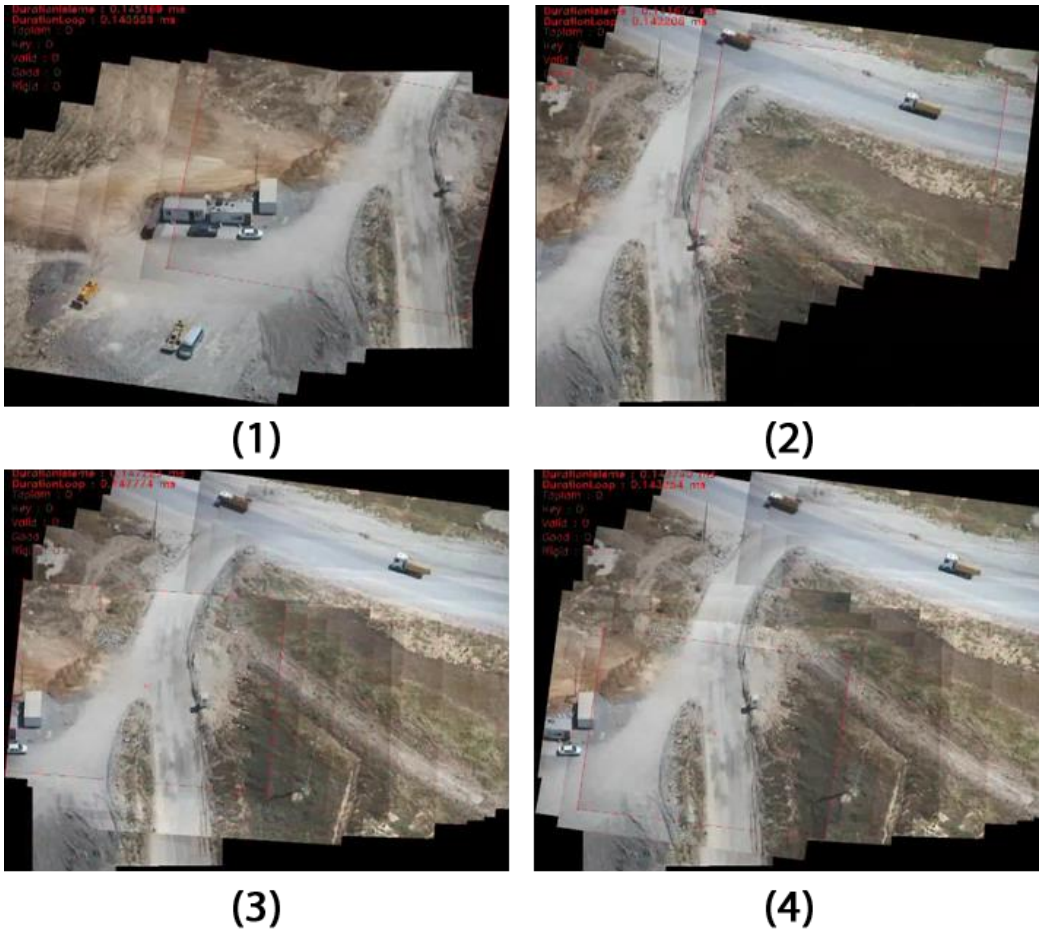


Figure 5.1 Moving objects that are captured in Mosaic image. Total duration of the sequence was 10 seconds. (1) shows the view of a construction site at first seconds. At 3th, operator moved camera upward to briefly reveal two moving tracks for less than 1 second (2). At 6^{th} (3) and 8^{th} (4) seconds while camera view was at construction site, trucks were still visible to operator. Average processing speed was 135 ms per frame.

operators harder. Local mosaics are able to capture such kind of objects making them visible to operator for an extended period of time. In the short sequence below, two trucks can be briefly seen at the original video for less than 1 second. In the mosaic image constructed by using the same frames, appearance of trucks is clearly recorded and can be seen for approximately 5 seconds.

5.2.4 Mapping

Another key benefit of constructing image mosaics is creating a temporary map of the area. Since UAVs are generally equipped with narrow field of view of cameras in order to provide enough zoom for aerial imaging, operators lacks environmental awareness of the surveillance area. By integrating individual frames into an image mosaic a temporary map of the surveyed area is constructed. Figure 4.7 two mosaics that are such kind of temporary maps. It should be noted that dimensions of the mosaic image can be increased without having drastic effects on computation speed enabling larger maps to be produced.

5.2.5 Position Finding

One of the key advantages of mosaics is being able to determine the position of a zoomed view with respect to the general view of the landscape. During a surveillance mission, it is usually required to zoom in the camera view in order to get a closer look on the object of the interest. During this process, field of view decreases rapidly and with the presence of vibrations, it is easy to be disoriented and lost the target. On the other hand mosaic image clearly shows the position of the currently view location with respect to target location. Below sequence shows such an actual event occurred during flight test. Operator in this sequence is trying to locate and get a closer look of a black van and car located at the top of the screen (1). As he zooms in, severe vibrations cause him to loose the targets (2). Later on he continues to search for the target but fails to re-find it (3). On the other hand location of the current scene with respect to target area is clearly visible in the mosaic image (4).

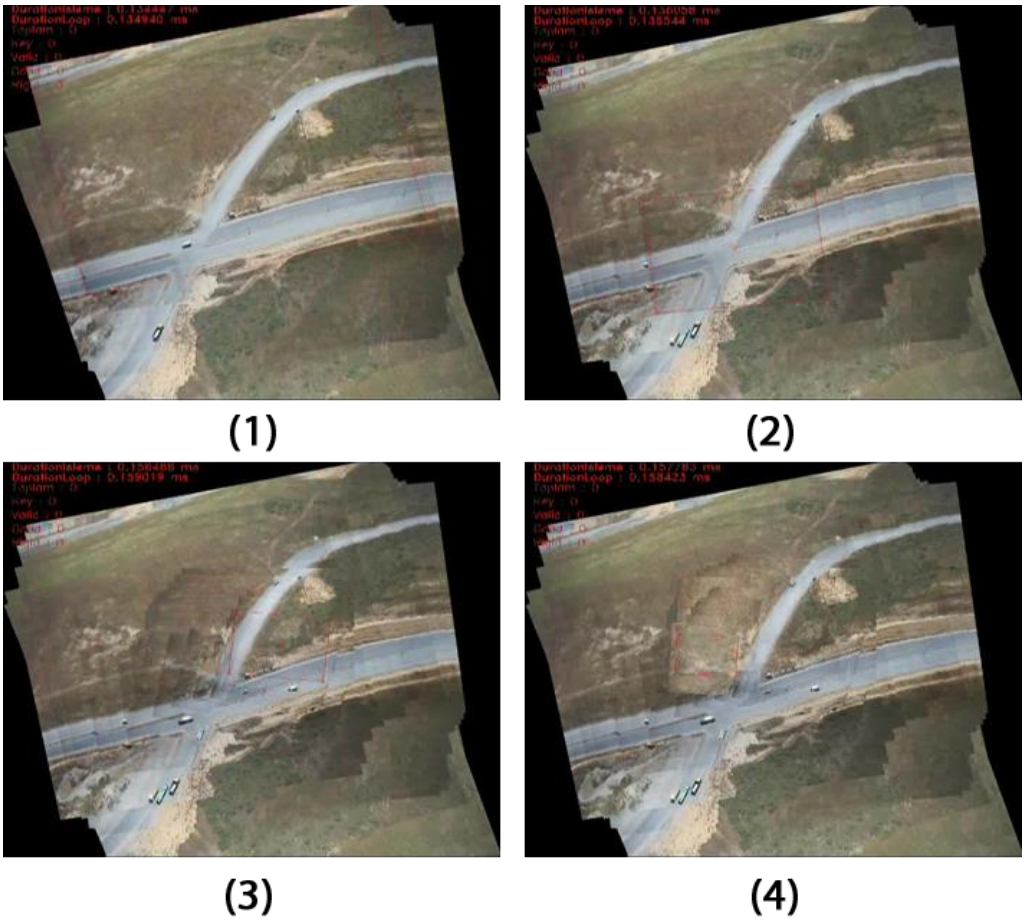


Figure 5.2 4 frames from a 31 second mosaic sequence processed at 135 ms per frame.

5.2.6 Enhancing Zoomed Images

As it was shown in the previous examples, because of the vibrations and fast UAV movements, zooming can make aerial video footage harder to understand. Also if images are not sharpened by using camera controls, resultant image can be degraded completely making it totally useless. These factors put a limit to camera zoom and in general UAV usage in surveillance tasks while actually limitations of the hardware could enable better performance. Aerial image mosaics provides user with a clear advantage in such kind of

situations. Image mosaics provide a natural stabilization effect and turns vibrations into advantage by increasing the field of view by integrating several frames around an area. This way, closed view surveillance of the objects become possible beyond the traditional limits. Figure 4.8 sequence shows such kind of a severe vibration and movement case.

5.3 Algorithm Comparison

In order to evaluate the performance of the presented methods at different conditions various test comparing different registration algorithms and modified algorithm were conducted. Main purpose of these experiments was to measure the performance of the whole system instead of just the registration methods in order to provide testing conditions more like actual operational conditions as much as possible.

