Reconstruction of 3D Environments from UAV's Aerial Video Feeds

D. R. P. P. Hettiarachchi
2021

Reconstruction of 3D Environments from UAV's Aerial Video Feeds

A dissertation submitted for the Degree of Master of Computer Science

D. R. P. P. Hettiarachchi
University of Colombo School of Computing
2021

DECLARATION

I hereby declare that the thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis. This thesis has also not been submitted for any degree in any university previously.

Student Name: D. R. P. P. Hettiarachchi

Registration Number: 2018/mcs/030

Index Number: 18440301

 28/11/2021

Signature of the Student & Date

This is to certify that this thesis is based on the work of Mr. /Ms. _____ under my supervision. The thesis has been prepared according to the format stipulated and is of acceptable standard.

Certified by,

Supervisor Name:

Signature of the Supervisor & Date 29th Nov 2021

ACKNOWLEDGEMENTS

This thesis is a result of the support and assistance of several different people in various ways. First and foremost I would like to express my sincere gratitude to my supervisor, Prof. Prasad Wimalarthne for the invaluable guidance, supportiveness, and comments he had provided me throughout this study. I sincerely thank all the lecturers at the University of Colombo School of Computing for being extremely supportive. Finally, I would like to thank all the people who motivate and support me to make this project a success.

ABSTRACT

Graphical 3D models of a real-world scene could contribute to the knowledge, better understanding, and further investigations for a variety of scenarios such as accident sites, historical sites, construction progress monitoring, etc. This study proposed a proof of concept prototype for low-cost offline 3D reconstruction using 2D video frames obtained by a UAV using the method of Structure from Motion (SfM). The work presented consists of novel approaches for video frame selection and feature matching in the reconstruction pipeline to improve the accuracy, robustness, and efficiency which are also considered as major challenges in 3D reconstruction. The introduced video frame selection stage to extract frames is based on the affine transformation between views and it is to obtain an optimal set of video frames to be processed with the required amount of feature points while maintaining the performance. Introduced feature matching stage with improved and a novel algorithm is presented with the underline assumption of spatial cohesion which is the consecutive video frames have higher matchings between them. The algorithm is designed to perform feature matchings between consecutive frames to avoid unnecessary processing thus improving the performance and final output. The feature matching stage consists of multiple filtering steps to remove noise and outliers among matched features to obtain robust matches. The evaluation results show that the improved feature matching algorithm has a significant performance improvement in terms of time and memory usage as well as the increased final output model accuracy. The evaluation results of Hausdorff distance comparisons between ground truth models and the output 3D models of the implemented prototype and between the output of existing 3D reconstruction tools show that the implemented prototype outperforms the existing tools.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

FAST	Features from Accelerated Segment Test
FPS	Frames per Second
BA	Bundle adjustment
GPS	Global Positioning System
GSD	Ground Sampling Distance
KLT	Kanade-Lucas-Tomasi
Lidar	Light Detection and Ranging
MVS	Multi-View Stereo
PnP	Perspective-n-Point
RANSAC	Random Sample Consensus
SfM	Structure from Motion
SIFT	Scale-Invariant Feature Transform
SLAM	Simultaneous Localization and Mapping
UAV	Unmanned Aerial Vehicle

CHAPTER 1
INTRODUCTION

Reconstruction of environments such as buildings and landscapes into 3D graphical representations using video frames or sequence of images is an interesting topic of research in computer vision and photogrammetry (Chen et al., 2018). With computer vision advancements it allows to obtain 3D information of the scene without the knowledge of different important information (Pepe and Costantino, 2020) such as camera parameters, 3D point locations, etc. By only using 2D images will lose the geometry and other valuable information of real-world objects (Zheng, 2016). For some applications, 3D information is the key to success. For example, augmented/virtual reality, robots and autonomous car navigation, image-based rendering, and image enhancement are some application areas. Moreover, 3D information can utilize to improve many computer vision tasks as well. Such as object classification, recognition, and human pose estimation. An accurate and realistic reconstructed 3D model with preserved structural aspects enables detailed analysis of the reconstructed models or environment (Pepe and Costantino, 2020). Therefore, there is a strong requirement and importance (Zheng, 2016) to recover reliable 3D information from 2D photos and videos.

Therefore, one of the most important parts in the reconstruction of a 3D environment is the capturing of required 2D photos or videos of that geo-specific area. UAVs will simplify the process of capturing those required images/videos since they can easily set the trajectory of the vehicle (D. Lapandic et al., 2017) since they consist of an autopilot system, navigation and orientation systems, various sensors, and the data links for the communication with the ground station (Javadnejad, 2018). Furthermore, UAV aircraft can be controlled by using a radio controller manually, from a ground control system, or by autopilot via the predefined flight path. These technology improvements and current ongoing developments of small unmanned aircraft systems allow consistent and convenient acquisition of high-quality aerial images at a low cost.

1.1 Motivation

Small UAVs with attached consumer-grade cameras are widely used to generate high-resolution geospatial imagery. Low cost, widespread availability and ease of maneuvering as well as the rapid development of technology allow UAV-based photogrammetry (aerial imagery) to be used widely in a range of applications

Disaster response and disaster management is one such example. UAVs are becoming first responders during a time of natural or man-made disasters when there is a high risk associated with sending first responders as humans for analysis of the environment. Easy maneuvering of the UAV in a such harsh environment or a critical scenario is an added advantage (Pepe and Costantino, 2020). UAV data can be used to reconstruct the accident sites. Another use case is inspection and surveillance from media and law enforcement. UAV data can be useful for civil engineering applications such as construction, structural and maintenance inspection by visualization with 3D mapping. All of these use cases required some analysis of a specific geographic location. If it is possible to obtain a 3D scene of such an environment, it can be a vast advantage as further analysis can be carried out and it may also reveal hidden details in the environment.

Even though different techniques in the field of computer vision are available to be used to reconstruct 3D scenes from inexpensive, consumer-grade cameras, significant research questions remain (Javadnejad, 2018) regarding the accuracy of UAV-based 3D scene generation.

1.2 Statement of the Problem

3D reconstruction using a sequence of 2D images is a low-cost approach compared to more expensive reconstruction approaches based on Lidar sensors and depth sensors (Pepe and Costantino, 2020; Xiao et al., 2020). With the technology improvement, UAVs are capable of capturing high-quality aerial videos. Capturing process can be simplified to a greater extent with its ability to fly a UAV in a predefined flight path. Therefore, it is essential to evaluate the 3D reconstruction using UAV video feeds.

This study will focus on the recovery of camera position and orientation information from video feed frames received from a UAV and using those data to study the process of reconstructing 3D scenes.

This thesis will further address the problems of dense 3D reconstruction with video streams captured from a UAV, which is offline 3D reconstruction.

1.3 Novelty

This work will contribute significantly to advancing the techniques for the problems of offline static scene reconstruction. The main objective is to implement a proof of concept prototype that can use UAV video footage to construct a 3D environment.

The outcome of the 3D reconstruction is heavily dependent on feature matching (Kumar, 2018). For accurate feature matching, and to get correct orientation and position values, focuses on implementing an improved noise removal algorithm to obtain robust matches and as a major improvement to improve the accuracy as well as the performance, implementation of the feature matching algorithm will be implemented by assuming the spatial cohesion which is the consecutive video frames have higher matches to avoid performing matches against each frame as in previous other studies (Chen et al., 2018; J. Hlubik et al., 2018; Yuan et al., 2018) hence to avoid unnecessary processing.

A video file consists of a large number of frames. There should be selected a set of frames to incorporate in the reconstruction process. This work will introduce a novel and simple approach to video frame selection by measuring the transformation between frames based on a certain threshold value instead of selecting frames at specific intervals (D. Lapandic et al., 2017; J. Ke et al., 2020) as with the varying FPS rate of the camera can lead to other problems such as insufficient overlapping features between frames or wastage of computation resources.

1.4 Aims and Objectives

The main goal is to implement a proof of concept prototype to perform automatic detailed offline 3D models reconstruction from UAV video feeds received of an environment. For this study, an environment such as a building with its surrounding will be selected.

To achieve the goals specified in the above sections, there need to be some objectives to be met which can be further specified as below,

- A critical review of different methods and architectures available (literature) with regards to the problem domain.
- Identification and analysis of algorithms exist for the 3D reconstruction from video frames.
- Identify and set up a simulation environment for the purpose of development and evaluation.
- Study the feasibility of developing a highly parallel and memory-efficient algorithm since it has to process a large amount of data (in a large environment).
- Implementation of a proof of concept prototype solution for the automatic 3D reconstruction.

1.5 Scope

This work is focusing on computer vision and image processing-related methodologies to fulfill that demand of creating a 3D environment from multiple video feeds received from a UAV. Those methods will be used to process frames of videos, identifying correspondences, and to generate a point cloud of the 3D scene offline. Development and testing will be carried out in a simulated environment such as AirSim. Which may also use to gather video feed data for the developer environment.

This study is focusing on implementing a proof of concept prototype that runs on a ground computer in which UAV videos can be streamed via wifi, and using those video feeds to reconstruct detailed 3D models automatically and efficiently in the offline mode, of an environment. For this study environment such as a building with its surrounding will be selected.

The techniques which will be involved in the 2D to 3D conversion are feature extracting and tracking, feature matching, three-dimensional geometry estimation, and refinement, and scene structure reconstruction. This study will mainly use the Structure from Motion (SfM) algorithm for the reconstruction process and algorithms such as The Random Sample Consensus (RANSAC) will be implemented to remove the outliers (Li, 2010) in the correspondences. Relative depth information for feature points will be estimated using the multiple views of the scene. Different data sets with their ground truth models will be used to evaluate the 2D to the 3D conversion process, and the analysis of the experimental results will be presented. Visual feature point data will be used to estimate the vehicle and camera pose for each video frame. Then using that camera poses and stereo vision to recover dense depth measurements of the surfaces visible in the video.

This work is assumed cameras have been pre-calibrated and that the objects in the environment to reconstructs are rigid and stationary.

1.6 Structure of the Thesis

The dissertation consists of different chapters with specific details including charts and diagrams to give an overview of the study. Chapter one has presented a detailed introduction of the study including, the problem, objectives, and scope.

Chapter two reviews the background and the existing literature related to this work including current knowledge, methods, and limitations relevant to the work. Review on background

concepts required for the study including structure from motion pipeline, camera projection, feature detection and matching, and triangulation will be discussed.

The third chapter discusses the methodology of the study on the architecture design and steps in the proposed work in detail. This chapter will explain how the UAV video data feeds are used to perform the reconstruction process.

The fourth chapter of the document will present results and evaluate the accuracy gained by this approach.

The final chapter will conclude the dissertation with final comments, findings, and thoughts about the study and future improvements will be presented.

CHAPTER 2
LITERATURE REVIEW

In traditional 3D modeling, it is manually building a 3D model of the objects or area of interest in the environment by manipulating a 3D modeling tool such as Maya or Autodesk 3ds Max. This procedure is widely used in 3D animation and some other fields due to its controllability, accuracy, and complete texture information (Yuan et al., 2018). The drawback is, it's a time-consuming and expensive process. When it comes to modeling scenes in a real-world environment, hidden details (Chen et al., 2018) may not be revealed by this traditional modeling technique.

Even though there are different expensive 3D model reconstruction techniques available using Lidar sensors and depth sensors (Daftry et al., 2015), 3D model reconstruction using a 2D camera is cost-effective (D. Lapandic et al., 2017) solution.

It creates a demand to create ground models of the environment to offer 3D visualizations (Pollefeys et al., 2008) of cities. However, the reconstruction of the 3D model for a large-scale environment accurately using a sequence of images/videos is a challenging task (D. Lapandic et al., 2017). These generated 3D models provide measurements that can be used for a variety of applications including city planning, disaster response, robot navigation (Gallup, 2011), etc.

