Statistical Methods in Photogrammetry and Image-Lidar Fusion

by

Kyle Andrew Holland

A dissertation submitted in partial satisfaction of the requirements for the degree of

Doctor of Philosophy

in

Environmental Science, Policy and Management

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Gregory S. Biging, Chair
Professor Peng Gong
Professor John Radke

Fall 2013
Abstract

Statistical Methods in Photogrammetry and Image-Lidar Fusion

by

Kyle Andrew Holland

Doctor of Philosophy in Environmental Science, Policy, and Management

University of California, Berkeley

Professor Gregory S. Biging, Chair

A primary application of photogrammetry is to process, discriminate and classify optical imagery into 3D objects. Light detection and ranging (lidar) is an active sensor that measures the surfaces of 3D objects as discrete points in a point cloud. From the perspective of 3D objects in a common scene, point clouds and photogrammetry are related. Through this relationship there are numerous photogrammetric applications of lidar.

This dissertation is concerned with three specific applications of lidar: modeling radiometric properties as a basis for comparison with imagery, estimating camera pose and data fusion. It is partial to the problems of object discrimination and classification, and presents solutions in a statistical context.

A reflectance image is derived from reflectance, shadow and projection models. The reflectance image model is applied to compare the point cloud and imagery. The collinear
equations of imaging are reparameterized as an object to image space transformation and estimated using maximum likelihood. Reflectance images are applied to quantify errors in this transformation across multiple images and to study the convergence properties of estimates. Finally, the process of image-lidar fusion is discussed in the context of uncertainty and probability. An estimator is specified for image-lidar fusion, derived from a generalized theory of the process. The estimator is shown to be unbiased and relatively efficient compared to the sample mean.
# Table of Contents

- List of Figures ........................................................................................................ iv
- List of Tables ........................................................................................................ vii
- List of Equations ..................................................................................................... viii
- Acknowledgement .................................................................................................... xiii

Chapter One: Introduction ...................................................................................... 1
  - 1.1 Background and Motivation ........................................................................... 3
    - 1.1.1 Modern Photogrammetry ....................................................................... 4
    - 1.1.2 The Point Cloud ..................................................................................... 7
  - 1.2 Contribution .................................................................................................... 9
  - 1.3 Structure of Thesis ....................................................................................... 10
    - 1.3.1 Notation ............................................................................................... 12

Chapter Two: Reflectance Image Model from Lidar Data ....................................... 13
  - 2.1 Overview ...................................................................................................... 14
  - 2.2 Reflectance Model ....................................................................................... 16
    - 2.2.1 Parameters ........................................................................................... 19
      - 2.2.1.1 Reflectance ..................................................................................... 19
    - 2.2.2 Time Integral Applications ..................................................................... 20
  - 2.3 Shadow Model ............................................................................................. 21
    - 2.3.1 Parameters ........................................................................................... 24
  - 2.4 Image Model .................................................................................................. 26
  - 2.5 Constructing the Reflectance Image .............................................................. 29
  - 2.6 Application ................................................................................................... 30
    - 2.6.1 Study Site ............................................................................................. 30
    - 2.6.2 Data ....................................................................................................... 30
    - 2.6.3 Modeling Reflectance .......................................................................... 31
    - 2.6.4 Modeling Shadow ................................................................................. 32
  - 2.7 Results ......................................................................................................... 35
    - 2.7.1 Accuracy Assessment ............................................................................ 36
  - 2.8 Discussion ..................................................................................................... 39
    - 2.8.1 Contribution ......................................................................................... 41

Chapter Three: Image Ray Solutions to Camera Pose ............................................. 42
  - 3.1 Overview ...................................................................................................... 43
  - 3.2 Object to Image Space Transformation .......................................................... 45
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.3.2</td>
<td>Related Work</td>
<td>109</td>
</tr>
<tr>
<td>4.4</td>
<td>Theory</td>
<td>112</td>
</tr>
<tr>
<td>4.4.1</td>
<td>Sources of Uncertainty as Events</td>
<td>113</td>
</tr>
<tr>
<td>4.4.2</td>
<td>Fusion Probability</td>
<td>117</td>
</tr>
<tr>
<td>4.5</td>
<td>Probability Fusion Model</td>
<td>124</td>
</tr>
<tr>
<td>4.5.1</td>
<td>Pose Probability</td>
<td>126</td>
</tr>
<tr>
<td>4.5.2</td>
<td>Attribute Probability</td>
<td>127</td>
</tr>
<tr>
<td>4.5.3</td>
<td>Transformation Probability</td>
<td>127</td>
</tr>
<tr>
<td>4.5.3.1</td>
<td>Optical Depth Covariance Ratio</td>
<td>132</td>
</tr>
<tr>
<td>4.5.4</td>
<td>Reduction Probability</td>
<td>134</td>
</tr>
<tr>
<td>4.5.5</td>
<td>Statistical Properties</td>
<td>141</td>
</tr>
<tr>
<td>4.5.5.1</td>
<td>Bias</td>
<td>141</td>
</tr>
<tr>
<td>4.5.5.2</td>
<td>Standard Error</td>
<td>142</td>
</tr>
<tr>
<td>4.5.5.3</td>
<td>Relative Efficiency</td>
<td>142</td>
</tr>
<tr>
<td>4.6</td>
<td>Simulation</td>
<td>144</td>
</tr>
<tr>
<td>4.6.1</td>
<td>Approximate Density Function for Fusion Transformation</td>
<td>144</td>
</tr>
<tr>
<td>4.6.2</td>
<td>Methods</td>
<td>148</td>
</tr>
<tr>
<td>4.6.3</td>
<td>Results</td>
<td>151</td>
</tr>
<tr>
<td>4.7</td>
<td>Discussion</td>
<td>154</td>
</tr>
<tr>
<td>4.7.1</td>
<td>Contribution</td>
<td>156</td>
</tr>
<tr>
<td>5.1</td>
<td>Future Works</td>
<td>159</td>
</tr>
<tr>
<td>References</td>
<td></td>
<td>160</td>
</tr>
<tr>
<td>Appendix A</td>
<td>Convergence Paths for Camera Pose Estimation</td>
<td>172</td>
</tr>
<tr>
<td>Appendix B</td>
<td>Key to Variables</td>
<td>188</td>
</tr>
</tbody>
</table>
List of Figures