Artificial image creation methods that were applied by other authors [13] were employed at the early stages of the development. Later on this approach was abandoned in order to take effects of noise, optics and computation speed into account. For out-door flight tests, built in diagnostic components are utilized in order to make measurements on flights. For in-door tests, high resolution aerial images ranging from 3456 x 5184 pixels to 9624 x 9568 pixels were printed to hard copies ranging from 1 m x 1m to 5 m x 4 m in dimensions such as shown in Figure 4.3. Camera used in mini UAV connected to a PC via capture card and printed copies of aerial photos were imaged. Ground truth information was provided by the hardcopies. Test setup is show at Figure 5.3. A 5 axis 6 by 3 meter CNC normal used in manufacturing of large aircraft molds was employed for path following tests. Camera used in UAV is attached to the head of the CNC as shown in Figure 5.3 (2). Video stream was supplied to an Intel i7 3.40 GHz desktop PC as shown at Figure 5.3 (3). For path following tests, a 600 mm x 1000 mm rectangular translation path and an 822.7 mm long path having translational, rotational and scaling motions were used.

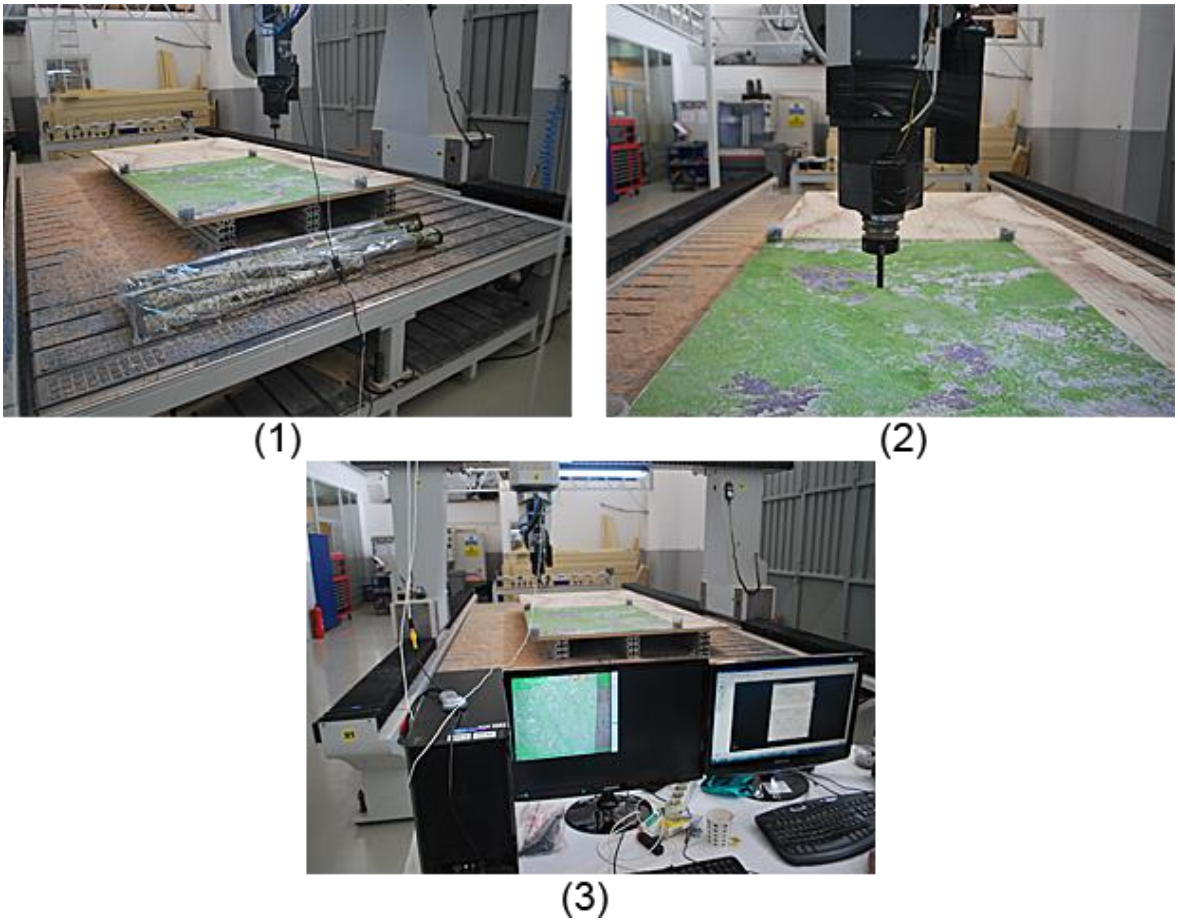


Figure 5.3: Photos of the test set up used in path experiments. (1) shows several of printed aerial images where one of them was prepared for experimentation. (2) shows the camera mounted on the CNC head. (3) shows a wider view of the CNC and computers used in this experiment.

Path following speed was kept constant at 6000 millimeter / minute and a complete path time for translational was 40 seconds and for rotational path was 50 seconds.

By using this approach it is possible to simulate actual camera noise, optical effects, motion blur and environment lightning while still having a controlled test with ground truth

information. Also instead of processing every frame consecutively, video stream is supplied to processing software enabling methods to skip frames if computation rate is less than frame rate as in actual working conditions. This enables effect of computation speed on the quality of registration to be taken into account since slower algorithms were required to register frames further away from each other. It should be noted that exact value of results such as average pixel error e_{avg} is affected by many parameters such as environment illumination, intensity structure of the aerial image that is used etc. and may vary according to testing conditions. On the other hand relative performance of the algorithms was observed to be consistent throughout the experiments so average values obtained from one set of tests can be used for performance evaluation.

First set of experiments were conducted in order to determine effects of modifications done to SURF in terms of speed and accuracy. Table 5.1 shows a comparison between performance of original SURF and Modified Algorithm. It should be noted that processing speed is not based on the single detection or descriptor extraction step but speed of the whole mosaicing algorithm employing respective detector and descriptors for image registration. Tests were conducted by using rural area scenery such as in Figure 2.1 (2). Average pixel error e_{avg} was calculated by averaging difference between mosaic and frame pixels in overlapping area in new frame registration and includes all distortion effects such as camera noise. Average number of outliers n_{out} was also chosen as a form of performance metric. Details and equations of these metrics are described at Section 11. Methods were tested by both applying a 0.75 scaling factor in preprocessing and without any scaling. Camera was subjected to highly accelerated and rapid movements throughout the experiments.

Table 5.1: Performance comparison of SURF and Modified Algorithm

Methods/Scale Factor	**Algorithm Performance**		
	Processing Speed (ms)	e_{avg}	n_{out}
SURF/ Full	146	12.3	0.3
SURF/0.75	87	17.2	1.5
Mod./ Full	75	11.6	1.2
Mod./0.75	44	12.9	2.4

Results showed that accuracy of the Modified Algorithm and SURF algorithm was identical in most of the metrics. On the other hand, Modified Algorithm was approximately two times faster at same scaling factors. If full scale SURF algorithm is compared to 0.75 scales Modified Algorithm, it can be clearly seen that approximately 3.3 times speed increase was achieved without a significant loss in accuracy enabling real-time processing. One surprise of the results was the lower average pixel error of the Modified Algorithm compared to SURF. This may be due to the fact that SURF has ability to produce key points at higher levels of image pyramid than Modified Algorithm which may create additional errors.

In actual fast low altitude flight, images motions observed can be considered as a combination of steady motions, high translational vibrations and rotational movements. By testing algorithm performance under these three classes of motions, performance of the algorithm during unstructured movements of an actual flight can be examined. So in order to investigate the performance of various image registration algorithms with developed system, experiments applying three different motion models were performed. Average processing speeds of mosaicing algorithm employing different registration methods can be seen at Table 5.2. Table 5.3 and Table 5.4 provide a detailed comparison between mosaicing algorithm utilizing respective registration methods. Test are conducted by using 0.5 scaling factor which observed to provide high speed increases without reducing

performance in flight tests. Same setup configuration employing printed aerial photos and camera connected to PC was used. Test are repeated by using several different aerial views similar to the ones shown at Figure 2.1 in order to approximate scenes encountered during actual flight as much as possible. For steady motions camera was subjected to a series of movements combining hold positions and slow translational motions. For high translational motions, 5 Hz vibrations with amplitude of 200 pixels in average are applied. For rotational vibrations, 4 Hz rotation reaching up to 80 degrees was performed. Average pixel error, number of outliers, outlier percentage and number of registration failures are taken as performance metrics. Registration failure was defined as the failing to determine correspondence between two consecutive frames. It should be noted that algorithm utilizes all the filters and controls described earlier in order to decide a registration is successful or not.

Modified algorithm, SURF, SIFT, Harris detector coupled with BRIEF descriptor and FAST detector coupled with BRIEF descriptor are selected for evaluation. Modified algorithm and upright SURF algorithm also employed adaptive threshold method described at Section 10 in order to determine Hessian threshold. SIFT detector threshold was adjusted as 0.068 for rural photos and 0.117 for urban photos in order to set the number of detected points to nearly 100. Harris detectors parameters were adjusted to its default values presented at [17]. FAST threshold for difference to between the central pixel and segment pixel is set to 42 for rural images and 70 for urban images.