Some of the major challenges and considerations are

- Generation of complex and large environments can be time-consuming and expensive (Gallup, 2011) since it requires capturing video footage that covers the entire area.
- The algorithm for reconstruction must be optimized to process large amounts of data in a minimal time constraint (Gallup, 2011), with the utilization of high-performance computers and graphic cards (Gallup, 2011; Schöning and Heidemann, 2015).
- Memory usage is also a concern (Chen et al., 2018). To process a large dataset that cannot fit into memory, the algorithm should exhibit locality (Gallup, 2011) (process parts independently). The algorithm needs to be parallelizable and scalable.
- Storage of final generated 3D models in a way they can easily access (Gallup, 2011).

The initial stage to reconstruct a 3D environment is the capturing of required and sufficient videos or images of that specific area with the completeness/coverage of the scene (Daftry et al., 2015). This can be simplified and achieved using UAVs compare to other methods.

Flightpath planning is a critical component for the acquisition of geodata (Gallaway, 2018). For a safe and productive data gathering, it needs to consider different parameters in the UAV such as height, speed, waypoint and pathing information. Apart from those parameters, weather can be affected as 18 knots wind can cause horizontal and vertical deviations of the UAV of 10m and 5m respectively (Gallaway, 2018). For a better 3D reconstruction outcome, it is important to use a flight system with a programmed/automated flight path consist of waypoints to follow using some coordinate system. Manual control for the UAV is also possible and for some scenarios, it is essential to use manual control such as flying close to or underneath trees or buildings.

2.1 Camera Model

Before the conversion process, it is important to understand its inverse in which the camera projects the information of a 3D world into a 2D image. The simplest way is the process in the pinhole camera projection model (Martell, 2017). This model does not include any lenses or any distortions (Kumar, 2018) it model maps any point in 3D space to a point in the image projection plane by the means of straight lines that connect through a fixed point in space which is the center of projection or camera center (C). This is known as the central projection camera model as illustrated in figure 2.1

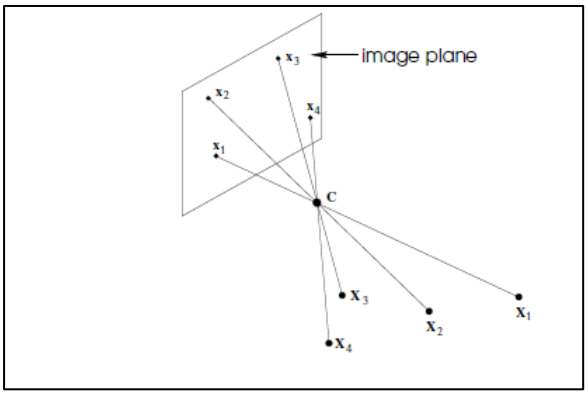

Figure 2.1: Central projection camera model

The focal length of the camera becomes the distance between the aperture and the image plane. A virtual image plane can be assumed to be formed in front of the camera at a distance equivalent to focal length. The mathematical relation (Kumar, 2018) between the real world (3D) points in the world coordinates and the 2D points in the image frame can be expressed as

$$w[x\ y\ 1] = [X\ Y\ Z\ 1]P \tag{2.1}$$

In the above equation (2.1, X, Y and Z represent the coordinates of the point in the world frame and x, y represents the coordinates in the 2D image frame. w is the scaling factor for the image frame and P is the 4x3 camera matrix which represents camera intrinsic and extrinsic parameters according to the relation (equation (2.2)),

$$P = \binom{R}{t} K \tag{2.2}$$

R and *t* in equation 2.2 represent the camera extrinsic, rotation and translation relative to the world origin in the world frame and K is the intrinsic matrix (equation (2.3)) which also known as the camera calibration matrix (Li, 2010) and is given by,

$$K = \begin{bmatrix} f_x & 0 & 0 \\ s & f_y & 0 \\ c_x & c_y & 1 \end{bmatrix} \tag{2.3}$$

Where "s" is the camera sensor skew, f_x and f_y are the focal lengths in the x and y directions in pixels and c_x and c_y are the camera optical centers in pixels.

The extrinsic matrices transform the world points into the camera coordinates and the intrinsic matrix transforms the camera coordinate points into the 2D image coordinates.

Apart from camera intrinsic and extrinsic matrices, lens distortion should be considered as well. These distortions can be either, radial or tangential.

2.2 3D Reconstruction

3D reconstruction on multiple 2D image views is comprised of a wide variety of techniques and algorithms (Chen et al., 2018). As techniques, detection, extraction, and matching of image features from multi-views, calibration of camera matrix, sparse point cloud generation, dense point cloud generation, surface reconstruction, and texture mapping are some. The main algorithms are Structure from Motion (SfM) (Martell, 2017), Poisson Surface Reconstruction, and Multi-View Stereo. Geometrical camera calibration has major importance (J. Hlubik et al., 2018) for the 2D image-based reconstruction process. Two major steps (J. Ke et al., 2020) of reconstruction are 3D point estimation and 3D surface rendering.

2.2.1 Structure from Motion (SfM)

Structure from Motion (SfM) can be considered as a robust approach for 3D reconstruction in the field of computer vision and photogrammetry. SfM algorithm is consists of steps, which

are, feature detection, feature matching and extraction, and incremental reconstruction (Chen et al., 2018). SfM is a well-defined approach to solve the problem of finding the camera motion (translation and rotation) and identifying the geometries (sparse point cloud) of the scene in an automatic and simultaneous manner (J. Hlubik et al., 2018; Pepe and Costantino, 2020)

Feature detection: Responsible to search and identify interesting points to differentiate each frame from every other frame. These identified feature points with the surrounding information get stored and represented in a feature descriptor. Feature descriptors are depended on different factors such as rotation, scale, illumination, contrast, etc.

Feature matching and extraction: It will use an approximate nearest neighbour algorithm to calculate the distance (in high dimension space) to identify corresponding descriptors between frames. This process can be accelerated by using a preemptive feature matching algorithm such as SIFT image feature detection algorithm (Martell, 2017). this preemptive feature matcher works by sorting the features in frames by decreasing scale order, Then it considers a certain threshold for matching features in image pairs.

In order to work with an unordered collection of data, previous studies (Chen et al., 2018; J. Hlubik et al., 2018; Yuan et al., 2018), as well as different tools, were implemented to execute the feature matching stage by performing feature matches against each frame or image with each other frames/images. Therefore, feature matching and extraction is a time-consuming process in the reconstruction pipeline (Martell, 2017).

Incremental reconstruction (figure 2.3): Incremental reconstruction will be initiated by estimating the relative pose of a good image pair and features visible in both images will be triangulated. This step follows by adding suitable next views incrementally to the reconstruction. This process continues until all reconstructable views are part of the scene. Camera distortion parameters can be estimated during reconstruction. The incremental SfM algorithm process is as in the following figure 2.2

Some of the common limitations and considerations of such an approach are sparse output, simple and static scenes can give better results, requiring a controlled and well-planned data acquisition, the baseline should not loo large, non-planar objects can be harder, required accurate camera calibration, geometric consistency (J. Hlubik et al., 2018)

Structure from Motion will result in a sparse point cloud and using Multi-View Stereo (MVS) algorithm enables to densify that sparse point cloud given identified camera motions (J. Hlubik et al., 2018). Using the high-quality UAV images and combining the SfM and MVS algorithms

allows the reconstruction of 3D environments at a low cost compared to laser scanners (Pepe and Costantino, 2020) and allow for the identification of different complex structures in the scene.

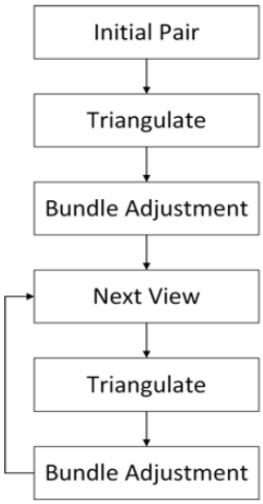

Figure 2.3: Incremental SfM process

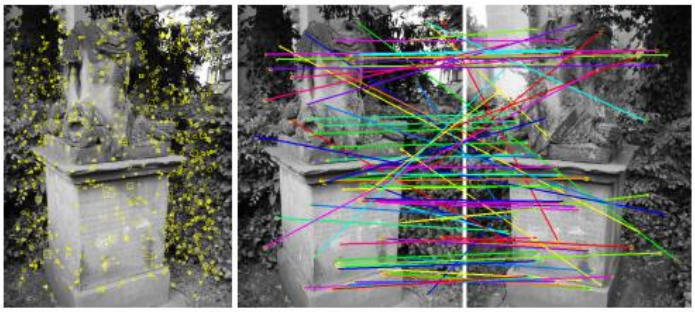

Figure 2.2: Identified features (left) and matches feature between two frames (right)

2.2.2 Triangulation

In SfM, camera poses are estimated based on matching two views and their geometries. After obtaining these poses, the triangulation is performed using cameras and 3D points (D. Lapandic et al., 2017). Triangulation allows estimating unknown points in space by applying projective geometry using the fixed known positions of two points that are in the known distance apart. Camera pose and camera intrinsic matrix can be used to express the camera projective matrix.

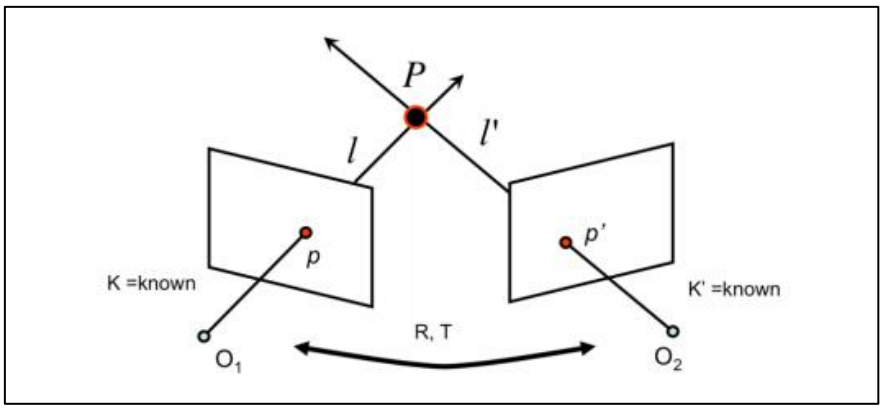

Figure 2.4:The setup of the triangulation problem when given two views

Figure 2.4 illustrates the triangulation problem given two views. Unknown point P in 3D space, known camera intrinsic parameters K and K' respectively, known relative orientations and offsets R, T of these cameras with regards to each other. camera centers O1, O2 and the image locations p, p'

Even though it is theoretically feasible to estimate and recover coordinate positions of 3D points by using only coordinates of two views or observations, correct matches (corresponding points) between the two image points is a significant factor for accuracy. Therefore, it generally involves an error (Ling, 2013). Usually, overestimations are solved by using a set of points. Some algorithms further proposed to minimize the sum of squared errors (Ling, 2013) by considering the 3D positions of points between the measured and predicted values for all the views.

2.2.3 Bundle Adjustment

Bundle adjustment (BA) is a technique for simultaneously optimizing the camera pose as well as 3D point locations for visual reconstruction (Kumar, 2018). It achieves that by minimizing the re-projection errors between observed and predicted locations of image feature points. The result of BA is a refined more fitted 3D reconstruction. BA gets applied as individual or batches of image frames are getting introduces to the initial seed (Gallaway, 2018). That is, the bundle adjustment is a process of global optimization that executes recursively for all the views. The basic algorithm behind bundle adjustment implementation is the Levenberg-Marquardt algorithm. That algorithm is a combination of the Gauss-Newton algorithm with the gradient descent method (Ling, 2013) with the purpose of solving all the correspondence points non-linear criteria.

2.3 Other Related Work

A system was developed (Chen et al., 2018) to run on windows using the techniques and algorithms which are Structure from Motion, Multi-View Stereo, and Poisson Surface Reconstruction. It takes images of an object in multiple different views as the inputs and generates the 3D representation of that object. It was designed to produce a sparse point cloud, densified point cloud, polygonal mesh, and 3D model of the object throughout the process. The input images have to be uniformly scaled with the same resolution and for better results, images should have to be uniformly distributed around the target object.