Figure 1: A DSM raster of objects in a scene (a) viewed in the 2D orthographic plane and (b) in
the full 3D space (low z-values in red relative to high z-values in green). ........................................ 6
Figure 2: A point cloud of objects in a 3D scene (low z-values in red relative to high z-values in
green). ........................................................................................................................................ 7
Figure 3: Relationship of solar position parameters (N is north). .................................................... 18
Figure 4: Relationship of surface slope angle and aspect relative to the normal vector (a) of the
surface (b) (N is north). .................................................................................................................. 18
Figure 5: The solar ray (dotted line, N is north). ............................................................................ 22
Figure 6: Relationship of pi and pu to the solar ray (N is north). .................................................... 23
Figure 7: The effect of the direction vector on the orientation of the focal plane in object space
(N is north). Given a fixed focal length, the focal plane without rotation is (a) and a focal plane
with rotation is (b) relative to the origin. .......................................................................................... 27
Figure 8: The focal plane depicted with square pixels showing the image principle point as (a), a
coordinate i, j as (b) and a coordinate with additive lens distortion δ as (c). ................................. 27
Figure 9: Aerial image (left) and enhanced image (right) obtained from the first principle
component ........................................................................................................................................... 31
Figure 10: Modeled reflectance for the DSM at the given time and location of the aerial image
(north is up) ......................................................................................................................................... 32
Figure 11: The pseudo-reflectance map in grayscale, the combined shadow map and reflectance
models (north is up). ......................................................................................................................... 34
Figure 12: Modeled reflectance values (left) compared to α values (right) in grayscale (north is
up). .................................................................................................................................................... 34
Figure 13: Enhanced image (left) compared to modeled reflectance image (right) in grayscale. 35
Figure 14: Identified tie points (+) overlaid on the original aerial image. ..................................... 37
Figure 15: Identified tie points (+) overlaid on the enhanced aerial image. ................................. 37
Figure 16: Identified tie points (+) overlaid on the DSM presented in grayscale (north is up,
darker values are lower elevation than lighter values). ............................................................... 38
Figure 17: Geometry of the image ray and image plane in the object space where α is yaw, β is
pitch and γ is roll. The principle point is depicted as (a), image frame on the image plane as (b),
principle axis as (c) and arbitrary image ray as (d). ................................................................. 45
Figure 18: Geometry of image frame in the camera frame. The principle point is depicted as (a),
image as (b) and principle axis as (c). .................................................................
Figure 19: Geometry of the image frame as it relates to (28). The space spanning all $i, j$ is the image plane.................................................................47
Figure 20: Histogram of vector normed difference in pixels on the horizontal axis (see Table 1 for variable declarations, $n=434$)....................................................................................................62
Figure 21: Histograms of bivariate difference in pixels on the horizontal axis (see Table 1 for variable declarations, $n=434$). ........................................................................................................63
Figure 22: Covariate plots showing general trends in the bivariate error in pixels on the horizontal and vertical axes (see Table 1 for variable declarations). ........................................66
Figure 23: Quantile plots of residuals showing heavy tails at the extremes in pixels on the horizontal and vertical axes (see Table 1 for variable declarations). ........................................66
Figure 24: Convergence in $\mathbf{C}$ by iteration. .........................................................................................85
Figure 25: Histograms of estimated camera pose parameters by bootstrap with $n=24$ ($X$, $Y$ and $Z$ in meters; Yaw, Pitch and Roll in degrees; dashed lines indicate true values). ..................91
Figure 26: Histograms of estimated camera pose parameters by bootstrap with $n=48$ ($X$, $Y$ and $Z$ in meters; Yaw, Pitch and Roll in degrees; dashed lines indicate true values). ..................92
Figure 27: Histograms of estimated camera pose parameters by bootstrap with $n=64$ ($X$, $Y$ and $Z$ in meters; Yaw, Pitch and Roll in degrees; dashed lines indicate true values). ..................93
Figure 28: Potential occlusion of the (a) image ray passing through a point $p$ in the two-dimensional object subspace (empty dots represent points in the cloud, arbitrary units on axes). ........................................................................................................106
Figure 29: Uncertainty region about the (a) image ray passing through a point $p$ in the two-dimensional object subspace (empty dots represent points in the cloud, dotted lines are bounds on the uncertainty region, arbitrary units on axes)........................................................................................................107
Figure 30: Multiple image rays (a-c) to a point $p$ in the two-dimensional object subspace (empty dots represent points in the cloud, arbitrary units on axes). ..................................................108
Figure 31: Directed Acyclic Graph (DAG) of events corresponding to sources of uncertainty in the case of a single sampling unit........................................................................................................117
Figure 32: Directed Acyclic Graph (DAG) of events corresponding to sources of uncertainty in the case of duplicity: (a) $K \times m$ pixels in image frames and (b) $K$ image frames. ........................................................................117
Figure 33: Effect of power parameter on probability weights by increasing power (a-d). .......122
Figure 34: Directed Acyclic Graph (DAG) of the fusion probability where shaded nodes represent observed events: (a) $K \times m$ pixels in image frames and (b) $K$ image frames. .......................126
Figure 35: Probability levels of an example distribution on $si$ in an image frame relative to pixels (+). .........................................................................................................................128
Figure 36: Example probability levels as a function of distance variance by increasing optical depth (a-d) in image frames relative to $si$ (+). .................................................................131
Figure 37: Geometry of lidar point occlusion in object space.............................................................136
Figure A.1: Convergence path of lambda in X-Y subspace for n=400: (a) starting point, (b) point of convergence, X denotes true value. ................................................................. 172
Figure A.2: Convergence path of lambda in X-Y subspace for n=80: (a) starting point, (b) point of convergence, X denotes true value. ........................................................................ 173
Figure A.3: Convergence path of lambda in X-Y subspace for n=40: (a) starting point, (b) point of convergence, X denotes true value. ........................................................................ 174
Figure A.4: Convergence path of lambda in X-Y subspace for n=8: (a) starting point, (b) point of convergence, X denotes true value. ........................................................................ 175
Figure A.5: Convergence path of lambda in X-Z subspace for n=400: (a) starting point, (b) point of convergence, X denotes true value. ................................................................. 176
Figure A.41: Convergence path of lambda in X-Z subspace for n=80: (a) starting point, (b) point of convergence, X denotes true value. ................................................................. 177
Figure A.6: Convergence path of lambda in X-Z subspace for n=40: (a) starting point, (b) point of convergence, X denotes true value. ........................................................................ 178
Figure A.7: Convergence path of lambda in X-Z subspace for n=8: (a) starting point, (b) point of convergence, X denotes true value. ........................................................................ 179
Figure A.42: Convergence path of lambda in Y-Z subspace for n=8: (a) starting point, (b) point of convergence, X denotes true value. ........................................................................ 180
Figure A.8: Convergence path of lambda in Y-Z subspace for n=40: (a) starting point, (b) point of convergence, X denotes true value. ........................................................................ 181
Figure A.9: Convergence path of lambda in Y-Z subspace for n=80: (a) starting point, (b) point of convergence, X denotes true value. ........................................................................ 182
Figure A.10: Convergence path of lambda in Y-Z subspace for n=400: (a) starting point, (b) point of convergence, X denotes true value. ........................................................................ 183
Figure A.11: Convergence in lambda by iteration for n=8. ................................................................. 184
Figure A.12: Convergence in lambda by iteration for n=40 ................................................................. 185
Figure A.13: Convergence in lambda by iteration for n=80 ................................................................. 186
Figure A.14: Convergence in lambda by iteration for n=400. ................................................................. 187
List of Tables