Table 5.2: Average processing speed per frame of different registration methods (0.5 Scale Factor).

	Mod	**SURF**	**SIFT**	**Harris Adjusted**	**Harris Normal**	**FAST Threshold:36**	**FAST Threshold:42**
Average processing speed per frame (ms)	25	44	132	19	32	12	10

Table 5.3: Vibration tests using low intensity variation image

	Mod			**SURF**			**SIFT**			**Harris BRIEF**			**Fast BRIEF**		
	e_{avg}	n_{out}	e_{out}	e_{avg}	n_{out}	e_{out}	e_{avg}	n_{out}	e_{out}	e_{avg}	n_{out}	e_{out}	e_{avg}	n_{out}	e_{out}
Low Vibration	7.67	1.1	1	8	0.3	0.3	8.67	0.4	0.4	6.67	0.3	0.3	8.33	0.4	0.4
High Vibration	10	2.3	2.1	11	1.4	1.3	12.7	1.6	1.4	10	1.1	0.9	12	1.2	1.1
Rotational Vibration	14	2.9	2.6	14.7	2.1	1.9	17.7	2.3	2	14.7	4.2	3.5	15.3	5.3	4.7

Results of vibration tests showed that, all of the registration methods perform well under slow movements and translational vibrations. Fastest computation is achieved by FAST detector BRIEF descriptor couple which is followed by Harris detector and BRIEF descriptor configuration. It should be taken into account that Harris BRIEF was utilized with default Harris detector values resulting at a greater number of feature points than 100 which was reducing the speed. For average pixel errors, Harris BRIEF performed slightly better than rest of the configurations which is closely followed by Modified algorithm. Average pixel error and percentage of the outliers were observed to increase at high speed vibration reaching its peak with rotational movements for every algorithm.

Modified algorithm in general performed well as archived to be third in processing speed and second in accuracy. A notable condition is that average pixel errors tend to increase with high intensity variation images while number of outliers decrease. This may be due to the high sensitivity of the high gradient images to the pixel displacements. Overall difference in performance metrics were considered insignificant for UAV surveillance.

Table 5.4: Vibration tests using high intensity variation image

	Mod			**SURF**			**SIFT**			**Harris BRIEF**			**Fast BRIEF**		
	e_{avg}	n_{out}	e_{out}	e_{avg}	n_{out}	e_{out}	e_{avg}	n_{out}	e_{out}	e_{avg}	n_{out}	e_{out}	e_{avg}	n_{out}	e_{out}
Low Vibration	7.3	0.8	0.7	8.2	0.2	0.2	8.4	0.2	0.2	6.4	0.1	0.1	8.4	0.3	0.3
High Vibration	12.6	1.2	1.1	12.7	0.7	0.6	12.8	0.8	0.7	10.1	0.5	0.4	12.8	0.6	0.5
Rotational Vibration	13.3	1.9	1.7	16.7	1.1	1	19.7	1.1	1	13.3	2.2	1.8	15.7	2.9	2.6

Table 5.5: Failure rates at low intensity variation image

	Mod	**SURF**	**SIFT**	**Harris BRIEF**	**Fast BRIEF**
Low Vibration	0	0	0	0	0
High Vibration	0	0	0	0	0
Rotational Vibration	0	0	0	2.7	5.5

Most important result of the vibration tests was the performance of the methods in high speed rotations. All tested methods including Harris BRIEF and FAST BRIEF couples were observed to be able to perform without failure in slow rotations. On the other hand when rotations reach a certain speed, failures in registration started to take place. This is considered because of the descriptor part of the configurations since BRIEF descriptor is not rotationally invariant. [21] Number of failures encountered at one second is shown in Table 5.5 and Table 5.6. FAST BRIEF produced more failures compared to Harris BRIEF. Hessian based detector coupled with local feature based descriptors on the other hand performed without any failure at the same vibration speeds. Results obtained by using low intensity variation images were similar in nature although failure rates were slightly increased.

Results of the translation and rotation-scale paths are shown at Table 5.7 and Table 5.8 respectively. For translation path test two different parameter values for Harris BRIEF and FAST BRIEF were employed in order to adjust the number of detected features. Both Modified algorithm and SURF utilized adaptive algorithm for threshold adjustment while SIFT, Harris BRIEF and FAST BRIEF configurations used fix parameter sets selected for detection of similar number of features. This created fluctuation in detected number of features because of the changing intensity nature of the aerial image throughout the path. SIFT had the slowest computation speed with an average of 132 ms per frame, fluctuating between 114 and 169 depending on the number of features detected. Number of feature detected also changed drastically by FAST and at a lesser degree by Harris, resulting in a significant speed reduction. This effect was attributed to the changing nature of the intensity distribution throughout the path.

Table 5.6: Failure rates at high intensity variation image

	Mod	**Surf**	**SIFT**	**Harris BRIEF**	**Fast BRIEF**
Low Vibration	0	0	0	0	0
High Vibration	0	0	0	0	0
Rotational Vibration	0	0	0	2.1	5.0

Table 5.7: Drifting errors in translation path

	Mod	**SURF**	**SIFT**	**Harris Adjusted**	**Harris Normal**	**FAST Threshold:36**	**FAST Threshold:42**
e_{drift}	132	142	112	281	235	205	219
e_{drift} ***Percentage***	2.6	2.8	2.2	5.6	4.6	4.1	4.3

Results of the path tests showed that modified algorithm performed with lower drifting error compared to original SURF algorithm at a speed increase approximately with a factor of 2. This was consistent with the vibration tests and considered due to the fact that SURF algorithm detects feature at higher levels of the scale pyramid creating additional errors. In terms of drifting error it was only surpassed by SIFT algorithm in translational path tests. SIFT algorithm although performed with relatively higher average error in vibration tests, it had the highest accuracy in translational path tests which was unexpected. On the other hand Harris BRIEF configuration showed the largest results although it had low average pixel error in vibration tests and known for its accuracy. Using a fixed parameter set instead and adaptive approach as in SURF and Modified algorithm may affected the results but on the other hand SIFT algorithm also having a fixed parameter set performed better than adaptive approaches in terms of drifting error at translational path tests. FAST BRIEF configuration performed better than Harris BRIEF but still poorly compared to multi-scale

Hessian based detectors. This situation may be due to the fact that main error source on path tests were not fixed frame to frame registration errors inherit in nature of algorithms but algorithms response to changing intensity distribution and performance at various intensity structures. Vibration tests were conducted by imaging a relatively short range having similar intensity nature. On the other hand intensity nature of the scenery during the path tests was constantly changing which was also observed by the fluctuations in number of detected points. Results of the tests showed that SIFT had robust nature for such kind of conditions while FAST and Harris algorithms were more receptive to changes in intensity distribution nature. Modified algorithm on the other hand performed second best in translational path tests and best in rotational-scale path test by only having a fraction of computational time of SIFT.

When results of the experiments were examined, Modified Algorithm selected for further use in flight tests. Accuracy of the all methods tested were close to each other for vibration tests and was adequate for the purpose of this study so ability to register images with different fast camera movements, response to low and changing intensity gradient nature and speed of computation was considered main criteria of selection. Although Harris BRIEF and FAST BRIEF had faster computational speed, they had higher number of registration failures at excessive rotational movements and higher drifting errors at path tests. SIFT performed better than other algorithms in terms of drifting error in translational path tests but it had the slowest computational speed which was not adequate for real-time processing. Overall Modified algorithm demonstrated the most optimum results in terms of speed, accuracy and robustness and selected as registration method of the developed system. It should be noted that for surveillance missions conducted by the mini UAV tested in this study, low altitude

Table 5.8: Drifting errors in rotation-scale path

	Mod	**SURF**	**SIFT**	**Harris Normal**	**FAST Threshold:42**
e_{drift}	74	81	78	153	137

imaging with zoomed camera view is important and employed methods are required to capture very sharp sudden movements in both rotational and translational degrees of freedom. For other applications such as surveillance with a larger UAV having mechanical gimbal stabilization, navigating with constraint motions like mapping with a forward moving vehicle and a downward pointing camera, faster methods such as FAST BRIEF and Harris BRIEF can be considered. On the other it should be considered that after reaching real-time computational speeds at frame rate further speed increase does not affect the outcome and multi scale blob-like feature detection adds to the robustness of the methods at changing scenery which can be considered as a crucial factor even with such kind of applications.

5.4 Flight Tests

Main performance criteria in flight tests were to measure developed algorithms ability to enhance surveillance in actual operation situations. In order to achieve this, tests under various lighting conditions in different times of the day, at different weather conditions including rain, over both urban and rural areas with both infrared and day light camera were conducted.

Mosaic image at Figure 4.7 (1) was composed of 1375 frames integrated over 55 second and Figure 4.7 (2) was composed of 325 frames integrated over 13 seconds. Although excessive rotational and scaling effects were present in both cases, resultant

mosaics were relatively well formed. Average processing time was 44 ms per frame in both cases.

Figure 4.9 shows six samples of a rotating mosaic sequence produced over 16 seconds with approximately 400 frames when UAV is undergoing rotational and translational movements. Rotation of the background mosaic was clearly visible. Average processing time was approximately 74 ms per frame.