The study that was done by Chen et al. (2018) was able to provide acceptable results for a small object, but when it used to generate and reconstruct a street view from images captured by a UAV, they have experienced some holes in the dense point cloud. It was due to they were unable to capture images with large overlapping features and another major drawback they have faced is a practical limitation of memory (Chen et al., 2018; Gallup, 2011) consumption issues in the system.

The final analysis of that study (Chen et al., 2018) shows that it's not only important to capture high resolution and quantity images, but also the quality of the images and the captured angles of the camera will also vastly affect the quality of the final 3D models. For compact objects, densely sampled images in a spiral path with many overlaps within images will lead to the optimal outcome. However, it may not practical for large open environments. They have experienced, parts that are not connected in SfM and point cloud holes in the multi-view stereo when using sparse sampling.

D. Lapandic et al. (2017) proposed a six-stage framework for 3D model reconstruction in near real-time. As stages, it consists of, image acquisition, detection and extraction of feature points, feature matching and identifying correspondence points, filtering of the point cloud, estimating camera poses, triangulation, and calculate points and generate a point cloud.

In the Features Detection and Extraction stage, detection of points of interest known as features is used to locate, identify, and categorize objects in image frames. The main characteristic of point detection algorithms is the possibility to detect the same point of interest in multiple views. A set of feature vectors (descriptors) are the output from the feature extraction. To achieve a rapid 3D reconstruction, the algorithm must be fast enough. They have employed the FAST algorithm (D. Lapandic et al., 2017) which is one of the fastest features detection and extraction algorithms available.

In the 2D Point Correspondence stage, when considering two consecutive frames, there are many corresponding points available. The optical flow approach was used to compute 2D point correspondences between two views with the Lucas-Kanade algorithm. The objective is to identify feature point motion from one image to the second image. The computation of the optical flow algorithm is based on a search window (patches) with centers in points of interest identified by the FAST feature detector algorithm.

In the estimation of camera poses and triangulation stages, camera poses are computed and estimated from the matches of two views and feature geometries with using structure from

motion (Martell, 2017). The triangulation will execute once the camera pose estimations are available.

In the stages of estimating the camera poses and 3D point triangulation, they have employed two algorithms (D. Lapandic et al., 2017). The first algorithm determines the subsequent camera locations based on the currently generated point cloud and the matching of interesting feature points in the next view. The rationale behind this first algorithm is with the identification of the motion of image features with optical flow from one view to the other view, evaluation of the camera fundamental matrix and by using that, calculate the essential matrix. Having the essential matrix, it will allow identifying each camera position and rotations in the 3D space. Then projection matrix gets calculated by composing the essential matrix in order to determine feature point locations in the 3D space.

The second algorithm is used to calculate two successive camera locations and then to use triangulation to compute the positions of 3D points (D. Lapandic et al., 2017). Then those point positions determined by continuous and consecutive triangulations are merged into the point cloud being generated, hence 3D reconstruction.

The first algorithm above that they have proposed generates a smaller amount of points with faster execution in increased accuracy. However, the second algorithm is capable of detecting more correspondences but it introduces noise, hence decreases the accuracy.

D. Lapandic et al. (2017) had tried to improve the performance and to archive near real-time reconstruction by sampling the UAV video feeds on a specific frame count while ignoring other frames. By discarding some frames, they have experience incompleteness (holes) of the 3D construction.

J. Ke et al. (2020) presented a real-time 3D visualization from a video feed based on multi-view geometry. To achieve a visualization at the speed of frame rate, they are only able to reconstruct a sparse point cloud. For the video frame feature tracking, the KLT tracker was incorporated. For the acceleration of feature matching and filtering of feature tracks, used trifocal tensor and epipolar geometry. It was proposed to use the SfM to identify camera pose and features for the initial frame, then features will get tracked for newly added frames. J. Ke et al. (2020) experienced limitations of the implemented system including high time consumption when the feature count gets increased and when the feature count becomes reduced, the final point cloud contains holes.

Another research work (Li, 2010) was proposed based on MATLAB to convert from 2D to 3D using multiple images based on sparse depth map calculations. It was developed to use with uncalibrated handheld cameras with unknown camera parameters (intrinsic and extrinsic) or geometries in the scene

The above system in MATLAB comprises of different stages including feature detection, extracting and tracking, image registration, two-view three-dimensional estimation of geometries, calibration of cameras by updating metric transformation, and the reconstruction of the projective scene in 3D. The method they have proposed uses the scale-invariant feature transform (SIFT) (Li, 2010) algorithm to extract the features of the scene and register the feature points in different views. The Random Sample Consensus (RANSAC) algorithm was implemented to remove the outliers in the correspondences identified. Triangulation and bundle adjustment was employed later to estimate and refine the projective reconstruction of the 3D scene. As an important step, they have introduced, an auto-calibration technique to upgrade the projective reconstruction of structures to the metric coordinates. Through these combined techniques, the relative depth information is estimated for feature points among multiple views of the scene.

As a limitation, they have faced an issue of the system being sensitive to the noise in the images and it tends to produce some mismatched 2D feature points between multiple views.

Yuan et al. (2018) have done a study to reduce the time consumption of 3D reconstruction by proposing an improved method based on SfM. Input is a video stream. They proposed a keyframe extraction technique based on the discovery of feature similarity and also proposed a dense algorithm to increase the accuracy of models. It was also proposed to incorporate a 3D model filtering approach to remove resulting models which are redundant. especially incremental SfM required intensive computation compared to SLAM. The capability to produce a dense result is an advantage in the SfM (SLAM usually generates sparse models). A densification algorithm has been used for the final models in the 3D scene.

Incremental SfM includes two major components (Yuan et al., 2018), which are correspondence computation and incremental reconstruction.

Correspondence computation focus on matching different parts of input images and geometrically verify those matches. Different algorithms, techniques, and steps are involved in this stage. SfM generally uses the RANSAC algorithm to eliminate mismatches (Yuan et al., 2018). The output of this stage which are image correspondences and a scene graph (Yuan et al., 2018) will be the input to the next stage. This scene graph is obtained On the completion of

all the pair-wise image matches and generate by chaining the identified feature points which are common among images. The graph is represented by images as nodes and edges will be the pair of confirmed images.

The next Stage, Incremental Reconstruction, includes different steps such as initialization, triangulation and bundle adjustment (Daftry et al., 2015; Yuan et al., 2018). In triangulation, due to the noise, structure from motion typically uses the least square approach to compute spatial points. In SfM, new views will get registered to the model by solving the Perspective-n-Point (PnP) (Yuan et al., 2018). Perspective-n-Point is a technique to solve the camera positions using the points which are triangulated and the projections of their correspondences. Then Bundle Adjustment is utilized for the optimization of camera parameter matrixes and spatial point positions. The output of this stage is a sparse model and optimized camera parameters matrix.

The keyframe Extraction (Yuan et al., 2018) step is incorporated into the correspondence computation stage. That is, there can be a large number of frames that have high similarity among them from the same video. That can lead to redundancy and high time consumption. In this step, unnecessary frames will be ignored. They have initially detected AKAZE feature point sets from consecutive frames and perform feature matching between two frames by a KD-tree, then obtain a set of inliers using RANSAC. At last, the keyframe extraction is performed using inliers.

For the densification, the sparse point cloud is parsed through an operation consists of patch generation and surface reconstruction. Poisson surface reconstruction algorithm (Yuan et al., 2018) is utilized for the surface reconstruction since that algorithm has advantages, including better geometric surface, watertight closure, and details.

A major limitation in the study done by Yuan et al. (2018) is without a consistent video stream, it tends to produce defective results due to dataset incompletions.

Another development (Yu and Park, 2016) has been done with a DJI Phantom 3 drone to construct a 3D scene based on conventional SfM, with using adaptive RANSAC (Gallup, 2011; Pollefeys et al., 2008; Yu and Park, 2016) optimization. This development was mainly focused on the reconstruction of a single object such as a building in the environment. It assumes that camera intrinsic parameters are known and objects are static. Moving objects are considered as outliers from the system. This proposed method was composed of three steps including, aerial image acquisition, extraction and matching of 2D feature points, camera pose estimation and the 3D point cloud generation.

For extraction and matching of 2D feature points, Harris Corner detector and SIFT feature matching algorithms (Yu and Park, 2016) were used in their implementation. For camera pose estimation and the 3D generation, computation of the camera projection matrix and generation of new 3D points are performed at each image.

SfM algorithm should incorporate RANSAC (Random sample consensus) to compute F (Fundamental matrix) and P (Projection matrix) (Yu and Park, 2016). RANSAC is useful to identify outliers using a prior defined threshold value for reprojection error. Instead of using a fixed threshold, this development (Yu and Park, 2016) was done with an adaptive threshold value. It helps to improve the stability of the SfM algorithm. In the adaptive approach, it performs RANSAC calculation iteratively with decreasing threshold values until it extracts sufficient inliers. Limitations of this study can be considered as images should be captured at low altitude (below 100 meters), and for example, if a construction object is a building, its top view (roof) should always be in each image and the path of the UAV should be a curved path around the building with the camera pointing to the center position of the building and it consumes a considerable amount of time for the reconstruction.

Accurate reconstruction heavily depends on the quality of input data (images) (Chen et al., 2018; Daftry et al., 2015). A study (Daftry et al., 2015) was proposed a closed-loop interactive approach to process the reconstruction incrementally in online mode with providing continuous real-time feedback to the user regarding different quality parameters including Ground Sampling Distance (GSD), redundancy, etc on the mesh being generated. As claimed by recent studies, image base reconstruction may archive an accuracy level that can be compared to laser-based reconstruction. However, there are many constraints for that and an arbitrary set of images would never meet that accuracy.

The accuracy of the model is important for industrial applications like automatic façade reconstruction, however current techniques are not up to the required accuracy under unconstrained circumstances (Daftry et al., 2015). The image acquisition strategy is one concern for this aspect. For that and as well as to minimize the accumulated error due to drift, they have proposed an image acquisition strategy that takes images at different distances.

Other aspects for an accurate 3D model are, angels between two views in two consecutive frames should not be too large and there should be overlapping in view cones for better feature matching. Another important parameter is camera calibration (Daftry et al., 2015; Ladikos, 2011). The accuracy of the outcome can be increased with an accurately calibrated camera setup. Another major concern is the completeness/coverage of the scene. To overcome these

issues, (Daftry et al., 2015) proposed to integrate the acquisition process with the reconstruction pipeline itself rather than processing after the acquisition of a set of images. It should perform incrementally to achieve a real-time reconstruction. At each iteration, the process estimates the camera poses for the newly acquired views, updates the sparse point cloud, and then constructs a surface mesh using feature points that are triangulated. This approach will allow computing quality parameters including Ground Sampling Distance (GSD) and the image overlaps.

For better feature matching, (Daftry et al., 2015) have identified that image texture should be non-repetitive and lighting should not vary too much between images and there should be proper illumination in the image. However, it is not always practical to achieve such image sets with the required illumination. A study (J. Hlubik et al., 2018) done with the PhotoScan tool found that even it was able to provide acceptable results with varying lighting conditions, it was unable to reconstruct finer details and the result was a surface that was over-smoothed. (Y. Xie et al., 2019) have identified that due to the shadow areas of captured images, texture maps of the final reconstructed output will have darker areas which result in models that are not sufficient to use in some application areas such as gaming applications or virtual reality applications. As a solution (Y. Xie et al., 2019) have proposed a process known as "intrinsic decomposition" in which the properties of images are inferred into different components. The approach they have used is the decomposition of the intensity of pixel values into specular, illumination, and reflectance components which can be illustrated in equation (2.4). (K. Luo et al., 2020) also identified extend the MVS algorithm to work with specular and reactive areas as well as weak textures is a challenge.

$$I(x) = S(x)R(x) + C(x) \tag{2.4}$$

I(x) is the pixel intensity value observed, S(x), R(x), and C(x) represents the illumination, reflectance, and specular components in order. Their (Y. Xie et al., 2019) proposed solution is a CycleGAN based approach to make a prediction on reflectance component using the captured image. The goal is the elimination of major inconsistent color variations due to shadows. Their reconstruction system was developed on top of the OpenMVS and OpenMVG open source libraries. The result is dependant on the data that is used to train the IntrinsicGAN can be considered as a limitation of that approach. Another work (K. Luo et al., 2020) was done a study on the learning-based multi-view stereo method. They identified that how to incorporate learned percept feature set to have a robust matching confidence volume is a significant question that remains in the learning-based MVS approach. They also identified that with higher quality training data, the system can provide accurate results with increased accuracy.