Table 1: Variables of the dataset used in exploratory data analysis. ................................................. 59
Table 2: Variable statistics of the dataset used in exploratory data analysis. ................................. 61
Table 3: Statistics of the effects matrix $\mathbf{B}$ ($^+$ indicates significance at the 95% confidence level). ................................................................................................................................. 65
Table 4: Statistics of the covariance matrix $\Sigma$ .................................................................................. 65
Table 5: True parameter values and estimates based on varying sample sizes (estimate error of the true known parameter in meters or degrees, and as percentage of true parameter). ............... 83
Table 6: Parameter estimates of the effects matrix and observed error ........................................... 84
Table 7: Sample sizes and estimates of camera pose parameters for camera frames in the Yellowstone data (error relative to GPS/IMU measurement in meters or degrees). .................. 88
Table 8: Parameter estimates of the effects matrix and their differences from those in Table 3 ($^+$ difference between parameter estimates and estimates in Table 3). ........................................ 89
Table 9: Parameter estimates of the model for the concentration parameter $\kappa$ as a function of $\boldsymbol{\Sigma}\theta$ ........................................................................................................................................ 151
List of Equations

(1) ..................................................................................................................... 16
(2) ..................................................................................................................... 17
(3) ..................................................................................................................... 17
(4) ..................................................................................................................... 17
(5) ..................................................................................................................... 17
(6) ..................................................................................................................... 19
(7) ..................................................................................................................... 20
(8) ..................................................................................................................... 20
(9) ..................................................................................................................... 21
(10) .................................................................................................................. 22
(11) .................................................................................................................. 23
(12) .................................................................................................................. 23
(13) .................................................................................................................. 24
(14) .................................................................................................................. 24
(15) .................................................................................................................. 24
(16) .................................................................................................................. 26
(17) .................................................................................................................. 26
(18) .................................................................................................................. 28
(19) .................................................................................................................. 28
(20) .................................................................................................................. 29
(21) .................................................................................................................. 33
(22) .................................................................................................................. 33
(23) .................................................................................................................. 35
(24) .................................................................................................................. 40
(25) .................................................................................................................. 40
(26) .................................................................................................................. 40
(27) .................................................................................................................. 46
(28) .................................................................................................................. 47
(29) .................................................................................................................. 47
(30) .................................................................................................................. 48
(31) .................................................................................................................. 48
(32) .................................................................................................................. 48
(33) .................................................................................................................. 49
Initially, my intention was to contribute theory and application on the classification of light detection and ranging (lidar) data. As part of this endeavor, I studied the extensive literature on photogrammetry where I discovered its complements to lidar. I also uncovered a literal paucity on the integration of these rich and somewhat dissimilar data. Subsequently, my research shifted to fill this void and to reexamine the assumptions of some established photogrammetric techniques using lidar.