Figure 4.11 shows a stabilization sequence recorded over approximately 2 seconds where at 1.2 second of the footage, camera was subjected to excessive translational and rotational motions. As it is seen from the Figure 4.11 these sudden motions were filtered out while the smoother camera motion was being followed. Frames were processed at 45 ms per frame.

Figure 4.14 shows a situation frequently encountered in surveillance missions where camera was adjusted to its maximum zoom in order to examine an object of interest far away. It should be noted that while actual frame shown in red rectangle was subjected to excessive translational vibrations amplified by zooming, operator received a background preserved fixed looking view. It should be noted that in the first four frames view is having excessive vibrations and at Figure 4.14 (5) and Figure 4.14 (6) hibrid view is panned upward and right following user controlled camera motion. Actual sequence was approximately 1.5 second long processed at 55 ms per frame.

Figure 5.4 (1) shows a mosaic image composed of 420 frames constructed by using a slower version of the developed algorithm with an average processing speed of 142 ms. Figure 5.4 (2) was composed of 213 frames and were processed at an average speed of 47 ms. Both cases demonstrated well-formed mosaic images with minor registration errors noticeable.

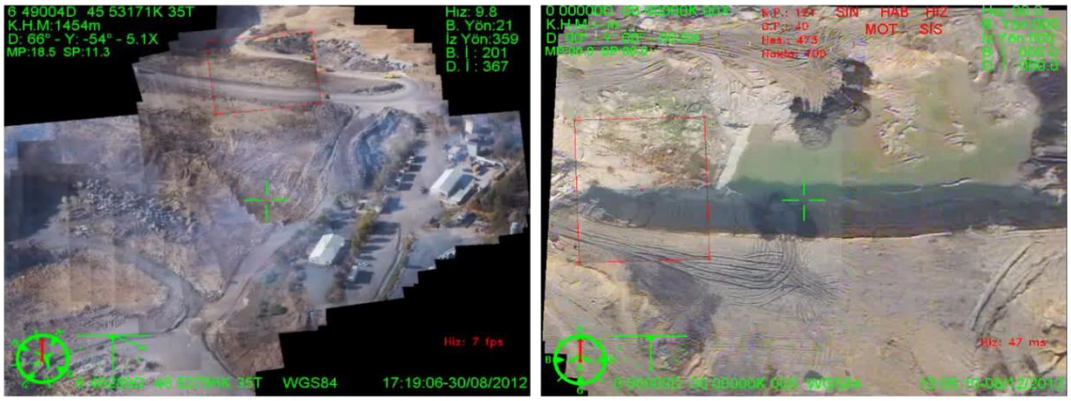


Figure 5.4. Two mosaic maps created in real time with day light camera

Figure 5.5 (1) demonstrates the effects of accumulated error when a previously constructed region is revisited. Mosaic was composed of 463 frames and processed at an average speed of 41 ms per frame. Straight structures such as highways are also good features for examining errors. Figure 5.5 (2) demonstrates such a case where slight registration errors are noticeable through the way. It is composed of 488 frames processed at an average speed of 43 ms.

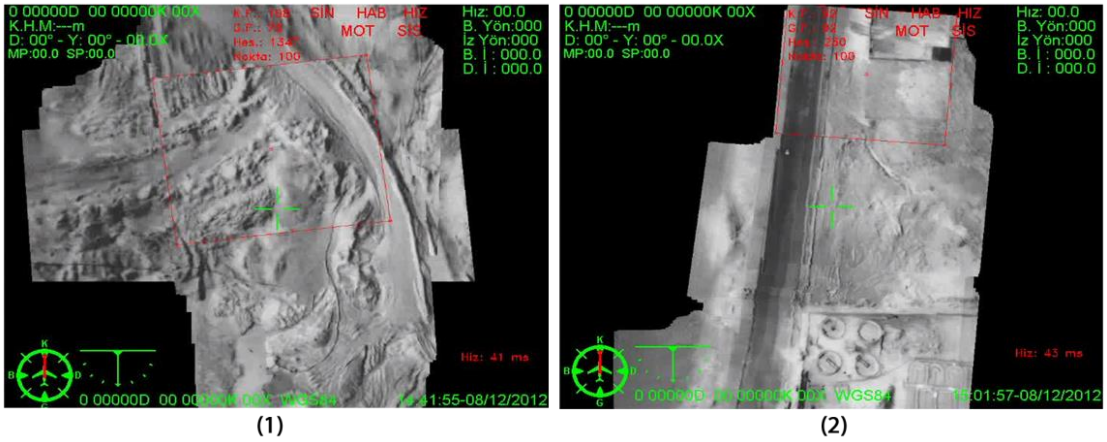


Figure 5.5. Mosaics constructed by using IR camera.

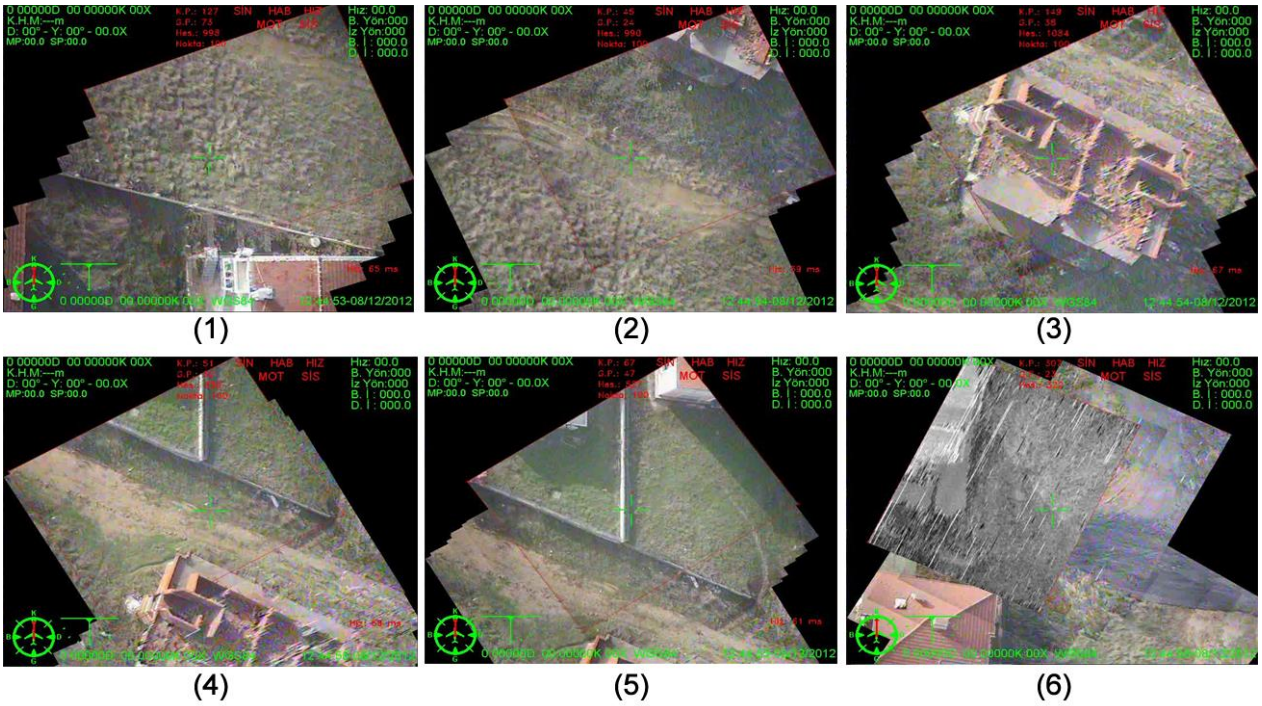


Figure 5.6: Mosaic sequence demonstrating interference errors. Frames are distorted by communication interference from (2) to (4) where effect is more visible at (3). While developed algorithm is robust against distortion in that magnitude, (6) shows where sequence ultimately fails due to severe interference distortions.

5.5 Notable Conditions and Errors

One of the results of erroneous estimation of the transformation matrix is masking effects described by other authors [1]. In this study, masking effects are witnessed especially in low gradient images, excessive optimization conditions for speed, distorted images due to communication interference. In order to increase robustness in such kind of conditions and prevent from one frame corrupting the mosaic image altogether, a structure and limit check on the estimated transform is applied. Structure and the symmetry of the transform is checked and expected to be in the predefined limits of a similarity transform.

Also even under fast UAV movements and sever camera vibrations, translational and rotational motions are observed to be under certain values at most of the cases. These user defined motion limits are also checked and any matrices exceeding these thresholds are considered as an erroneous estimation. This methodology is observed to prevent mosaic distortions under low gradient few feature point images but has an adverse effect when actual image motion is large especially in translation.