Xiao et al., (2020) explored another use case of the combination of SfM and MVS for topography analysis with the use of input as low-resolution satellite image data (elevation data). The output will be generated higher resolution ortho mosaics and digital elevation models which can be used for the geographic surface analysis with critical information including slope and height data.

Drift (Daftry et al., 2015) is an issue in the incremental SfM method. Drift will occur due to the border area of the interested area is covered with only a few images by the camera network comparing to the center area of the scene. They have proposed to modify the strategy in image acquisition as a solution to that problem by using a multiple-scale camera network to capture image data at varying distances to achieve a more accurate dense reconstruction outcome. However, this approach is incapable of solving limitations such as tradeoffs between high-resolution and accuracy. The lower resolution gives higher efficiency and lower memory usage (J. Hlubik et al., 2018) but with less accuracy of the output point cloud.

Major strengths and limitations for above reviewed previous related work can be summarized as in table 2.1

Related Work	**Strengths**	**Limitations**
Chen et al. (2018)	• Output acceptable results for a small and single object.	• Required uniformly scaled images with the same resolution. • Memory limitation issues for large data sets
D. Lapandic et al. (2017)	• Fast feature detection since it uses the FAST algorithm. • Achieve near real-time performance	• Not as robust and accurate as SIFT algorithm. • Video frame selection was don on a specific frame interval
J. Ke et al. (2020)	• Real-time 3D visualization with less feature count.	• High time consumption with increased feature count. • Only capable of constructing a sparse point cloud
Li (2010)	• Capable to use with uncalibrated handheld cameras	• Sensitive to noise and produced mismatched corresponding points.
Yuan et al. (2018)	• Introduces a keyframe selection strategy based on feature similarity	• Tend to produce defective results for inconsistent video streams.

Yu and Park (2016)	• Uses adaptive thresholding method for RANSAC	• Unable to work with an uncalibrated camera. • UAV should capture by following a curved path in a low altitude
Xiao et al., (2020)	• Combined SfM and MVS for better densification	• Limited to satellite image data (elevation data)

Table 2.1: Major strengths and limitations in related work

2.4 Summary

There is a strong requirement and importance to construct 3D information from 2D images or video frames. UAVs can simplify the capturing process and capable to feed video data with additional information such as GPS and orientation data which can help for the reconstruction task. However, there are only a few references to research attempts that utilize additional data available from UAVs to improve the accuracy.

It is a challenging task to recover reliable 3D information from 2D images of an environment. There have been several research attempts to meet this challenge with each having different limitations. Several research works (Chen et al., 2018; Daftry et al., 2015) prove that it is important to have high-quality input data with overlapping features between multiple views.

3D reconstruction, in general, can be considered as three stages that are correspondences, geometry, and surface. Structure from motion (SfM) is a major photogrammetry and computer vision technique to estimate 3D structures from a sequence of 2D images. SfM consists of key stages including feature point detection, extraction and matching, camera pose estimation, triangulation, and point cloud reconstruction.

Different algorithms such as Bundle Adjustment (BA) and RANSAC (RANdom SAmple Consensus) should be incorporated in the reconstruction process to optimize the camera poses by minimizing the reprojection error and to remove outliers or mismatched points in feature detection and matching stages.

CHAPTER 3
METHODOLOGY

3.1 Representation of the Problem

With the literature review, it can be identified that a 3D graphical representation of a real environment can be helpful for the improvement of many industries and areas. Even though there are more expensive 3D model reconstruction techniques available using Lidar sensors and depth sensors, reconstruction using 2D images captured from a camera is a cost-effective solution.

Capturing process can be automated and simplified with the help of UAVs and captured video data including the GPS data can be fed to the ground computer via wifi. This thesis focuses on the recovery of camera position and orientation data and then the process of 3D scene reconstruction from received video frames. This study will address the problems of camera pose estimations and offline dense 3D reconstruction.

3.2 Proposed System Overview

This study proposes a reconstruction approach with dense outcome based on the structure from motion (SfM) technique to estimate the three-dimensional structure of objects in the environment from two-dimensional image sequences using the video feeds and flight data such as GPS received from a UAV. Therefore the main problem considered in estimating 3D point positions from multiple frames and their feature correspondences. SfM process involves continuous estimation of both 3D geometry (structure) and camera pose (motion)

The reconstruction process will be executed on a ground computer. Core algorithms will be operated on frames of video feeds. However, cameras with a high FPS (frames per second) value, can result in inefficient and unnecessary processing of frames as there can be a large number of frames to process and many frames can be almost the same and overlap. To overcome that, the algorithm will be implemented with the capability to defined a time interval to process the frames and video frame selection algorithm under the assumption that the area of interest remains static over the reconstruction process.

The proposed reconstruction pipeline consists of different stages (Figure 3.2),

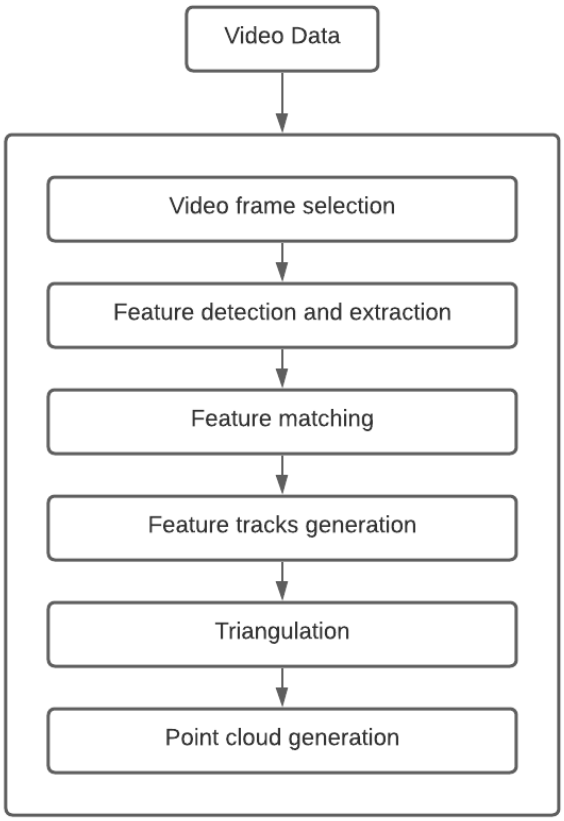

Figure 3.1: Proposed Reconstruction System

1. **Preparation of data**

The data required for this process are video feeds, UAV retrieved video data files should be presented to the next stages of the algorithm.

2. **Video frame selection**

Video frame selection is done by measuring the transformation between frames and by selecting based on a certain threshold. That is, initially obtain the feature points for subsequent frames by Lucas-Kanade optical flow algorithm and using those points calculate the affine transformation between frames. Frame selection is performed by using threshold values for translation and rotation between frames.

3. **Feature detection**

In this stage, important points known as features will be detected. These features are useful to identify, locate, and categorization of objects in image frames. The idea behind feature detection is to ultimately identify the same interesting points in multiple views. This process will output a set of feature vectors (descriptors). This study involves the process of a large amount of data

(data that covers large environments). The feature detection phase will get incorporated with the SIFT (Scale Invariant Feature Transform) feature detection algorithm. Comparison and characteristics (Ajayi, 2020; Mouats et al., 2018; Govender, 2009) of different feature detection algorithms illustrated in table 3.1

Feature detector algorithm	Characteristics
Scale Invariant Feature Transform (SIFT)	Invariant to rotation, scale, and also for illumination changes, and noise. Compared to others, execution is time-consuming but gives an optimal detection capability.
Principal Component Analysis (PCA)-SIFT	Lower effectiveness in feature detection compared to SIFT. The algorithm intends to reduce the time consumption in detection and matching.
Speeded Up Robust Features (SURF)	Scale and rotation invariant. Lower execution time consumption compared to other feature detector algorithms but the accuracy and detection capability is lower.
Features from Accelerated Segment Test (FAST)	Invariant to scale and rotation. A circle of 16 pixels is used around the interesting pixel to identify whether that pixel is a corner pixel. Computationally efficient algorithm when the noise does not exist.
Smallest Uni-value Segment Assimilating Nucleus (SUSAN)	Not invariant to scale. A corner detection algorithm that computes intensity differences to identify corners.
Modified Harris Corner Detector (MHCD)	Rotation invariant. Optimal performance can obtain without a scale variance. The algorithm is tests pixels to identify corners. the sum of squared differences (SSD) is used as the similarity measure

Table 3.1: Characteristics of feature detection algorithms

4. Feature matching and extraction

Once identifying interest points and have feature descriptors, the feature matching phase will use an approximate nearest neighbour algorithm to calculate the distance (in high dimension space) to identify corresponding descriptors between frames. It needs to be considered a certain threshold for matching features in image pairs. The Random Sample Consensus (RANSAC)

algorithm will be incorporated to remove the outliers from the identified correspondences. Usage of the RANSAC algorithms is described later in this chapter.

Since the input for this work is assumed to be a video feed and not a set of unordered images, implementation of the feature matching stage will be done by assuming consecutive frames have higher matchings compared with random pairs of frames. Hence for a selected frame, instead of obtaining matches against all other frames, it matches only a predefined number of subsequent frames to achieve improved performance as well as an improved final output. When compared to existing reconstruction tools such as Bundler SfM, which try to obtain matches against all image pairs by assuming an unordered collection of images require extensive computation and produce more outliers leading to degraded performance and final output.

5. Camera pose estimation and triangulation

The camera pose which is the camera position and heading will be estimated using identified 2D feature points, which are visual inputs. Estimation of camera poses from visual data will be provided by pair of view matches.

For each camera pose obtained, triangulation is performed using those poses and 3D points. In this process, it computes the locations and orientation (pose) of two consecutive camera views and determines the positions of 3D points using triangulation. These points will be added to the point cloud.

When estimating camera poses, since this study assumes cameras have pre-calibrated, the essential matrix can be calculated using corresponding points. Then this matrix will contain information about the relative orientation. Then current projection matrix can be computed from the essential matrix. With these matrices, it can identify the position and rotation of each camera view in space, and with obtained projection matrix, the triangulation process will be performed to determine points in 3D space.

6. Depth map estimation and model generation

From the above stage, by triangulation, the depth of 3D points can be obtained and generate a point cloud with a large number of points. Triangular meshes will be created for these depth maps to generate 3D models.

3.2.1 Incremental Reconstruction

Reconstruction in the proposed system is incremental. Once the reconstruction is initiated by estimating poses of an image pair followed by triangulating visible features of that image pair which generates 3D points and adding them to the point cloud, this will iterate by adding the suitable next view incrementally to the process. Therefore, this is a frame-by-frame iterative process to achieve a dense surface reconstruction. This process stops when all the reconstructable views are part of the scene.

In this incremental process, bundle adjustment (BA) optimization is also performed to optimize the camera pose as well as 3D point locations for visual reconstruction. It will be achieved by minimizing the re-projection errors between observed and predicted locations of image feature points. Bundle adjustment is performed as video frames are added to the incremental process. The result of BA is a refined more fitted 3D reconstruction. It is a process of global optimization that executes recursively for all the views.

3.2.2 Geo Specific Pose Estimation

Camera pose estimation, that is, based only on image features can only obtain the pose in a local coordinate system which is defined by and relative to the first camera view. Therefore, to obtain 3D models which are geotagged, the camera poses can be identified in a geographic coordinate system.

The proposed system could be extended to use GPS/INS data send from the UAV and an extended Kalman filter can be used to combine the image feature-based vision measurements and the GPS/INS measurements to retrieve geo-specific camera poses. From those poses, it can be obtained geo-specific 3D points. That is, 3D point locations of identified image features in each camera can obtain in an orthogonal, earth-centered, earth-fixed coordinate system such as Universal Transverse Mercator (UTM). The procedure in the extended Kalman filter is a smooth motion model. Therefore, to model the pose over time, it assumes there is a constant velocity change in rotation and translation.