This research was complex and it took a long time to assemble in this dissertation. During this time I was fortunate to have the terrific support of several people whom deserve special acknowledgement. Foremost, I express the deepest gratitude for my spouse. Kelly Holland sacrificed many weekends, nights and holidays separated from me during my writing. She provided endless encouragement and praise.

I especially thank Professor Greg Biging as my advisor for his suggestions and accommodations to this work. His constructive and thoughtful advice has shaped my learning over the six years of my tenure at Berkeley. I also thank Professor David Brillinger for feeding my interest in applied statistics. His lectures inspired me to think critically about statistical problem-solving and that linear algebra really isn’t that hard.

I thank my entire committee for their oversight on this thesis: Professors Peng Gong and John Radke for their points of direction.

Finally, I thank my colleagues at ecoPartners for affording me the time to complete this thesis and for carrying the business during my periodic absence. I thank Kelly Holland, Zach Barbane and Melanie Jonas for their assistance collecting data.
Chapter One: Introduction

Photogrammetry is a long-standing and useful technology with many apparent applications while light detection and ranging (lidar) is a relatively new technology with many real and forthcoming applications. Both technologies are sensor-based and involve measurements of objects in scenes: traditional photogrammetry with passive optical cameras and laser scanning with point clouds. Many applications of these technologies are concerned with the discrimination and measurement of objects in scenes.

A primary task of photogrammetry is to identify and quantify the geometric relationships between multiple camera frames and an absolute, world coordinate system. The camera frame is a local coordinate system in that of the world, defined by camera-external and internal geometric parameters. The difficulty in this task is finding the parameters that define the transformation of the camera frame relative to the world given an image of the world in a 2D projection. In a minority of cases, some of these parameters are measured during imaging. This problem is referred to as the 2D perspective projection problem (Haralick et al. 1989).

Considering this fundamental task, laser scanning provides valuable and abundant data to solve the 2D perspective projection problem. However to effectively solve this problem, a transformation must be specified and the geometric parameters of the transformation must be known. To leverage lidar in the solution to this problem, the radiometric properties of optical imagery must be estimated for the point cloud as a basis for comparison between these two types of sensor data.

Upon defining the transformation and estimating its parameters from lidar, the information in imagery and the point cloud can be integrated to improve object discrimination. The problem of integrating these data is called image-lidar fusion. Applications of object discrimination from fusion data have varying requirements of accuracy and precision. The accuracy and precision of fusion data is affected by uncertainties in the transformation, among other sources. Therefore
it is important to quantify these sources of uncertainty to minimize uncertainty in the fusion data and to perform inference on fusion data.

This dissertation is concerned with identifying and estimating the geometric relationships of camera imagery using point clouds, estimating radiometry from point clouds as a basis of similarity and generalizing image-lidar fusion in the context of accuracy and precision. Chapter Two proposes the radiometric image model to compare the point cloud with camera imagery. Chapter Three builds upon the model to solve the 2D perspective point problem using the point cloud in a statistical manner. Finally, Chapter Four explores the sources of uncertainty in image-lidar fusion and proposes the probability fusion model as efficient estimator of fusion data as a specific case in a general probabilistic context of the theory of image-lidar fusion.

This dissertation also contributes to several associated topics: time-integral applications of radiometric models from point clouds, maximum likelihood estimators (MLEs) of camera pose and the relationship of prior probabilities to sample-based estimation. Lastly, this dissertation identifies several important areas for further research, including initialization conditions and expectation steps for MLEs, among others.
1.1 Background and Motivation

Photogrammetric methods and technologies have steadily developed to a state where they now overlap with lidar technology. Historically, photography was first aided by the stereoscope in 1838 and subsequently introduced to literature by Meydenbauer in 1867 (Ghosh & Durer 1951). The mathematical geometries of internal and external camera parameters were formalized in analytical photogrammetry by von Gruber (1924) early in the 20th Century and later by Church (1934). During this time, statistical methods in aerotriangulation were developed to estimate external parameters using ground control points, overlapping fields of view and an absolute coordinate system of the world. Early applications of these methods were numerous (Miller 1957; Allam 1978). However, these methods require a sufficient quantity of control point data which can be challenging to acquire under cost constraints on fieldwork. This is especially the case when control points are difficult to identify in imagery or to measure in the field; for example, in dense vegetation or featureless desert land.

Herein lies the value of lidar data to photogrammetry; the point cloud inherently provides the necessary control and at sufficient quantity to solve for the parameters. Difficulties in identifying suitable control points are ameliorated by the coverage of the point cloud; unique features on the surfaces of dense vegetation or objects can be used as control, as measured by the point cloud. However, camera imagery and point clouds are fundamentally different measurements. The camera measures radiation while the laser scanner measures the physical surface of objects. To automatically identity control, the point cloud must be augmented with radiometric or pseudo-radiometric information.

The combination of optical imagery and point clouds provides a terrific amount of information about objects. The fusion of these data integrates the spectral and physical (and in some cases temporal) dimensions of objects into a higher dimensional space. However, the integration of these data is difficult because image-lidar fusion is confounded by overlapping transformations, occlusions and uncertainties in measurements. To effectively integrate these data requires the
enumeration and description of these factors as they relate to each other in a theoretical framework.