One of the primary effects that causing failure of the developed algorithm is

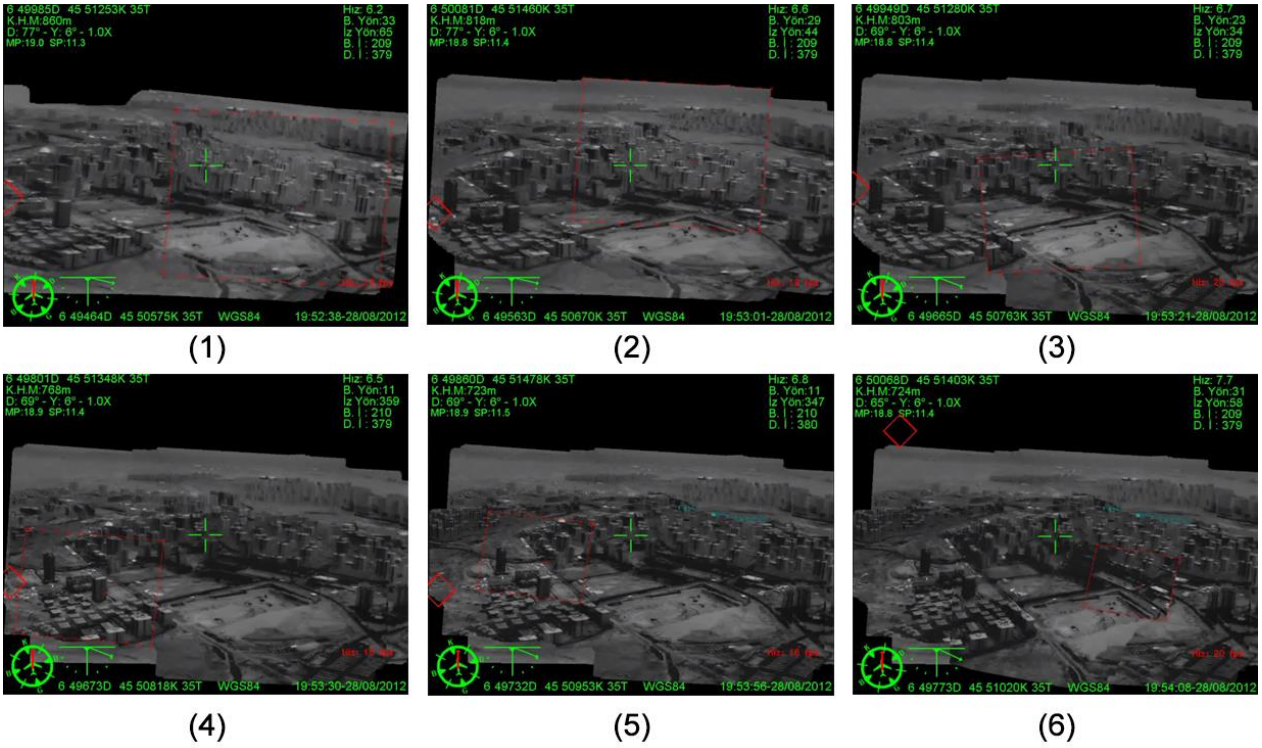
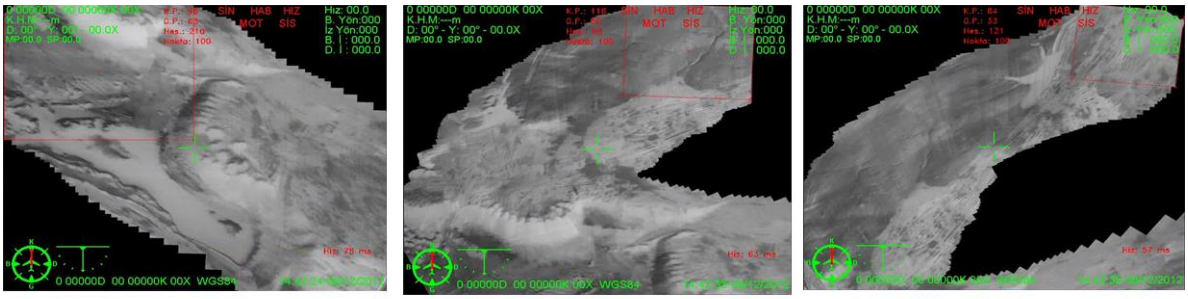


Figure 5.7: Sequence demonstrates effects of moving view point. Images are taken from a night flight where UAV approaches toward a city.

communication interference. Main disadvantage of processing at ground control station instead of on UAV is the communication factors affecting the quality of the frames before

processing. Throughout the flight tests, while most of the time communications did not cause a noticeable drop in performance of the algorithm, excessive distortions such as seen at Figure 5.6 inter, if persisted for a relatively long period, resulted in failure. Figure 5.6 shows a sequence covering 7 seconds recorded in fast low altitude flight. Average processing speed is 62 ms per frame and 116 frames are involved in total. Frames Figure 5.6 (1) to (4) covers 1 second period where received frames are corrupted by slight interference seen at Figure 5.6 (3) for a brief amount of time. It should be noted that although distortion effects of the interference are noticeable, developed algorithm still managed to register the new frame correctly and continued with the sequence at Figure 5.6 (5). This is considered to be because of the filtering of mismatches in refinement step and RANSAC coupled with robust hessian based registration. Six seconds later, at Figure 5.6 (6), system was subjected to excessive communication interference resulting in failure and reset of the sequence.

Another important factor to be noted is the effect of imaging from a moving point. If UAV used for planar mapping by moving at a fixed altitude over a relatively flat terrain with a straight down looking camera, resultant image motions are planar causing no registration error. On the other hand if UAV is moving toward a target, every new frame of the same region is slightly different because of the changing camera position. Figure 5.7 shows a sequence formed by integrating 1748 frames over 92 seconds at an average processing speed of 53 ms per frame. In this sequence an infrared camera equipped UAV approaches toward a city with a forward looking camera position. As it is shown at Figure 5.7 (1) to (6), size of the newly registered frame with respect to total mosaic image decreases since new frames are acquired from an increasingly closer point. Also slight



(1) (2) (3)

Figure 5.8: Sequence demonstrates the scaling effects due to misalignment of the camera.

registration errors when new frame visits a previously covered region in the mosaic due to change in the camera point coupled with accumulated error were present. This effect was more noticeable near the edges of the mosaic where the change in the view was larger. On the other hand, although such effects are present, resultant mosaic still aids to operator in navigation makes it possible to see the position of the currently viewed region with respect to general city view.

Figure 5.8shows an effect that is encountered when camera is not correctly aligned to straight downward looking position during mapping. Sequence shows a rotating mosaic sequence composed of 129 frames integrated over 8 seconds with an average processing speed of 62 ms per frame. During the forward motion of the UAV, newly acquired frames gradually scaled down as it is seen from Figure 5.8 (1) to (3). This phenomena is due the fact that camera was slightly deviated forward from downward looking position and new interest points first enter to scene having less distance than they should have.

One of the notable conditions observed at flight test is the enhanced image motions encountered low altitude flights with excessive camera zoom. Figure 5.9 shows a 10 second sequence integrating 217 frames at an average processing speed of 46 ms per frame. During this sequence UAV was making a low altitude flight at a windy weather in coastal environment and camera was zoomed in order to get closed up images of objects of

interest. Fast movements of UAV resulted in extensive image movements reaching up to 200 pixels between two consecutive images when coupled with the side to side movements due to amplified vibration effects and wind currents. Time between consecutive sample frames in Figure 5.9 is approximately 2 seconds. In Figure 5.9 (1) sequence starts with an upward fast movement continued by a sudden upward jump and low amplitude oscillations revealing the surroundings at Figure 5.9 (2). Later image movements continue with a very sudden sharp downward motion followed by a fast continuing movement two right (Figure 5.9 (3)) ended by a very sharp upward jump (Figure 5.9 (4)). At Figure 5.9 (5) frame continues to oscillate around its current location for a short period of time revealing a larger portion of the area ended with a fast upward motion seen at Figure 5.9 (6). This type of motion behavior is frequently observed in low altitude excessive zoom surveillance missions and fast, sudden image movements makes surveillance impractical. In order to deal with this situation operator may need to zoom out reducing image movement speed but this prevents closed up examination of the objects of interest presenting a limit to surveillance magnification. On the other hand image mosaics provide an adequate solution to problem. It should be noted that in the Figure 5.9, even the small bushes on the terrain is clearly observable. As an example object of interest bushes at the down side of Figure 5.9 (2) is seen on frame for only half a second while stays on mosaic for nearly 5 seconds in original video. One limit of the developed system in low altitude surveillance is if the image movements surpasses motion limit on estimated transform by reduction in altitude, increase in UAV speed or magnification of the image, image registration fails but as it is seen at Figure 5.9 general low altitude surveillance can be done with satisfactory results.