3.2.3 Dense Reconstruction

This study assumes cameras are calibrated and no need for a calibration step. Therefore, that can increase computational efficiency. Then the computationally complex stage is the surface reconstruction from multiple views. With the identified feature correspondences in the above feature matching stage, a collection of tracks can be created.

As in the figure 3.3, a track is a set of the matched point across multiple views. For example, if consider the point Xi in space, that point in the points of xi1, xi2 and xi3 of three views which

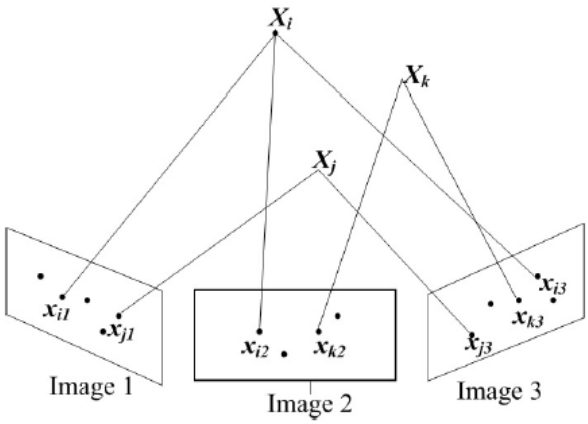

Figure 3.2: Feature tracks

forms a track of Xi. Image coordinates of those tracked features will be the input to the next stages of structure from motion algorithm. Then, the general SfM problem is to find the 3D coordinate of the point (Xi) in the scene using those tracked features. With a large number of correspondences and tracks, these different views can be densely connected.

In a typical 3D reconstruction process, the next stage after the feature detection and matching would be the calibration of camera intrinsic parameters. This calibration stage is not necessary for this study as it assumes cameras are pre-calibrated and camera extrinsic and intrinsic parameters are known.

The surface reconstruction process will get initialized with the first two frames/images. From those two images, the first image will be used as the reference image. That is the world origin is assumed to be in the first image. Then the projective matrix (equation (3.1)) of the camera is defined by,

$$P_1 = A[I|0] \tag{3.1}$$

Where "*A"* is the 3 x 3 intrinsic matrix, "*I"* is the identity matrix, The projection matrix will be evaluated using the camera motion parameters R (rotation) and t (translation) which can be determined by the essential matrix. With the R and t, the projective matrix (equation (3.2)) can be defined by,

$$P_2 = A[R|t] \tag{3.2}$$

Then the 3D coordinates will be computed from matching points using linear triangulation. Then the projective matrix is optimized using the bundle adjustment. As in figure 3.3, if x_{i1}, x_{i2}

are matching points of the 3D coordinate X_i, then P_2 is optimized with the minimized function (equation (3.3)),

$$min \sum_{i=1}^{n} d(PX_i - x_i)^2 \tag{3.3}$$

Multiview reconstruction will be performed by merging subsequent images into the initial reconstruction as an image by image increment. As in figure 3.3, when merging the third image, first, it will calculate a new projective matrix using the matches with already reconstructed points. Then reconstruction gets updated with new points as well as refinements of points and removal of incorrect points.

3.2.4 RANSAC

The proposed solution will employ the RANSAC (RANdom SAmple Consensus) algorithm which is a robust estimation algorithm in the feature extraction and matching stage to remove outliers or mismatched points. The RANSAC method will get converge to accommodate only the inliers after some iterations.

Figure 3.4 is from the literature that illustrates the resulting feature points after applying the RANSAC algorithm. It represents two consecutive frames/images which have two view angles. Matched feature points from two images before and after applying the RANSAC are shown. In figure 3.4, It can be seen that RANSAC, will result in a refined set of feature points.

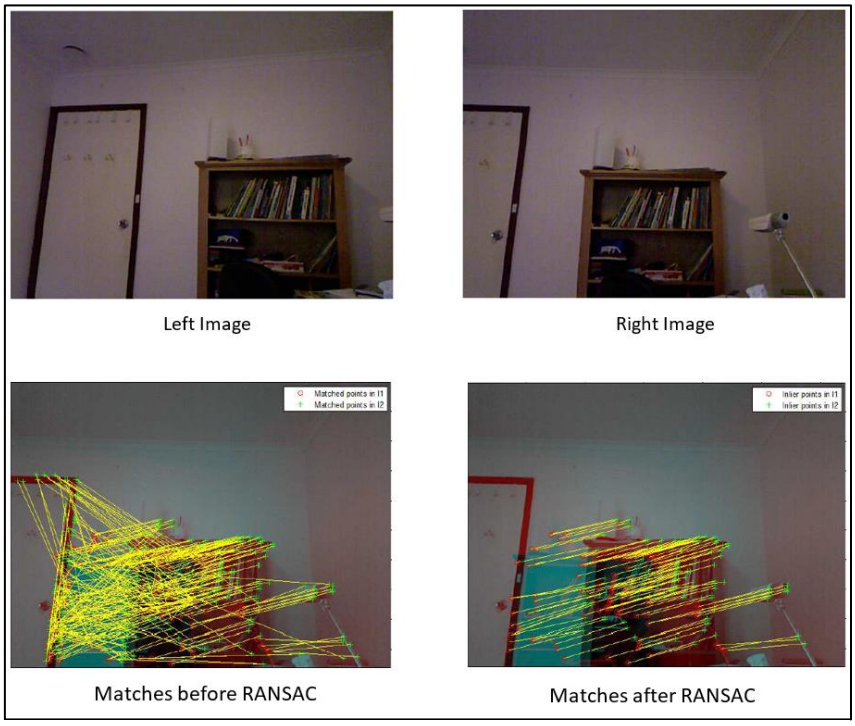

Figure 3.3: RANSAC example from literature

For the above refinement, this algorithm classifies data as inliers and outliers. For the classification, it uses a cost function with a threshold value. The threshold value is related to the feature point count. Since this study has to process a large amount of data (data that covers large environments), to achieve acceptable accuracy with fast and rapid reconstruction, from the literature it can be identified that a threshold value around 100 is appropriate. A new feature point will be added if the current feature count is below the threshold. A feature point gets removed if a matching point cannot be found in the new image frame.

3.3 Image/Video Capturing

When capturing the videos for the 3D reconstruction purpose, it's not only important to capture high resolution and a large number of images, but the quality and the viewing angles of capturing will also vastly affect the final 3D models. For the proposed system the area that needs to be reconstructed will be captured by flying the UAV in a spiral path with increasing the altitude to cover the entire area and to obtain densely sampled images with a large number of overlapping features. Figure 3.1 is from the literature that represents flying path (camera positions).

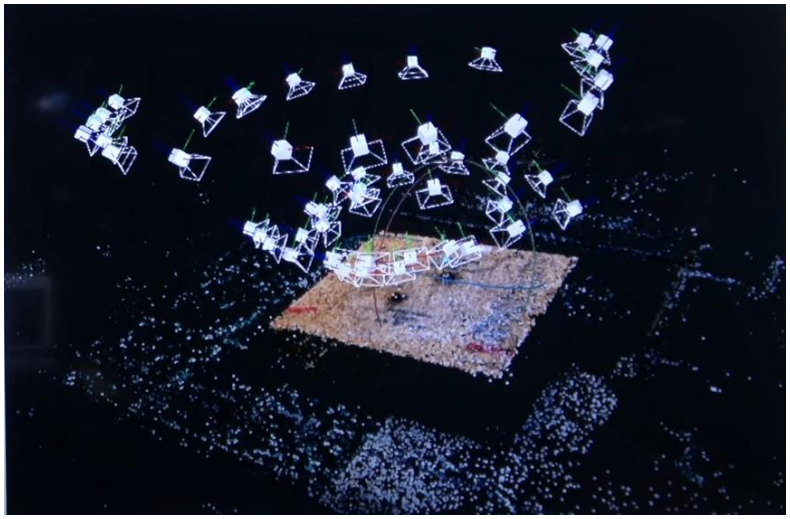

Figure 3.4: Capturing path sample from literature

This is important since sparsely sampled images can result in disconnected points or holes in reconstruction outcomes. Cameras in the UAV need to be arranged in a way that they have minimal overlap in terms of field of view. Image acquisition strategy will not much be considered in this work as it is not in the scope of the study. For this study, it assumes that the objects/scene that will be captured remain static.

3.4 Implementation

A system prototype was implemented as a proof of the concept and to exhibit the reconstruction results. C++ programming language was used with the OpenCV as the main library and different other libraries for the development. C++ is a powerful library for computer vision-related tasks. The prototype was implemented with a command-line interface to pass the parameters including reconstruction mode (from images or a video), the path to images/video file, and the feature detector algorithm to use. Implementation for different stages with challenges, considerations, and assumptions are explained in the next sections These stages are consist of debug outputs to visualize the progress of the reconstruction.

3.4.1 Video Frame Selection

The algorithm for video frame selection for the reconstruction process was implemented for each two video frames, it will first identify 100 strong corners in the first frame using the Shi-Tomasi algorithm and then calculate the optical flow for the identified corners using the iterative Lucas-Kanade algorithm and get set of feature points for the second frame. Using those two sets of feature point sets implementation is done to calculate the optimal limited affine transformation. Then it will compare with the predefined threshold values for translation in X and Y directions to select frames which get exceeds the threshold values. Selected frames will write to a folder and read them back and feed to the reconstruction pipeline.

3.4.2 Feature Extraction and Matching

Initially for each selected frame, using the feature detector algorithm defined in the command line, it will detect key features, compute and store feature descriptors in a feature descriptor. These descriptors will be the input to the feature matching stage. The improved and novel feature matching algorithm is implemented to compute matches for each selected frame for a predefined number of subsequent frames. The assumption behind that is, in a video frame sequence, consecutive frames have higher matchings compared with random pairs of frames. Track generation is important as it helps to have consistency among multiple views.

To obtain robust matches for each selected pair of video frames, the feature matching process is consists of multiple filtering stages. When considering two frames, for example, frame 1 and frame 2, first obtain raw matches from frame 1 to frame 2 using the knn matcher to obtain k best matches from descriptor followed by the Lowe's ratio test. Then obtain raw matches from frame 2 to frame 1 using the knn matcher followed by the Lowe's ratio test similar to above.

The output of these two will filter by removing the matches which do not exist in both. That is to make sure there are ono-to-one matches between two matching frames. The final filtering stage is performed by applying the epipolar constraint by calculating the fundamental matrix with the RANSAC algorithm to further remove outliers. Example debug outputs for detected and final filtered matches are illustrated in figure 3.5 and figure 3.6

Figure 3.5: Debug outputs for detected features

Figure 3.6: Debug outputs for final filtered matches

3.4.3 Feature Tracks and Point Cloud Generation

Generation of feature tracks is implemented using a graph data structure. Image features will be the vertices and feature matches will be the edges in the graph. Connected components in the graph become tracks. The final obtained tracks will be stored in a vector.

These feature tracks will become the input to the generation of the sparse point cloud. The sfm module in the OpenCV is utilized to generate the sparse point cloud, to recover the camera poses (rotation and translation) for each view, and to estimate 3D point locations by

triangulation. Then for each point, it will retrieve the point colors using image pixel locations. Identified point locations with color information will plot using viz3d module in the OpenCV. For the densification of this sparse point, implementation is utilized cloud, the OpenMVS library.

3.5 Summary

When compared with previous related work that has been reviewed in the literature review section, This work has addressed some issues that have been identified. Those are video frame selection algorithm to select frames based on the transformation between frames with a certain threshold instead of selection based on a certain frame count interval (D. Lapandic et al., 2017; J. Ke et al., 2020). That is, initially obtain the feature points for subsequent frames by Lucas-Kanade optical flow algorithm and using those points calculate the affine transformation between frames. Another major improvement is done to the feature matching stage by assuming consecutive frames have higher matchings compared with random pairs of frames. Hence for a selected frame, instead of obtaining matches against all other frames (Chen et al., 2018; J. Hlubik et al., 2018; Yuan et al., 2018), it matches only a predefined number of subsequent frames to achieve improved performance as well as an improved final output.