1.1.1 Modern Photogrammetry

Many applications of photogrammetry benefit from the integration of camera imagery and point clouds. Modern photogrammetry is the processing of optical imagery into secondary products for specific application: the digital terrain model (DTM), digital surface model (DSM), scene reconstruction and visualization (E. P. Baltsavias 1999). It has evolved to incorporate inertial navigation systems and the Global Positioning System (Schwartz et al. 1993) and has been applied to measurement problems in natural resource monitoring (Gong et al. 2002; Yongwei Sheng et al. 2001).

Other applications of photogrammetry include object classification and thematic mapping, two important and evolving fields of research. Object classification and thematic mapping are discrimination problems (T. Hastie et al. 2003).

These two applications particularly benefit from integration with point clouds. In the last decade, attention has been directed toward this class of problem in applications ranging from resource assessment (Schneevogt & Linden 2010; Caravaggi & Giada 2005; Torres & Salcedo 2010) to medicine (Li Wang et al. 2009; Lu et al. 2012). In terms of accuracy, integrating physical dimensions from the DTM and DSMs improves object discrimination and classification of optical imagery relative to imagery alone (Zhang et al. 2010).

Compared to the traditional DTM and DSM, the point cloud provided more information about objects. These products are regularly-spaced samples of an orthographic projection of the physical surfaces of objects (see Figure 1). Information about the sides of objects is lost in this projection as is information about the fine detail of surfaces is lost through raster sampling on a coarse grid. In contrast to the DTM and DSM, the point cloud contains point samples of the sides of objects and retains the fine detail of surfaces (see Figure 2). Therefore, it is reasonable to expect that in some applications, integrating the point cloud with optical imagery will provide improved results relative to the DSM or DTM.
Figure 1: A DSM raster of objects in a scene (a) viewed in the 2D orthographic plane and (b) in the full 3D space (low z-values in red relative to high z-values in green).
Figure 2: A point cloud of objects in a 3D scene (low z-values in red relative to high z-values in green).

1.1.2 The Point Cloud

Lidar is the technology employed by a laser scanner to measure a point cloud. As illustrated in Figure 2, the point cloud is measurements of the physical surfaces of objects. The laser scanner emits a laser pulse that is reflected from the surfaces of objects. In the case of discrete return lidar, one of two methods is used to calculate the coordinate of reflection. The time-of-flight method calculates the coordinate by measuring the pulse return time to the sensor and the pulse direction from the sensor. The phase-shift method calculates the coordinate by measuring the phase of a returning pulse and the pulse direction from the sensor (Guili Liu et al. 2006; Yoon & K. Park 2005). The sensor emits a continuous signal with continuously changing phase so that the phase shift determines the distance. For a fixed period of scanning time, the phase-shift method usually produces more measurements than the time-of-flight method.
The laser scanner is typically mounted to a platform that is fixed or mobile in the world. Fixed platforms are stationary in the world coordinate system while mobile platforms are not. Examples of mobile platforms include aircraft, automobiles and robots. Directly from the laser scanner, the point cloud is expressed in a coordinate system with an origin relative to the emitter, not the world coordinate system absolutely relative to a single datum. If the platform is stationary, the coordinates must be transformed to the world coordinate system by an affine transformation. When the platform is mobile, the transformation is conditional on the location and orientation of the sensor over time. This transformation is informed by precision measurements of these parameters simultaneous with scanning.
1.2 Contribution

Considering the important applications of point clouds to photogrammetry and the value of integrating these data, this dissertation is concerned with generalizing and solving these problems in a statistical manner. Further, this dissertation is concerned with identifying important assumptions in existing solutions to these problems and suggesting alternatives by relaxing or reformulating assumptions in a statistical manner. Specifically, this dissertation addresses four objectives:

1. To identify a model that estimates radiometric properties of camera imagery from the point cloud as a basis for comparison of the point cloud with imagery;
2. To derive statistical estimators of external camera parameters and control point error from lidar data using the comparative basis;
3. To propose a theory for image-lidar fusion that generalizes existing, specific applications and accounts for various sources of uncertainty.
4. To derive a statistical and efficient estimator for image-lidar fusion that accounts for the various sources of uncertainty.

Specific contributions are noted in sections 2.8.1, 3.7.1 and 4.7.1.
1.3 Structure of Thesis

This dissertation is organized into five chapters as follows:

- **Chapter One: Introduction**
  
  This chapter provides background, motivation and contribution to the dissertation. Applications and sensors of optical imagery and point clouds are discussed.

- **Chapter Two: Reflectance Image Model from Lidar Data**
  
  This chapter provides an overview of the spectral and camera geometry of imagery relative to the lidar point cloud and reformulates the radiative transfer model for the purpose of comparison of these data types in the image frame. Three fundamental models are derived for reflectance, shadow and the projection of this information onto a reflectance image along with simplifying assumptions. The models are developed to attain contextual information from the point cloud and are applied to identify tie points between the pixels of an image and points in the cloud using common features in imagery and the reflectance image, as initial step toward to image-lidar fusion. Other applications of these models are to estimating fine-scale light availability, plant community dynamics and image orthorectification.