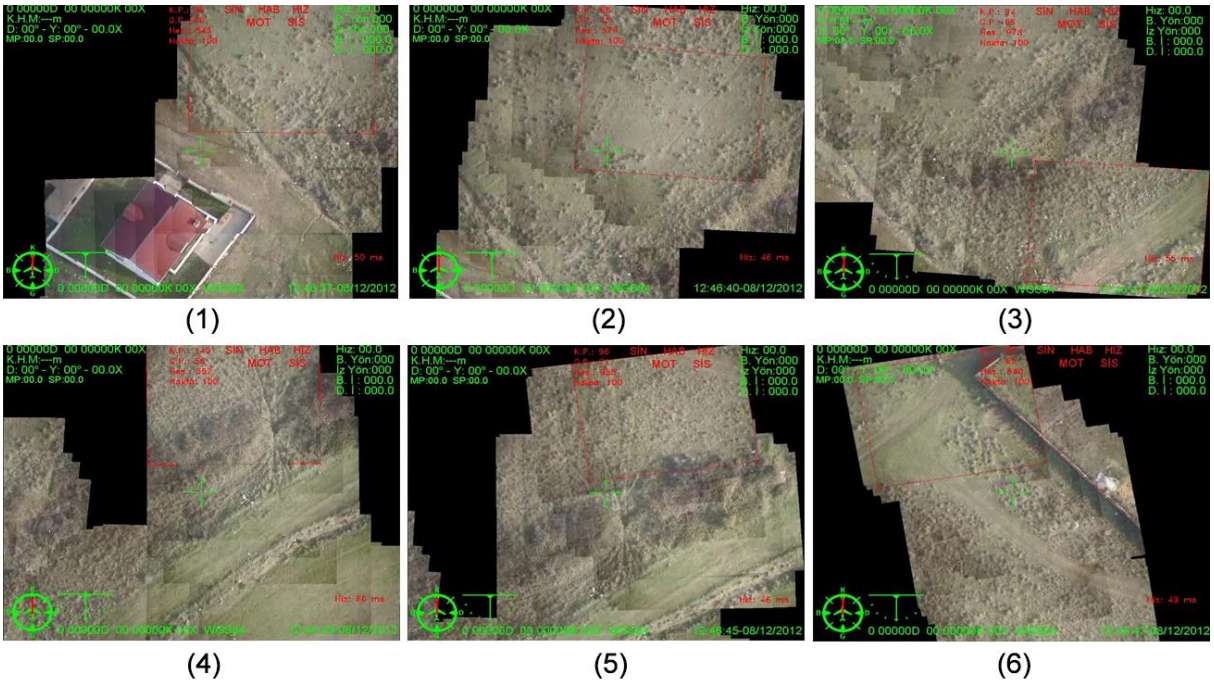


Figure 5.9: Sequence shows severe vibration and motion effects encountered during a low altitude flight. It should be noted that UAV was navigating at a straight path while all motion effects from (1) to (6) are due to unwanted vibrations and sudden winds.

Another important issue, affecting the performance of the developed system is the nature of the intensity distribution of the acquired images. While image with high intensity changes like urban areas perform well with most of the registration algorithms, low gradient images encountered in the rural operational range of the tested UAV presents additional challenges. Figure 5.10 shows the behavior of the developed system in such kind of conditions. Rotating mosaic sequence in Figure 5.10 is composed of 147 frames processed at an average speed of 88 ms per frame over a 13 second period. At Figure 5.10 (1) a total of 68 keypoints are detected reducing to 35 good matches after filtering. Because of the low number of the detected points adaptive algorithm adjusted the hessian threshold to its user defined minimum which is also vulnerable to false keypoint detections. It should

be noted that since intensity distribution in the sea is relatively flat, most of the keypoints are detected at the small land areas included in the frame. At Figure 5.10 (2) since more of the land are included into the frame, number of detected keypoints increased to 80 with 56 good matches after filtering and at Figure 5.10 (3) 134 with 134 good matches. At Figure 5.10 (4) most of the frame includes a low gradient land terrain structure frequently encountered in surveillance missions. Although intensity variation is relatively low compared to urban areas, it provides a more favorable distribution compared to water. Because of the increase in the intensity variations, number of detected keypoint is increased to 270 with 147 good matches. Adaptive algorithm starts to adjust the hessian threshold in order to detect fewer and better quality keypoints. At Figure 5.10 (5) frame mostly encloses the low gradient terrain and hessian threshold is further

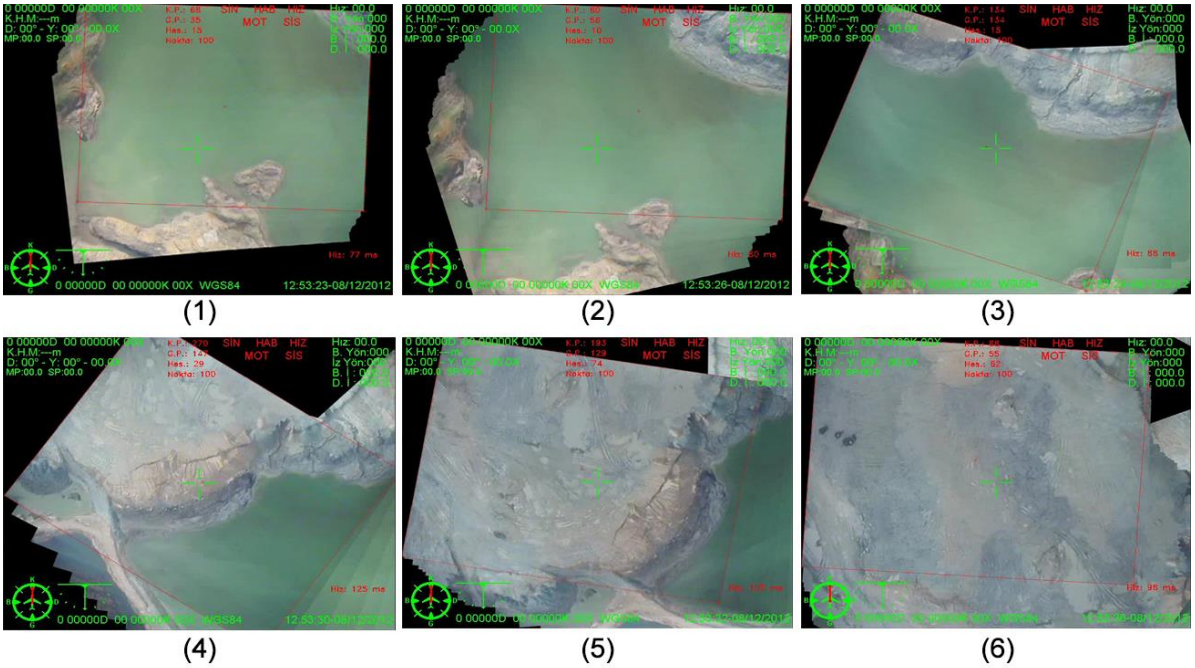


Figure 5.10: Sequence demonstrates the effects of low intensity variations where UAV navigates from low gradient water scenery to low gradient land scenery.

increased. Number of keypoints detected is 193 where good matches are 129. At Figure 5.10 (6) frame fully encloses the low gradient terrain and adaptive algorithm slightly reduced hessian threshold. This is due to the fact that high gradient shoreline is now excluded from the frame. Number of keypoints detected is 88 while good matches are 55. Throughout the sequence, although frame gradients are low and changing repeatedly, adaptive algorithm coupled with hessian based feature detection, developed system performs relatively well without any noticeable errors.

5.6 Infrared Mosaics

Although infrared cameras are widely used in surveillance missions because of their unique advantages, surprisingly, there is a lack of literature for creating aerial mosaics using infrared images. In order to address this issue, several flight tests using infrared camera equipped UAVs were performed.

Early tests with infrared images revealed that Modified Algorithm configured to process day light images showed poor performance for registering infrared images. This was due to the different gradient structure of these images, which provides fewer number of apparent features, resulting in very few number of key point detection most of the time. In order to handle this situation, scaling factors and detector parameters were adjusted to create a more favorable gradient structure, similar to daylight images. Figure 5.11 shows a 400 frame part of a 1275 frame infrared mosaic processed at 47 ms per frame. Sequence is recorded at a night conditions during a UAV flight over urban areas.