This study focuses on offline dense 3D reconstruction using UAV video data feeds. The reconstruction process will run on a ground computer in which UAV data can be streamed via wifi. It assumes that the objects/scenes that will be captured remain static for this study. The design overview illustrates a dense reconstruction based on the structure from motion (SfM) technique to estimate the 3D geometry (structure) and camera pose (motion). The main stages of the reconstruction pipeline consist of video frame selection, feature detection, feature matching and extraction, camera pose estimation and triangulation, and depth map estimation and model generation.

The prototype was implemented with the capability to work with a user-selected feature detection algorithm. This study has to process a large amount of UAV video data (data that covers outdoor environments). The RANSAC algorithm will be incorporated in the feature matching and extraction stage to remove outliers from the identified correspondences. Camera pose estimation will be performed using identified 2D feature points. For the obtained camera poses, triangulation is performed to determine the positions of 3D points and identified points will be added to the point cloud.

CHAPTER 4
EVALUATION AND RESULTS

4.1 Evaluation Plan

This work addresses the problem of recovering 3D scenes from a sequence of 2D video image frames captured using a UAV. This study will focus on the reconstruction pipeline, starting from the recovery of camera position and orientation information from video feed frames and using those pose estimations, carry out the reconstruction process, hence the reconstruction process mainly uses the Structure from Motion (SfM) technique with other algorithms including RANSAC and Bundle Adjustment (BA) to remove outliers and to perform optimization.

Therefore, the main objective is to implement a proof of concept prototype which automatically and efficiently reconstructs detailed 3D models offline from UAV video feeds received of a static environment. For this study and for the evaluation purpose, an environment such as a building with its surrounding will be selected. The following evaluation plan is proposed to determine whether the actual outcome of the work is effective in terms of the performance and results and whether the proposed outcomes are achieved.

4.1.1 Data Sets

Most of the computer vision and image processing-related algorithms evaluated against standard data sets such as MINST. Researchers use these data sets as a benchmark or ground truth for new implementations to compare the accuracy and efficiency. However, when it comes to scene reconstruction from An ariel video feed from a UAV, such standard and well-established video data with ground truth models couldn't be found.

There are some image sets such as Fountain (figure 4.1) and Herz-Jesu that can be found in the literature which were used to evaluate the reconstruction results. Those image data sets will be used in this implementation as well to calculate different metrics to compare the result with the other implementations. These benchmark data sets are available with high-quality 3D models generated using the Lidar sensors which can be used as a ground truth. Apart from those benchmark image data sets, own collected data including images and videos of different static scenes with different resolutions and different cameras will be used to calculate the metrics and evaluate the performance. The reconstruction will also be evaluated for areal image and video data available online for different scenes such as historical locations, buildings, accident sites,

etc., and their quantitative results such as reconstruction performance have been analyzed and presented.

Figure 4.1: Image samples from Fountain Dataset

Figure 4.2: Image samples from Hertzjesu Dataset

4.1.2 Evaluation Approach

The reconstruction pipeline is implemented to automatically calibrate the camera intrinsic and extrinsic parameters using input video frames instead of demanding the user to feed them as external inputs since, in the real world, it is not always possible to find camera parameters for all the available data. Therefore, it needs to evaluate the accuracy of the calibration process. That can be done by comparing the recovered camera matrices using data captured from a known calibrated camera (known intrinsic matrix).

To evaluate the overall accuracy of the generated 3D models, above-specified benchmark data sets such as Fountain and Herz-Jesu will be used to reconstruct the scene. Then a software tool such as Meshlab can be used to have qualitative and quantitative measures on the final output with their Lidar generated ground truth models.

For a qualitative measure, in the Meshlab, generated 3D models can be roughly aligned manually with the ground truth models by loading both models and specifying a few numbers of corresponding points between generated and imported models. Then Meshlab is capable of refining the alignment process (using the iterative closest point/ICP algorithm). With aligned models, it can have an idea of how well the reconstruction process performed.

For a quantitative measure, Meshlab allows computing the error/distance for 3D points between the generated model and the corresponding ground-truth models. Calculating the average distance for a different set of corresponding points will give a quantitative idea of the final output.

Different other metrics will also be used to evaluate the outcome and the performance of the reconstruction pipeline and to get quantitative measures, Those metrics are,

- Feature match percentage

This is the number of matches between images from the total detected features. It can be calculated using “Number of matches / Total detected features”

- Reconstruction performance

The performance will be evaluated by calculating the average time consumed, for the entire pipeline, and each stage in the pipeline. It will also measure the CPU and memory usage using a profiler.

- Correct match percentage.

This is the number of matches after removing outliers after applying the RANSAC algorithm. It can be calculated using “Number of correct matches / Total detected features”

- Mean distance

Distance measure gives an accuracy of a match. The smaller the distance is better.

- The number of points in the point cloud

The number of points in the generated point cloud indicates the density details level of the generated models.

The above metrics will be calculated for different feature detection algorithms including SIFT, ORB, and AKAZE and present the outcome and the performance. Evaluation and experimental results for different datasets with different frame resolutions will be presented and analyzed.

4.1.3 Hausdorff Distance

Hausdorff distance will be used to measure the distance between generated point clouds and ground truth models. In general Hausdorff distance is the maximum between the two meshes. Which can be calculated using the equation (4.1),

$$d_H(X,Y) = \max\{sup_{x\epsilon X}\ inf_{y\epsilon Y}\ d(x,y), sup_{y\epsilon Y}\ inf_{x\epsilon X}\ d(x,y)\} \quad (4.1)$$

However, the Meshlab tool measures one-sided Hausdorff distances. That is only the $sup_{x\epsilon X}\ inf_{y\epsilon Y}\ d(x,y)$. Therefore, the results will depend on the selection of *X* (sampling mesh) and *Y* (target mesh). The calculation of the above formula will be proceeded with a sampling approach by taking some points in mesh *X* and for each point *x*, search and measure the distance for the closest point *y* in mesh *Y*. Therefore, the results will be affected by the number of points selected over *X*.

For the evaluation of the reconstruction system, sampling mesh (*X*) will be selected as the ground truth model and target mesh (*Y*) will be selected as the generated model. Vertex sampling will be used as the sampling option with the sampling count as the number of vertexes in the sampling mesh. It gives results in the mesh units as well as with respect to the diagonal of the bounding box of the mesh which is a result independent of the mesh/model units.

4.2 Results

Quantitative results with the proposed prototype for different datasets will be presented in this section. Results will be evaluated for Fountain and Hertzjesu datasets as described in above sections since ground truth models are available for those datasets. Final point cloud outputs (Figure 4.3 - Fountain, Figure 4.5 - Hertzjesu) from the system and their recovered camera poses (Figure 4.4 - Fountain, Figure 4.6 – Hertzjesu) will illustrate below.

Figure 4.3: Dense point cloud output for Fountain dataset

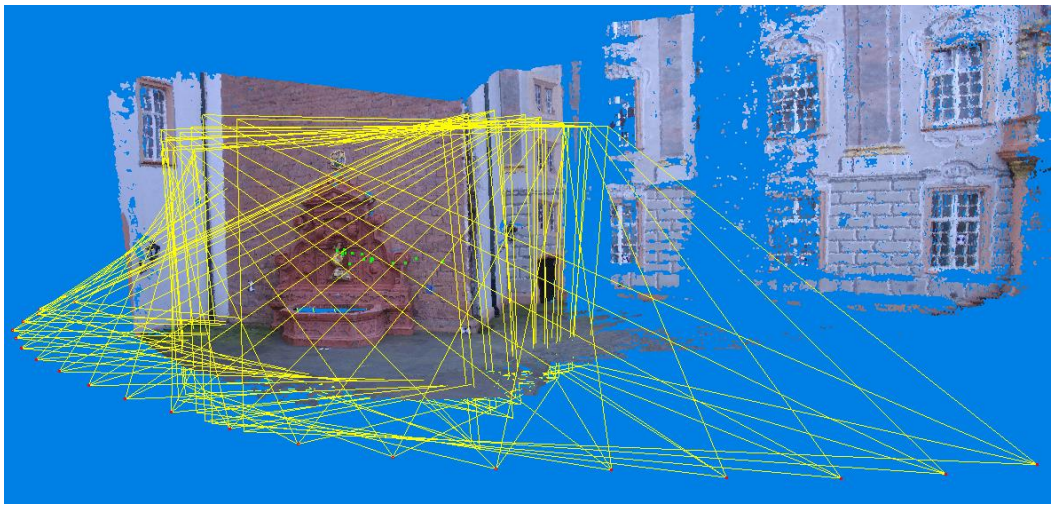

Figure 4.4: Estimated camera poses for Fountain dataset

Figure 4.5: Dense point cloud output for Hertzjesu dataset

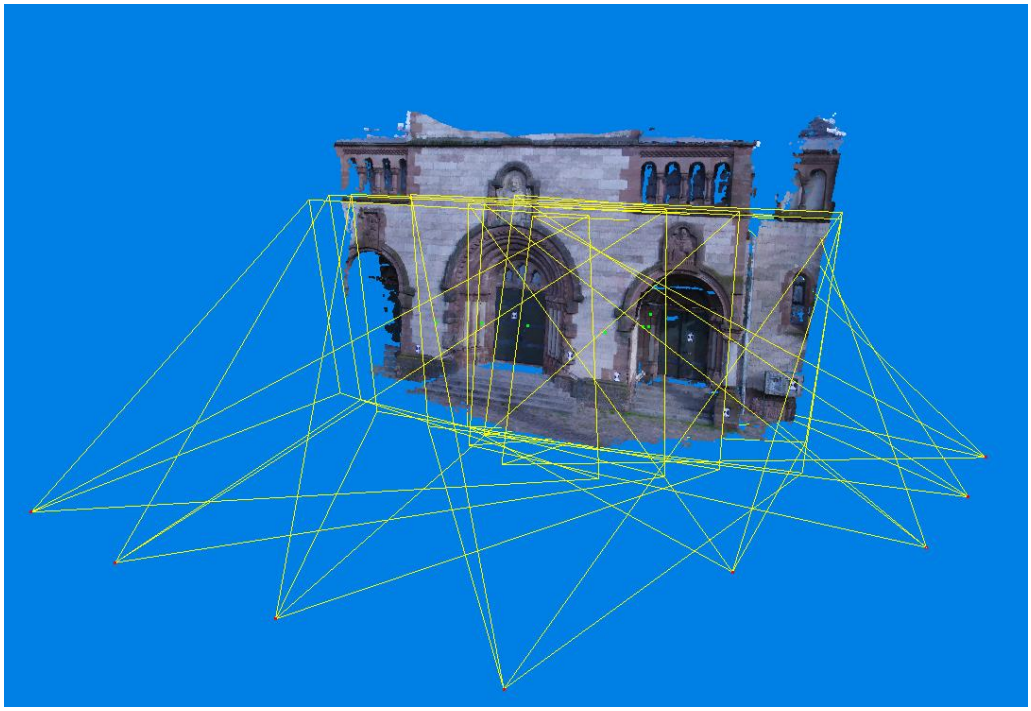

Figure 4.6: Estimated camera poses for Hertzjesu dataset

Hausdorff distance calculation between the ground truth and generated models from implemented prototype and the VisualSFM 3D-reconstruction tool will be compared in the below tables for Fountain (Table 4.1) and Hertzjesu (Table 4.2) datasets.

	VisualSFM	Implemented Prototype
Min Distance	0.000010	0.000004
Max Distance	0.036388	0.039027
RMS (root mean square) value	0.010515	0.010118
Mean Distance	0.005362	0.004639

Table 4.1: Hausdorff distance comparison for Hertzjesu dataset

	VisualSFM	Implemented Prototype
Min Distance	0.000000	0.000001
Max Distance	0.320314	0.062872
RMS (root mean square) value	0.015886	0.006118
Mean Distance	0.011085	0.002720

Table 4.2: Hausdorff distance comparison for Fountain dataset

From the above Hausdorff distance results, it can identify that implemented prototype is capable of producing better results compared to the VisualSFM tool.