- **Chapter Three: Image Ray Solutions to Camera Pose**
  
  This chapter redefines the image ray as a transformation from the object space to the image space, initially defined in Chapter Two. The transformation is shown to be a function of the camera projective center and the flight trajectory parameters. Together, these parameters define the camera pose with Euclidian geometry. The transformation is compared to those presented in the literature with respect to mathematical framework and parametric assumptions. The chapter identifies that the image ray is embedded in existing methods to solve for camera pose. An exploration of data from a study site in Yellowstone National Park infers that a bivariate normal distribution can be used to model the transformation and to estimate camera pose in the presence of error
solace across multiple camera frames. Based on literature review, the formal identity of the image ray in existing methods and exploratory data analysis, a new method for estimating camera pose is introduced. This method transparently accounts for the sources of uncertainty in the image ray using maximum likelihood estimators and an iterative, pseudo-expectation-maximization (EM) method. The method is applied to analyze some convergence properties of the estimators and empirical distributions on camera pose parameters. As specified by the likelihood function, a scale factor is applied to the rotation matrix and optical depth. The presented solution to the scale factor necessarily constrains the rotation matrix and is a novel approach. Applications of these new estimators are far-reaching and of particular importance to photogrammetry for remote-sensing and image-lidar fusion.

- **Chapter Four: Probability Model for Image-Lidar Fusion**

  This chapter presents the problem of image-lidar fusion, establishes a theoretical framework for the problem to maximize information in fusion data and provides measures on uncertainty, and proposes the probability fusion model as a solution. Fusion reduction and duplicity are identified as the cruxes of the problem; reduction is the occlusion of the point cloud by itself while duplicity is caused by overlapping camera images. The solution is unified by the relative geometry of camera images and the point cloud, derived as a statistical estimator. The estimator is shown to be unbiased with efficiency equivalent to or greater than the sample mean. Likewise the standard error of the probability fusion model is derived. An optical depth covariance (ODC) ratio is defined to measure the effect of the variances of the point cloud and camera projective center on the transformation probability, a component of the model. Finally, resampling methods are used to estimate an approximate density function for the transformation probability under the condition that the ODC ratio is small. Several corollaries are identified, including generalities about the Horvitz-Thompson weights and Inverse Distance Weighting (IDW) as they relate to prior probabilities on estimates.
• **Chapter Five: Conclusion**

This chapter discusses the relation of prior chapters to each other and identifies critical areas for further research.

1.3.1 **Notation**

This thesis expresses mathematical relationships in the notation of linear algebra and statistics in standard texts. The scalar is denoted by italics, vector by bold italics, matrix by bold uppercase and the set by an uppercase script. In some equations, an index is used to indicate the relationship of a quantity to a vector, matrix or set. An index to a matrix is a pair in which the first is to a row and the second to a column. The space of all real numbers is expressed as \( \mathbb{R}^d \) over dimensions \( d \) or \( \mathbb{R}^{r \times c} \) over dimensions \( r \) in the row space of a matrix and dimensions \( c \) in the column space of a matrix.

A statistic is expressed as an estimate denoted by a “hat,” as a log-likelihood denoted by a lowercase script \( \ell \) or a specific function in italics. A probability is denoted by an uppercase \( P \) and is defined on the outcome of an event. Events are expressed in shorthand by referencing an observed quantity. The condition of an event is also expressed in shorthand to the right of the event. Expectation of a random variable is denoted by and uppercase \( E \).

Where possible, notation is maintained between chapters. In some cases, a variable may be used in different contexts between chapters. A key to all variables is provided as Appendix B.
Chapter Two: Reflectance Image Model from Lidar Data

This chapter provides an overview of the spectral and camera geometry of imagery relative to the lidar point cloud and reformulates the radiative transfer model for the purpose of comparison of these data types in the image frame. Three fundamental models are derived for reflectance, shadow and the projection of this information onto a reflectance image along with simplifying assumptions. The models are developed to attain contextual information from the point cloud and are applied to identify tie points between the pixels of an image and points in the cloud using common features in imagery and the reflectance image, as initial step toward to image-lidar fusion. Other applications of these models are to estimating fine-scale light availability, plant community dynamics and image orthorectification.
2.1 Overview

Imagery of the Earth’s surface contains useful contextual information: location, size, texture and shape. This information is discerned from the digital image using its pixels which contain values that are samples of radiation incident to a sensor in some frequency of the radiometric spectrum (Jenson 2005). Associations between neighboring pixels and pixel values can be used to identify and classify features in imagery (Gong & Howarth 1992; Moller-Jensen 1997).

Passive cameras rely on solar radiation to illuminate features: radiation from the sun is transmitted through the atmosphere onto the surface of features, a portion of which is then reflected to the camera. However, the atmosphere affects transmission by scattering and absorbing radiation before and after radiation reaches the surface. These effects contribute to diffuse irradiance while the portion of radiation that is ultimately transmitted directly to the feature’s surface contributes to direct irradiance. This incident radiation is absorbed by the feature or transmitted through the feature, and the remainder is reflected back into the atmosphere and ultimately captured by the imaging sensor (J. V. Dave 1981).

Radiative transfer models attempt to capture these complex interactions of energy with the atmosphere and Earth’s surface (Chandrasekhar 1960). A good model specifies direct and diffuse irradiance, and by accounting for the effects of surface transmission, absorption and reflectance. Radiative transfer models can be used to normalize the pixel values of passively-sensed imagery to improve feature discrimination.

A component of the radiative transfer model is the topographic radiation model for direct surface irradiance. This type of model has been applied to answer numerous scientific questions involving imagery and was first introduced by Woodham and Gray (1987). The discrimination of land use and land class from satellite imagery relies on adjusting pixel values for topographic reflectance (Proy et al. 1989; Meyer et al. 1993; D. Gu & Gillespie 1998). Direct irradiance is an important to ecosystem production. Topographic radiation models have been
used to estimate primary production in ecosystems from imagery (Running & Coughlan 1988; Goward & Dye 1987; Sabetraftar et al. 2011; Y. Ryu et al. 2011).