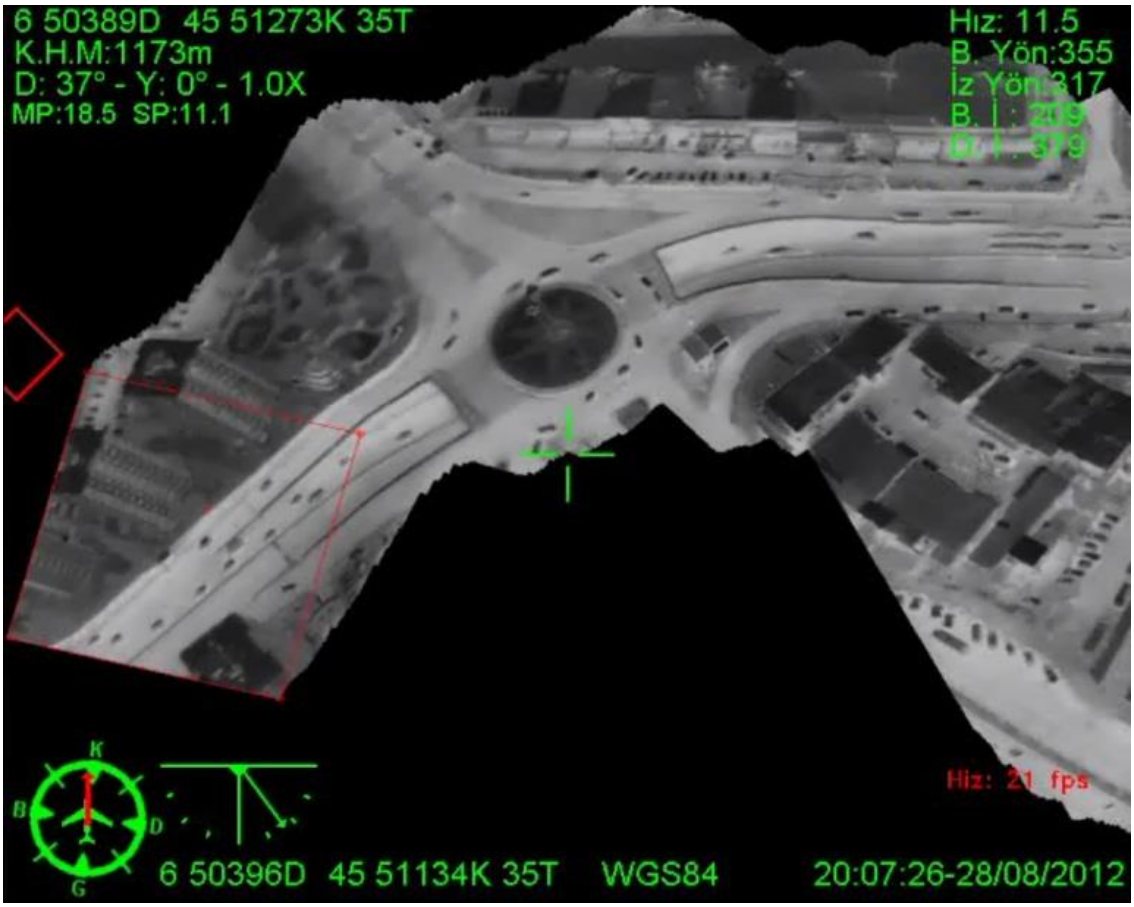


Figure 5.11: An example of infrared mosaics when UAV was flying over an urban area.

Figure 5.12 shows 6 samples of 3 second a stabilization sequence employing infrared images. First three images, Figure 5.12 (1) through (3) show unprocessed frames captured in less than 1

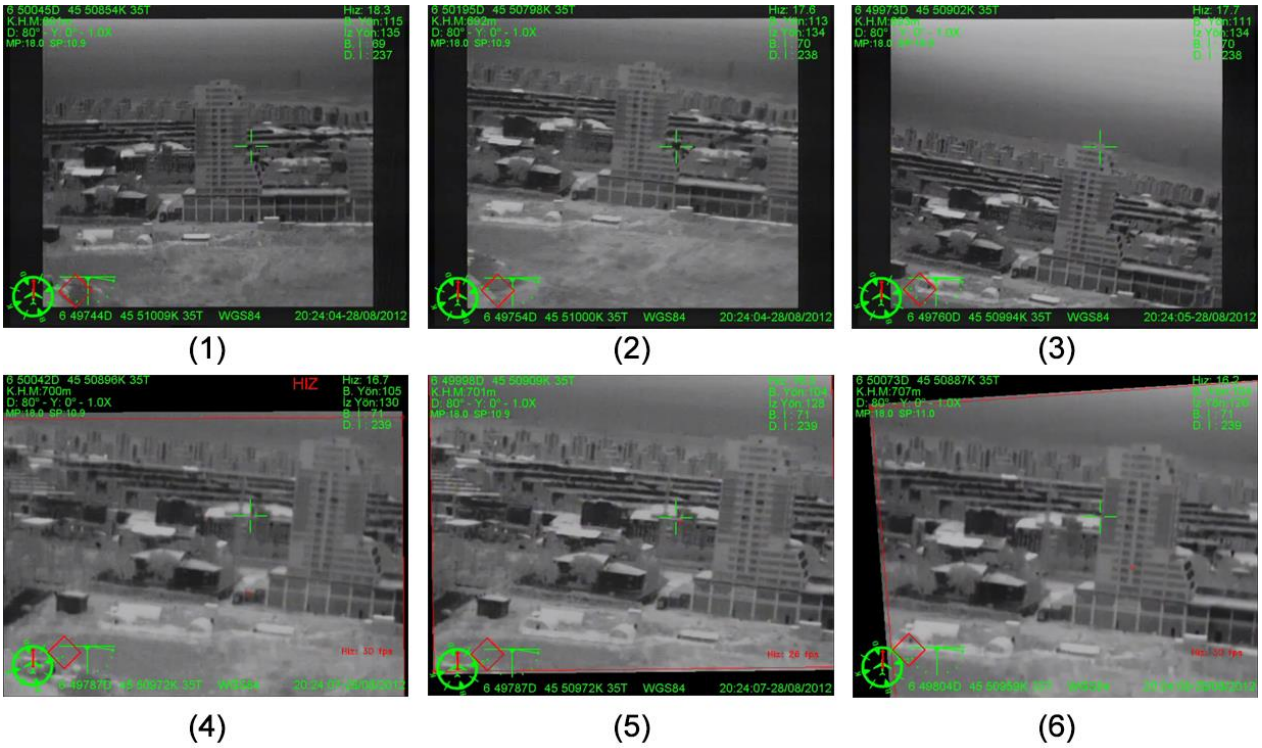


Figure 5.12: Stabilization sequence by using an IR camera. First three frames from (1) to (3) show vibrations at the unprocessed frames. Last three frames from (4) to (6) show the processed frames received by user by using stabilization mode.

second. As it can be noticed from the movements of the skyscraper, video footage is undergoing severe rotational and translational motion because of the windy weather conditions. Last three images, Figure 5.12 (4) through (6) covers a period of less than one second after the stabilization algorithms started. By comparing the two sets it can be seen that developed stabilization is also able to filter out vibrations while following camera movement with infrared images.

An important issue observed in infrared flight tests over rural areas is the performance of the registration algorithms. At Figure 5.8 (1) number keypoints detected are 96 with a good point number of 63. It is comparable to low gradient terrain image at Figure 5.10 (6) which had 88 keypoints and 55 good matches and has a higher hessian threshold. At Figure

5.8 (2) number of detected keypoints are 118 with 56 good matches. Its hessian threshold is lower than the previous sample showing adjustment to lower gradient intensity distribution. At Figure 5.8 (3) number of detected keypoints decreased to 84 with 33 good matches and hessian threshold increased relative to the previous sample. It should be noted that all of the infrared frames are preprocessed before actual feature detection. Results show behavior similar to low gradient images so methods developed with such condition worked well with infrared images.

In order to investigate performance of the developed system utilizing different registration methods with infrared images, tests similar to day light images were conducted. Infrared image were obtained by viewing same urban scenery at the same time period with the IR camera used on the UAVs. Camera was subjected to steady low vibrations motions and moderate vibrations at 4 Hz with amplitude of approximately 50 pixels. Faster pixel movements resulted in prominent motion blur effects making registration fail for every method tested. Result presented in Table 5.9 show lower average pixel errors in general. This may be due to the fact that amount of noise were different between day light and IR cameras and day light cameras were operating in relative low illumination conditions with shutter time decreased in order to retain image sharpness. On the other hand there were a significant increase in the number and percentage of outliers which signifies an increase in number of mismatches. This is considered to be because of the decrease in number of apparent points due to less favorable intensity variations.

Table 5.9: Vibration test using IR camera

	Mod			SURF			SIFT			Harris BRIEF			FAST BRIEF		
	e_{avg}	n_{out}	e_{out}	e_{avg}	n_{out}	e_{out}	e_{avg}	n_{out}	e_{out}	e_{avg}	n_{out}	e_{out}	e_{avg}	n_{out}	e_{out}
Low Vibration	2.6	5.3	4.8	4.7	2.6	2.4	2.3	7.2	6.3	2.4	3.7	3.1	3.6	3.2	2.9
Moderate Vibration	4.6	16.2	14.7	10.3	8.4	7.6	3.4	10.1	8.8	5.5	7.3	6.1	9.1	6.8	6.1

Table 5.10: Failure rates using IR camera

	Mod	SURF	SIFT	Harris BRIEF	FAST BRIEF
Low Vibration	0	0	0	0	2.2
Moderate vibration	0	0	0	2.1	5.4

One of the most notable results of the experiments was the performance of Hessian based multi-scale detectors SURF and SIFT. These detectors performed more robust compared to Harris BRIEF and FAST BRIEF configurations without causing any registration failure. Number of outliers and outlier percentage were also lower than Modified Algorithm. Also in higher speed tests that are not presented here, SURF and SIFT methods produced less number of failures. This situation is considered to be due to the ability to detect more distinctive feature points at higher levels of scale–space pyramid at IR images gradients structure at lack of apparent feature points. On the other hand although

having slightly more number of outliers Modified algorithm performed quite well with a faster computation speed. Also as in daylight images Modified algorithm showed lower average error than SURF working throughout all scale levels.