As described in previous chapters, implementation of the feature matching stage was done by assuming subsequent frames have higher matchings hence for a selected frame, instead of obtaining matches against all other frames, it matches only a predefined number of subsequent frames. Evaluation results for feature matching stage with improved algorithm to match an only pre-defined number of subsequent frames vs matching against all frames are illustrated in the following tables for Fountain dataset (Table 4.3) and Hertzjesu dataset (Table 4.4).

	Improved algorithm (Match only two subsequent frames)	Match against all frames
Time consumption	88.59 seconds	413.21 seconds
Memory usage	3394.11 KB	6228.72 KB
Mean Hausdorff distance with ground truth	0.002720	0.003411

Table 4.3: Feature matching stage with improved algorithm for Fountain dataset

Performance evaluation results in above table 4.3 can be represented graphically as below in the Figure 4.7

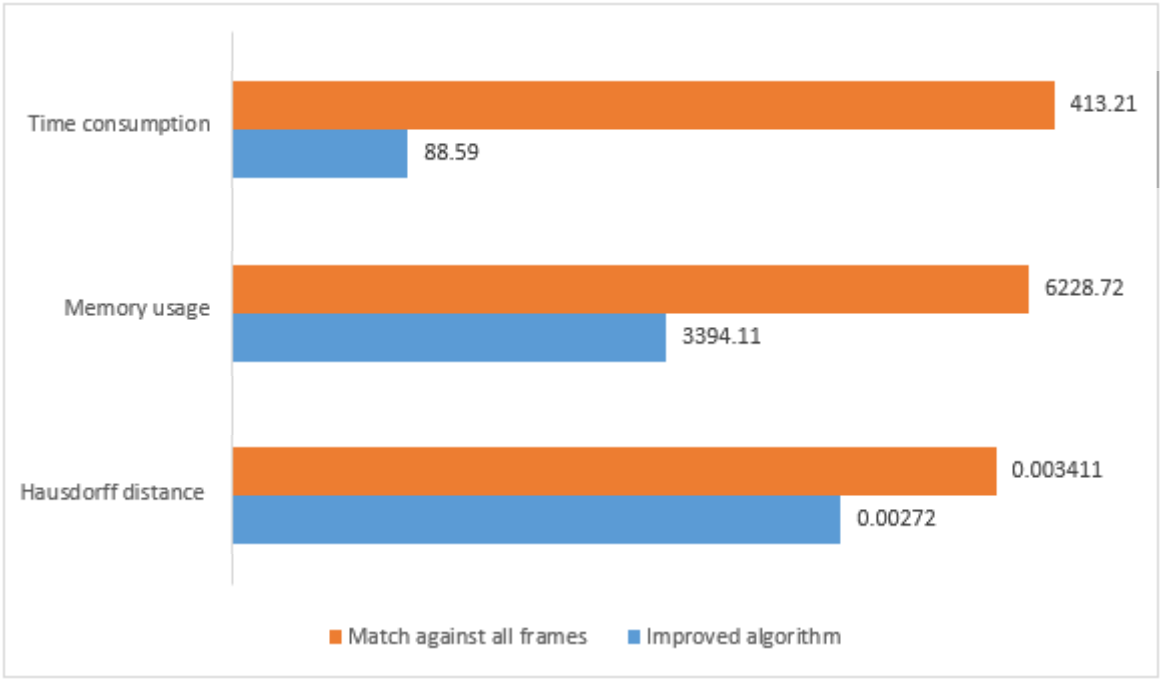

Figure 4.7: Evaluation results for feature matching stage with improved algorithm for Fountain dataset

	Improved algorithm (Match only one subsequent frame)	Match against all frames
Time consumption	5.3 seconds	16.2 seconds
Memory usage	129.12 KB	220.66 KB
Mean Hausdorff distance with ground truth	0.004639	0.006230

Table 4.4: Feature matching stage with improved algorithm for Hertzjesu dataset

Performance evaluation results in above table 4.4 can be represented graphically as below in the Figure 4.8

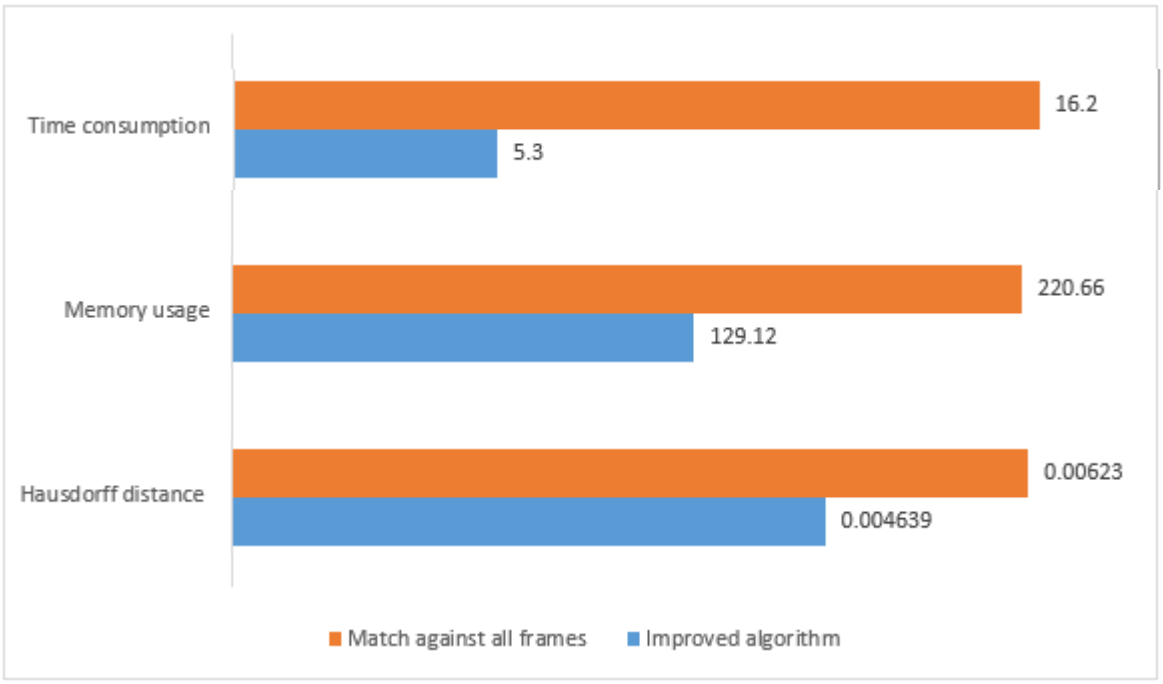

Figure 4.8: Evaluation results for feature matching stage with improved algorithm for Hertzjesu dataset

The above evaluation results for the feature matching stage show that the improved algorithm gives better results (lower Hausdorff distance with ground truth) and significant performance (time and memory usage) improvement. This is due to the subsequent frames in a video have a higher number of matches. When the matching stage is performed against all the frames there can be invalid matches which will result in a higher Hausdorff distance with ground truth.

Quantitative results of different other metrics to evaluate the outcome and the performance of the reconstruction pipeline are illustrated in the following tables for both the Fountain dataset (Table 4.5) and the Hertzjesu dataset (Table 4.6).

Total time consumed	887.242 seconds
Average feature match percentage between frames	37.25
Average correct match percentage (after applying RANSAC) between frames	35.65
Sparse point cloud density	50230
Dense point cloud density	3880655

Table 4.5: Performance and outcome measures with Fountain dataset

Total time consumed	177.14 seconds
Average feature match percentage between frames	22.6
Average correct match percentage (after applying RANSAC) between frames	21.54
Sparse point cloud density	4657
Dense point cloud density	1824456

Table 4.6: Performance and outcome measures with Hertzjesu dataset

Apart from the above benchmark data sets, output results, performance, and dataset samples for aerial video feeds are illustrated below.

Figure 4.9 illustrated a sample set of frames extracted for the “Independence Square” aerial video with the use of implemented frame extraction algorithm. X and Y translation threshold values were set to 15. The Video file is 6.16 megabytes, duration of 7 seconds, and a total of

380 frames. The extraction algorithm was able to select 23 frames from 380 available frames based on the defined threshold values. Figure 4.10 illustrates the reconstructed dense point cloud output for the “Independence Square” using extracted frames. Table 4.7 illustrates quantitative results of different other metrics to evaluate the outcome and the performance of the reconstruction pipeline for the “Independence Square” aerial video.

Figure 4.9: Extracted frame samples from Independence Square aerial video

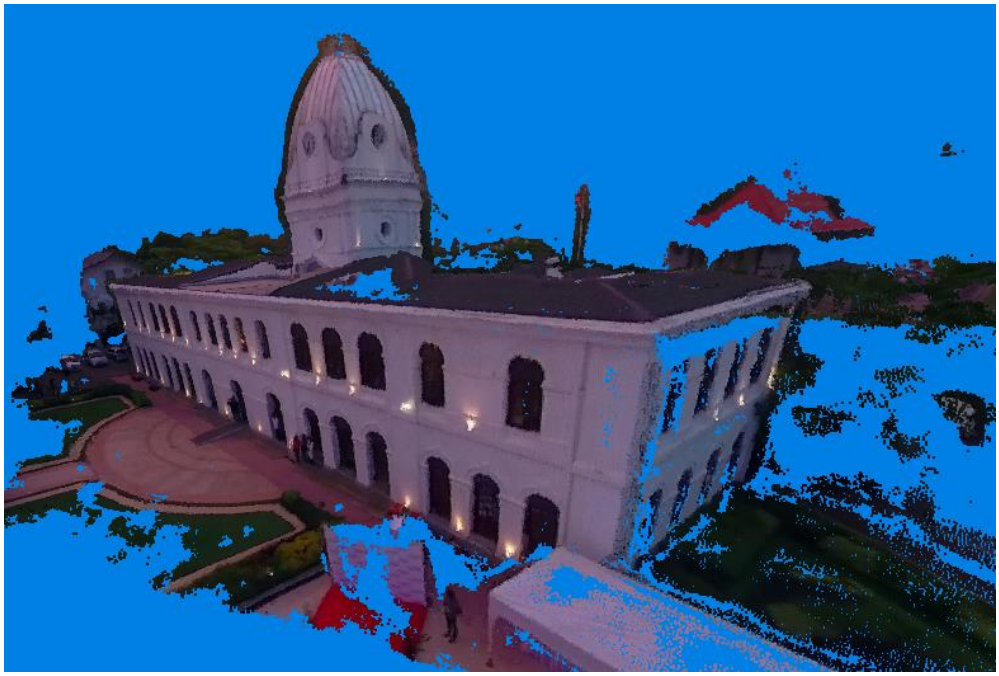

Figure 4.10: Dense point cloud output for Independence Square aerial video

Total time consumed	195.1 seconds
Time consumes for frame extraction	21.75 seconds
Average feature match percentage between frames	29.95
Average correct match percentage (after applying RANSAC) between frames	28.3
Sparse point cloud density	12949
Dense point cloud density	533284

Table 4.7: Performance and outcome measures with “Independence Square” aerial video

Figure 4.11 illustrated a sample set of frames selected for the Sigiriya historical site aerial video for the reconstruction process. Figure 4.12 illustrates the reconstructed sparse point cloud output. Figure 4.13 and Figure 4.14 illustrate the reconstructed dense point cloud output for the Sigiriya using extracted frames. Table 4.8 illustrates quantitative results of different other metrics to evaluate the outcome and the performance of the reconstruction pipeline for the Sigiriya historical site aerial video.