This chapter is concerned with developing generalized models of direct irradiance, shadow and Lambertian reflectance from the surface of features using point cloud data, as they would appear on a camera sensor. Point cloud data obtained from an airborne lidar sensor and a topographic radiation model are used to estimate incident irradiance as a fraction of solar radiance. Camera and flight geometry are used to project this information onto a focal plane. The resultant product is a reflectance image that resembles a black-and-white photograph when a grayscale color model is applied to it.

Three general models are defined between points of the cloud and the image: reflectance, shadow and image models. These models are generalized because they are not defined exclusively for raster data spaces, but rather the true vector space of the entire point cloud. This formulation allows for several extensions of these models which are discussed in section 2.8, namely probabilistic estimates of reflectance and shadow within the point cloud, and time integrals.

These models can be applied to a high-density point cloud of vegetation structure to study the micro-scale effects of light availability in the understory. Or, these models can be applied to estimate aircraft flight trajectory parameters for imagery from the point cloud in the absence of direct inertial measurements.
2.2 Reflectance Model

The problem of the reflectance model is to define the fraction of incident irradiance reflected from a surface to the camera sensor using the point cloud. The reflectance model is based on two simplifying assumptions: diffuse irradiance is negligible and the surface is Lambertian. Illumination by a Lambertian surface obeys Lambert’s cosine law (J. V. Dave 1981).

The effects of diffuse irradiance can be reasonably ignored in the visible spectrum if atmospheric conditions minimize scattering and absorption by particulates and gases. At this minimum and the solar zenith, diffuse irradiance is relatively small compared to direct irradiance (J. V. Dave 1981; Weiss & Norman 1985). Though, these conditions are never exact; the atmosphere will always cause some diffusion, however the effect of this diffusion is ignored in formulation of the reflectance model.

The reflectivity of the feature surface must be assumed in the absence of such information. As the point cloud does not directly provide irradiance values, nor does it provide direct information about reflectivity. Thus, the reflectance model assumes a Lambertian surface for simplicity. Although not discussed in this chapter, the reflectance model can be modified to account for both diffuse irradiance and surface reflectivity.

The derivation of the reflectance model is based on a model of direct solar irradiance proposed by Olyphant (1986). This model specifics that direct irradiance $D$ is the of the product of the cosine of the angle of incidence $\psi$ and direct normal irradiance $I$.

Reflectance is the fraction of incident irradiance that is not transmitted through the surface or absorbed by the surface. Ignoring diffuse irradiance reflected from neighboring objects, the law of the conservation of energy implies that

$$D = I(\alpha + \gamma + \rho) \cos \psi$$

(1)
where $\alpha$ is fractional absorption, $\gamma$ is fractional transmission and $\rho$ is fractional reflectance. This formulation leads to reflected irradiance as

$$ R = I \rho \cos \psi. \quad (2) $$

From (1) and (2) the reflectance model is derived as

$$ r = \frac{R}{I} = \frac{I \rho \cos \psi}{I} = \rho \cos \psi \quad (3) $$

which is a function of reflectance and the angle of incidence. The later of these is defined by Robinson (1966) as

$$ \psi = \cos^{-1} \left[ \cos \phi \cos S + \sin \phi \sin S \cos (\theta - A) \right] \quad (4) $$

where $\phi$ is the solar zenith angle, $\theta$ is the solar azimuth, $S$ is the surface slope angle and $A$ is the surface slope azimuth (collectively called the solar position parameters). The relationships of the solar position parameters are illustrated in Figure 3 while the relationships of the slope angle and aspect relative to the surface normal vector are illustrated in Figure 4. Hence the reference model can be formulated in terms of these solar position parameters as

$$ r = \rho \cos \phi \cos S + \rho \sin \phi \sin S \cos (\theta - A). \quad (5) $$

In this form, all parameters are known or can be estimated from the lidar point cloud.
Figure 3: Relationship of solar position parameters (N is north).

Figure 4: Relationship of surface slope angle and aspect relative to the normal vector (a) of the surface (b) (N is north).
2.2.1 Parameters

The reflectance model can be applied to estimate reflectance at any location in the point cloud or on the surface of the point cloud. Given a location on a plane \((x, y)\), its elevation value \(z\) and intensity return value can be interpolated from the point cloud, expressed as a set of points \(\mathcal{P} = \{p: p^T = (x, y, z)\}\). The intensity return is a measure of the energy of the laser pulse reflected from the surface back to the lidar sensor. Using a set of points \(\mathcal{P}_p \subset \mathcal{P}\) near a point of interest \(p'\), the slope angle \(S\) and aspect \(A\) at this point can be estimated by methods such as those in Fleming and Hoffer (1979) or Hodgson (1998). The theoretical bound on the slope angle and aspect are 0-90° and 0-365°, respectively.

To estimate slope angle and aspect on the surface of the point cloud, it may be necessary to interpolate a digital surface model (DSM) from the point cloud. Methods for constructing the DSM are well known (Maas and Vosselman 1999; Murakami et al. 1999; Priestnall, Jaafar, and Duncan 2000).

The solar position parameters can be estimated using \((x, y)\) at a known Julian date by one of many well-known models such as those derived by Meeus (1991).