Second important outcome of the tests were the registration failure rates of the Harris BRIEF and FAST BRIEF methods as shown at Table 5.10. Although outlier rates seem low, these values actually can be misleading because they are calculated at successful registrations. In spite of the stable performance under low and steady motions with faster movements Harris BRIEF began to produce registration failures. On the other hand FAST BRIEF showed registration failures with both low steady motions and faster vibrations later one causing a noticeable increase in failure rates. Increase in the failure rates with faster movements considered to be due to the increased amount of motion blur. Since shutter adjustments were done for day light tests, motion blur effects were minimal resulting in similar results with low rate and high rate translations. On the other hand these effects were more apparent with IR camera reducing the performance of Harris BRIEF and FAST BRIEF combinations while methods detecting feature points in a few scale levels staying relatively robust. FAST BRIEF having a large average pixel error, showed relatively low number of outliers but mosaic sequence was frequently interrupted with registration failures. It can be due to the fact that low information utilized by FAST detector increases number of mismatches with IR images resulting in low outlier and high failure rates. Overall Modified Algorithm is selected to be used with flight tests considering accuracy and speed on the other hand multi-scale Hessian detectors SURF and SIFT present a viable alternative to be used with IR images.

5.6 Conclusion

In this paper, a novel system for hardware independent, real-time stabilization and mosaicing of aerial images acquired by using a mini UAV was described. System designed for an on-service UAV that is actively being used in border patrols and rural operations and was tested successfully on operational conditions with both day light and infrared cameras. Results of the light tests showed that methods presented here provides robust operation under various illumination conditions, different weather including rain and various scenery including difficult low intensity variation scenes. At the time of writing of this paper, pilot operations of software employing methods described this paper began at several locations.

Study presented in this paper is result of a long on-going research in order to develop methods and tools that would enable dependable and practical aerial image stabilization and mosaicing which would be used in actual surveillance operations. In order to achieve this, many factors from acquisition of image to presentation of the final products to operator was thoroughly investigated. Factors affecting the performance such as motion blur and communication interference distortions were examined. Refinement processes in order to increase robustness and different options for estimated transforms for capturing unstructured motion were described. Several state of art feature detectors and descriptors were tested in order to find the most suitable configurations for mini UAV surveillance. Result of the tests showed that Hessian based multi-scale detectors are more robust for motions and scenery encountered in UAV surveillance. SURF detector and descriptor were selected for further examination and appropriate modifications are done in order to meet real-time processing speed requirements.

In order to enhance surveillance, four modes namely, mosaicing mode, rotating mosaics mode, stabilization mode and hybrid mode were developed. Each of the modes provide unique benefits and suitable for particular surveillance scenarios. Flight tests that demonstrating usage of these modes were conducted. Notable issues and sources of error

were thoroughly examined and presented. Understanding of these conditions is crucial in order to develop a robust system in real-world working conditions.

Distorting effects described at the beginning of this paper presents a limit to the effective surveillance that can be done by using mini UAVs. Especially in low altitude flights, zooming in order to have a better examination of objects of interest becomes impractical. This creates a necessity for deployment of larger aircrafts with more sophisticated mechanically stabilized gimbal systems which in turn increases cost and deployment times. By solving these problems in a practical way, methods described in this paper increases the limits on mini UAV surveillance. System described here was designed to be used with the smallest UAVs and requires minimum hardware that is present in most basic UAV systems. Possible developments would be addition of parallel processing and GPUs which in turn creates additional hardware restrictions. On the other hand methods described in this paper are applicable to existing and future UAV system without additional changes in hardware.

In this paper, a novel system for hardware independent, real-time surveillance and remote sensing system utilizing a basic mini UAV configuration was described. In order to measure mosaic quality, in-door and flight tests were conducted. In order to accurately measure effectiveness of state of art algorithms in operating conditions, a novel mosaic quality measurement method composed of 3D positioning and printed high resolution aerial images were developed. Results reveal optimum performance of Modified Algorithm in terms of speed and accuracy and developed system was able to create high quality mosaics at actual flight conditions in real-time.

BIBLIOGRAPHY

[1] B. S. Morse, D. Gerhardt, C. Engh, M. A. Goodrich, N. Rasmussen, D. Thornton, D. Eggett, Application and evaluation of spatiotemporal enhancement of live aerial video using temporally local mosaics, In Proceedings of CVPR, (2008), 1–8.

[2] R. Kumar, H. Sawhney, S. Samarasekera, S. Hsu, H. Tao, Y. Guo, K. Hanna, A. Pope, R. Wildes, D. Hiroven, M. Hansen, P. Burt, Aerial video surveillance and exploitation, Proceedings of the IEEE: Special Issue on Third Generation Surveillance Systems, 89(10) (2001), 1518–1539.

[3] B. S. Morse, D. Gerhardt, J. L. Cooper, M. Quigley, J. A. Adams, C. Humphrey, Supporting wilderness search and rescue using a camera-equipped mini UAV, Journal of Field Robotics - Special Issue on Search and Rescue Robots, 25(1-2) (2008), 89-110

[4] A. Rav-Acha, Y. Pritch, D. Lischinski, S. Peleg, Dynamosaicing: mosaicing of dynamic scenes, IEEE Transactions On Pattern Analysis And Machine Intelligence, 29(10) (2007), 1789-1801.

[5] K. Richmond, Real-Time visual mosaicking and navigation on the seafloor, Stanford University, 2009.

[6] M. Bryson, A. Reid, F. Ramos, S. Sukkarieh, Airborne vision-based mapping and classification of large farmland environments, Journal of Field Robotics - Visual Mapping and Navigation Outdoors, 27(5) (2010), 632-655.

[7] T. Nicosevici, N. Gracias, S. Negahdaripour, R. Garcia, Efficient three-dimensional scene modeling and mosaicing, Journal of Field Robotics - Three-Dimensional Mapping, 26(10) (2009), 759-788.

[8] J. Lai, L. Mejias, J. J. Ford, Airborne vision-based collision-detection system, Journal of Field Robotics, 28(2) (2011), 137-157.

[9] Y. Matsushita, E. Ofek, X. Tang, H.Y. Shum, Full-frame video stabilization, Proceedings of CVPR (1) (2005), 50–57.

[10] M. Irani, P. Anandan, Video indexing based on mosaic representations, Proceedings of the IEEE, 86(5) (1998), 905–921.

[11] M. Irani, P. Anandan, J. Bergen, R. Kumar, S. Hsu, Efficient Representations of Video Sequences and Their Applications, Signal Processing: Image Communication, (1996), 327-351.

[12] Szeliski, Image alignment and stitching: A Tutorial. Foundations and Trends in Computer Graphics and Vision, 2(1) (2006), 1–109.

[13] P. Azzari, L. D. Stefano, S. Mattoccia, An Evaluation Methodology for Image Mosaicing Algorithms, In Proceedings of 10th International Conference of Advance Concepts for Intelligent Vision Systems, Juan Les Pin, (2008), 89–100

[14] P. Paalanen, J. K. Kamarainen, H. Kälviäinen, Image Based Quantitative Mosaic Evaluation with Artificial Video, In Proceedings 16th Scandinavian Conference, Oslo, Norway, (2009), 470-479

[15] L. Zou, J. Chen, J. Zhang, Assessment approach for image mosaicing algorithms, Optical Engineering, 50(11) (2011), 110501-110501-3

[16] T. Tuytelaars, K. Mikolajczyk, Local Invariant Feature Detectors: A Survey, Foundations and Trends in Computer Graphics and Vision, 3(3) (2007), 177–280.

[17] C. Harris, M. Stephens, A combined corner and edge detector, In Proceedings of 4th Alvey Vision Conference, Manchester, UK, (1988), 147–151.

[18] E. Rosten, T. Drummond, Machine learning for high-speed corner detection, In Proceedings of the European Conference on Computer Vision, (2006), 430–443.

[19] D. G. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision, 60(2) (2004), 91–110.

[20] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, Speeded-Up Robust Features (SURF), Computer Vision and Image Understanding, 110 (2008), 346–359.

[21] M. Calonder, V. Lepetit, P. Fua, BRIEF: Binary Robust Independent Elementary Features. In Proceedings of the 11th European conference on Computer vision, 4 (2010), 778-792.

[22] D. A. Forsyth, J. Ponce, Computer Vision a Modern Approach, Prentice Hall, 2003

[23] J. Shi and C. Tomasi, "Good features to track," in IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'94), Seattle, 1994, 593–600.

[24] C. Schmid, R. Mohr, C. Bauckhage, Evaluation of interest point detectors, International Journal of Computer Vision, 37(2) (2000), 151–172.

[25] K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors, IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(10) (2005), 1615–1630.

[26] R. Hartley, A. Zisserman, Multiple View Geometry in Computer Vision, Cambridge University Press, (2000),

[27] M. A. Fischler, R. C. Bolles, Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, Commun. ACM, 24(6) (1981), 381–395.

VITA

Tolga Büyükyazı graduated from Özel Seymen Science High School in Kocaeli at first rank in school and admitted to Boğaziçi University Mechanical Engineering department at first rank at 1999. Upon graduation at 2004, he worked in firefighting and car security sectors for one year and started his graduate study at Koç University Mechanical Engineering department at 2005. Because of various reasons he quitted his studies and started his military service as purchase and inspection officer. After completion of his military service he joined Baykar Makina at 2011 as Research Engineer. Currently he leads a small team of engineers on object oriented programing based R&D projects ranging from UAV operator interfaces, mapping applications, graphical navigation displays, various navigation and telemetry devices, video augmentation and computer vision applications.