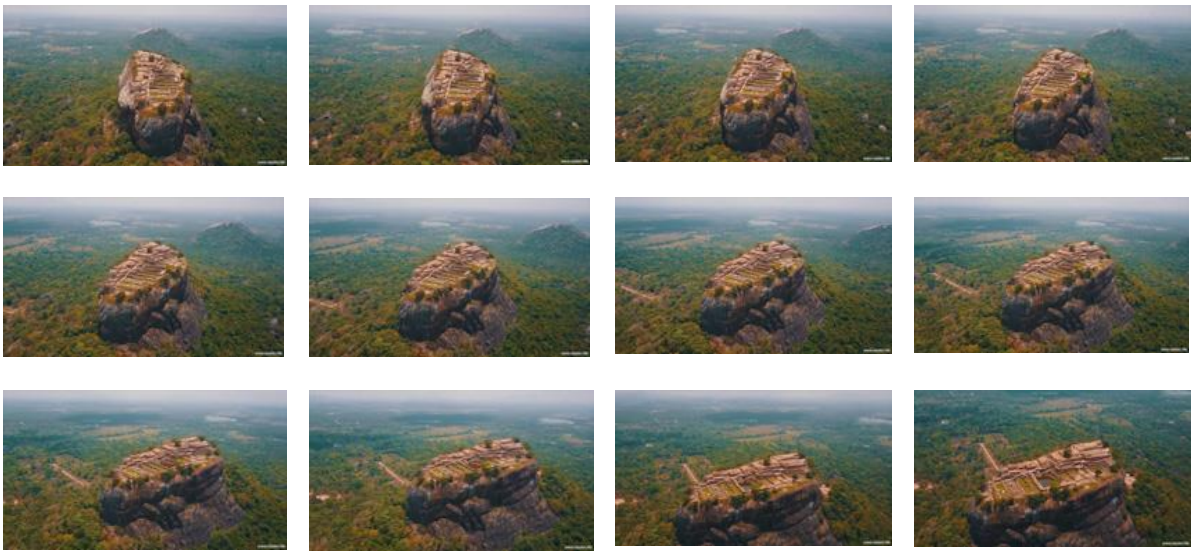

Figure 4.11: Frame samples from Sigiriya aerial video

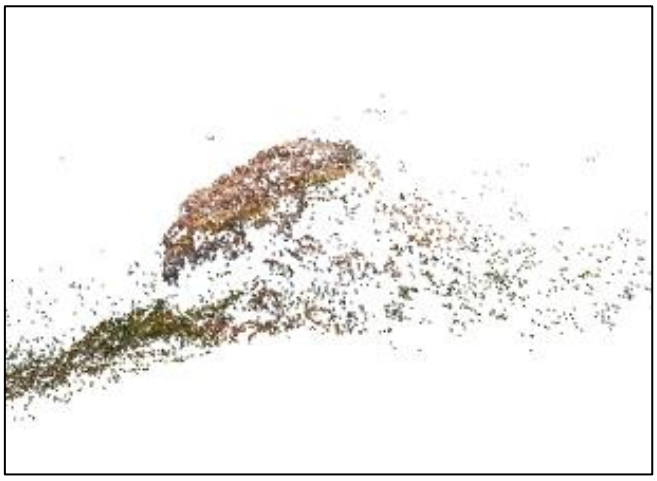

Figure 4.14: Sparse point cloud output for Sigiriya aerial video

Figure 4.13: Dense point cloud output for Sigiriya aerial video (view 1)

Figure 4.12: Dense point cloud output for Sigiriya aerial video (view 2)

Total time consumed	452.6 seconds
Average feature match percentage between frames	41.06
Average correct match percentage (after applying RANSAC) between frames	39.41
Sparse point cloud density	14129
Dense point cloud density	309158

Table 4.8: Performance and outcome measures with Sigiriya aerial video

4.3 Summary

Evaluation of the results with the proposed prototype for different datasets has been presented in this chapter. Results will be evaluated for Fountain and Hertzjesu datasets since ground truth models are available for those datasets. Dense point cloud output results for different aerial videos also presented with the quantitative performance and outcome measures including time consumption, average feature match percentage, average correct match percentage, and point cloud densities.

Final output 3D models has been evaluated against ground truth models which are Lidar generated models by calculating the Hausdorff distance. Output 3D models have been compared with the output of the VisualSfM reconstruction tool. Results show that the implemented prototype outperforms the VisualSfM tool.

The extracted frames using the implemented frame selection algorithm and their quantitative performance and outcome measures for the generated dense point clouds have also been presented. Improved feature matching algorithm with the underline assumption that the subsequent frames in a video have a higher number of matches has been evaluated against traditional method and presented results show that improved algorithm gives better results that are lower Hausdorff distance with ground truth and significant performance improvement in terms of time and memory usage.

CHAPTER 5
CONCLUSION AND FUTURE WORK

5.1 Conclusion

According to the literature and as described in the above chapters, the reconstruction of environments such as buildings and landscapes into 3D graphical representations using 2D images and videos is useful in a variety of applications and is an interesting topic of research in computer vision. This thesis presented a system prototype for offline 3D reconstruction from UAV video feeds with structure from motion (SfM) method. The approach presented is considered significant challenges including selection and processing of video frames from a video with a large number of frames, obtaining robust feature matches by removing noise, and efficient reconstruction with improved feature matching techniques. The work presented novel approaches for video frame selection and feature matching for video frames.

The introduced novel approach for the frame selection algorithm from the video was achieved by measuring the transformation between frames and by selecting based on a certain threshold value. When compared with some of the previous studies (D. Lapandic et al., 2017; J. Ke et al., 2020) which extract frames at specific intervals, implemented frame selection algorithm in this work is more effective as different videos consist of different frame rates.

An Improved and novel feature matching algorithm was implemented by considering consecutive frames have higher matchings hence for a selected frame, instead of obtaining matches against all other frames, it matches only a predefined number of subsequent frames. Therefore, the underline assumption is the input is to be a video feed and not a set of unordered images. In the literature review, it has been identified that the outcome is heavily dependent on feature matchings between views. Implemented feature matching stage in the proposed prototype also consists of multiple filtering to remove noise to obtain robust matches. When compared with some of the previous works (Chen et al., 2018; J. Hlubik et al., 2018; Yuan et al., 2018) which are implemented to obtain matches against all other frames, the evaluation results show that the improved feature matching algorithm has a significant performance improvement in terms of time and memory usage as well as the increased final output model accuracy.

Final output 3D models have been evaluated against ground truth models by calculating the Hausdorff distance. Output 3D models have been compared with the VisualSfM which is an existing 3D reconstruction tool. Results show that the implemented prototype outperforms the

VisualSfM tool. Furthermore, image-based 3D reconstruction is a cost-effective solution compared to Lidar-generated 3D models. It can conclude that the proposed solution is capable of producing a reconstruction that is an effective representation of the environment.

5.2 Future Work

The study in this thesis was carried out for a video input received from a UAV and there are some extensions available to incorporate into the implemented prototype for further improvements. UAVs are capable of retrieving GPS positions with time stamps. Those GPS data can be used to generate a geo-tagged point cloud which can have significant importance for some applications. There can appear holes in the dense point cloud due to the inefficient feature points and matches between frames. That can be improved with an implementation of a mesh repairing algorithm by filling holes in the final mesh. Implemented video frame selection algorithm only considers translation between frames, It can be further improved by using the rotation. Another suggestion to improve the reconstruction process hence to process a large amount of data is to use GPU processing for CPU-intensive stages in the reconstruction pipeline.

REFERENCES

Ajayi, O., 2020. Performance Analysis of Selected Feature Descriptors Used for Automatic Image Registration. *Isprs - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B3-2020, 559–566. https://doi.org/10.5194/isprs-archives-XLIII-B3-2020-559-2020

Chen, X., Wu, Q., Wang, S., 2018. Research on 3D Reconstruction Based on Multiple Views. In: *2018 13th International Conference on Computer Science & Education (ICCSE)*, pp. 1–5.

D. Lapandic, J. Velagic, H. Balta, 2017. Framework for automated reconstruction of 3D model from multiple 2D aerial images, in: *2017 International Symposium ELMAR*, pp. 173–176. https://doi.org/10.23919/ELMAR.2017.8124461

Daftry, S., Hoppe, C., Bischof, H., 2015. Building with drones: Accurate 3D facade reconstruction using MAVs, in: *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3487–3494. https://doi.org/10.1109/ICRA.2015.7139681

Gallaway, D., 2018. 3D Modeling of Ultra-High-Resolution UAV Imagery using Low-Cost Photogrammetric Software and Structure from Motion (PhD Thesis). University of North Carolina at Greensboro, USA.

Gallup, D., 2011. Efficient 3d Reconstruction of Large-Scale Urban Environments from Street-Level Video (PhD Thesis). University of North Carolina at Chapel Hill, USA.

Govender, N., 2009. Evaluation of feature detection algorithms for structure from motion. Council for Scientific and Industrial Research, Pretoria, Technical Report.

J. Hlubik, P. Kamencay, R. Hudec, M. Benco, P. Sykora, 2018. Advanced point cloud estimation based on multiple view geometry, in: *2018 28th International Conference Radioelektronika (RADIOELEKTRONIKA)*, pp. 1–5. https://doi.org/10.1109/RADIOELEK.2018.8376366

J. Ke, A. J. Watras, J. -J. Kim, H. Liu, H. Jiang, Y. H. Hu, 2020. Towards Real-Time, Multi-View Video Stereopsis, in: *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1638–1642. https://doi.org/10.1109/ICASSP40776.2020.9054391

Javadnejad, F., 2018. Small Unmanned Aircraft Systems (UAS) for Engineering Inspections and Geospatial Mapping.

K. Luo, T. Guan, L. Ju, Y. Wang, Z. Chen, Y. Luo, 2020. Attention-Aware Multi-View Stereo, in: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1587–1596. https://doi.org/10.1109/CVPR42600.2020.00166

Kumar, P., 2018. Online 3D Reconstruction and Ground Segmentation using Drone based Long Baseline Stereo Vision System (Masters Thesis). Virginia Polytechnic Institute and State University, USA.

Ladikos, S., 2011. Real-time multi-view 3D reconstruction for interventional environments (Dissertation). Technical University of Munich.

Li, J., 2010. 3D Modeling Using Multi-View Images (Masters Thesis). Arizona State University.

Ling, L., 2013. Dense Real-time 3D Reconstruction from Multiple Images (PhD Thesis). RMIT University.

Martell, A.A., 2017. Benchmarking structure from motion algorithms with video footage taken from a drone against laser-scanner generated 3D models (Masters Thesis). Luleå University of Technology.

Mouats, T., Aouf, N., Nam, D., Vidas, S., 2018. Performance Evaluation of Feature Detectors and Descriptors Beyond the Visible. Journal of Intelligent & Robotic Systems 92, 33–63. https://doi.org/10.1007/s10846-017-0762-8

Pepe, M., Costantino, D., 2020. UAV Photogrammetry and 3D Modelling of Complex Architecture for Maintenance Purposes: the Case Study of the Masonry Bridge on the Sele River, Italy. Periodica Polytechnica Civil Engineering. https://doi.org/10.3311/PPci.16398

Pollefeys, M., Nistér, D., Frahm, J.-M., Akbarzadeh, A., Mordohai, P., Clipp, B., 2008. Detailed Real-Time Urban 3D Reconstruction from Video. *International Journal of Computer Vision 78*, 143–167. https://doi.org/10.1007/s11263-007-0086-4

Schöning, J., Heidemann, G., 2015. Evaluation of multi-view 3D reconstruction software. https://doi.org/10.1007/978-3-319-23117-4_39

Xiao, Z., Wang, R., Lin, J., Zhang, W., 2020. Outcrop-scale Yardang Geometric Analysis using SfM-MVS Point Clouds in Hami Area, NW China.

Y. Xie, Y. Li, Y. Qi, 2019. A Data-Driven Method for Intrinsic Decomposition of 3D City Reconstruction Scene, in: *2019 International Conference on Virtual Reality and Visualization (ICVRV)*, pp. 87–92. https://doi.org/10.1109/ICVRV47840.2019.00023

Yu, J., Park, C., 2016. 3D structure reconstruction from aerial imagery, in: *2016 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia)*, pp. 1–2. https://doi.org/10.1109/ICCE-Asia.2016.7804771

Yuan, Y., Ding, Y., Zhao, L., Lv, L., 2018. An Improved Method of 3D Scene Reconstruction Based on SfM, in: *2018 3rd International Conference on Robotics and Automation Engineering (ICRAE)*, pp. 228–232. https://doi.org/10.1109/ICRAE.2018.8586689

Zheng, E., 2016. Toward 3D Reconstruction of Static and Dynamic Objects (Dissertation). University of North Carolina.