2.2.1.1 Reflectance

The parameter \(\rho\) can be estimated at point \(p'\) using an interpolated intensity value at this point and the known upper bound on intensity values from the lidar sensor output. Consider the special case when \(\cos \psi = 1\) and when reflected irradiance and direct normal irradiance are known, then (1) and (2) imply that

\[
\rho = \frac{R}{D}.
\]  

An estimate of this parameter can be obtained from the point cloud. The typical lidar sensor operates in the near-infrared portion of the radiometric spectrum generating intensity returns from the surface of features (Lefsky et al. 2002). When operated at low altitude relative to the surface, the laser path minimizes atmospheric attenuation in the data as the optical depth is
small (J. V. Dave 1964). Also, the generalized surface is orthogonal to the laser pulse at a narrow field of view. Relative to an upper bound $\zeta_D$ on the intensity return and the intensity return $\zeta_R$ as measured by the lidar sensor at each point, the approximation is

$$\hat{\rho} = \frac{\zeta_R}{\zeta_D}$$

2.2.2 Time Integral Applications

The reflectance model can be generalized to study the cumulative effects of reflectance over time. By allowing the solar position parameters to vary with time $t$, the integral reflectance over the time period from $t_1$ to $t_2$ can be expressed as

$$\frac{1}{c} \int_{t_1}^{t_2} \rho \cos \phi(t) \cos S + \rho \sin \phi(t) \sin S \cos[\theta(t) - A] \, dt$$

Where $c$ is some normalizing constant. This integral could be computed using finite approximations in the time domain via a simulation model. Although this chapter does not elaborate on this integral, it has important applications (see section 2.8).
2.3 Shadow Model

The reflectance model defines the fraction of incident irradiance reflected from a surface to the camera sensor. However in the event that a point of interest \( \mathbf{p}^T = (x, y, z) \) is occluded from the solar ray by some obstruction, such as a tree or hillside, direct irradiance is affected by the obstruction’s shadow.

Shadow models have been proposed using the raster DSM and ray tracing (Y. Li et al. 2005). However this model can be formulated in the vector space which has important applications to modeling shadow within the point cloud and over time (see section 2.8).

This formulation of the shadow model seeks to determine whether a point \( \mathbf{p} \) representing a point in the point cloud or on the surface of a feature is occluded from the solar disk by another point \( \mathbf{p}_i^T = (x_i, y_i, z_i) \). On the surface of the point cloud, this occurs when \( \mathbf{p}_i \) is above the solar ray to \( \mathbf{p} \) and within some horizontal tolerance \( \tau \) of the solar ray.

The solar ray \( \mathbf{u} \in \mathbb{R}^3 \) is a line passing through \( \mathbf{p} \) in the direction of the sun. In vector form, the solar ray can be written as

\[
\begin{align*}
\mathbf{u} &= \mathbf{p} + t \\
&= \mathbf{p} + t \\
&= \mathbf{p} + t \mathbf{s}_r,
\end{align*}
\]

where \( \mathbf{s}_r \) corresponds to the direction vector, \( \mathbf{p} \) to the position vector and \( t \) to some arbitrary scalar as defined by the implicit form of a vector equation (Lay 2002). The solar ray is illustrated
in Figure 5 showing that \( s_r \) is simply a unit vector that has been rotated about the z-axis and x-axis by \( \frac{\pi}{4} - \theta \) and \( \phi \), respectively.

![Figure 5: The solar ray (dotted line, N is north).](image)

To determine whether another point \( p_i \) is above the solar ray, consider the point \( p_{\text{top}} = (x_u, y_u, z_u) \) in the solar ray that is closest to \( p_i \). The shortest distance between \( p_i \) to \( p_{\text{top}} \) is a line orthogonal to the solar ray such that the dot product

\[
(p_i - p_{\text{top}}) \cdot s_r = 0
\]  

(10)

as illustrated in Figure 6.
One component of the model is to find $z_u$ such that $\eta_1 = z_i > z_u$ can be logically tested. Because $\mathbf{p}_u$ is in the solar ray, (10) can be written as

$$0 = (\mathbf{p}_i - \mathbf{p} - t\mathbf{s}) \cdot \mathbf{s}_r. \quad (11)$$

Finding $z_u$ is then equivalent to finding $t$, the solution to which is

$$t = \frac{1}{\|\mathbf{s}\|^2} \left[ (x_i - x) \cos \left( \frac{\pi}{4} - \theta \right) \sin \phi + (y_i - y) \sin \left( \frac{\pi}{4} - \theta \right) \sin \phi \right. \left. + (z_i - z) \cos \phi \right]. \quad (12)$$

Then from (9) and (12), the solution to $z_u$ becomes
24

\[ z_u = z - \frac{\sin \phi}{\| \mathbf{s}_r \|} \left[ (x_i - x) \cos \left( \frac{\pi}{4} - \theta \right) \sin \phi + (y_i - y) \sin \left( \frac{\pi}{4} - \theta \right) \sin \phi ight. \\
\left. + (z_i - z) \cos \phi \right] \tag{13} \]

Hence the solution to \( z_u \) can be used to test \( \eta_1 \).

Another component is to test whether \( \mathbf{p}_i \) is within some horizontal tolerance \( \tau \) of the solar ray in the x-y plane. This is easily accomplished by

\[ \eta_2 = \left\| \left( \mathbf{p}_i - \mathbf{p} + t \mathbf{s}_r \right)^T \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right\| < \tau \tag{14} \]

which is the norm of the vector difference between \( \mathbf{p}_i \) and \( \mathbf{p}_u \) projected onto the x-y plane.

Finally we arrive at the shadow model which is specified as the product of two logical tests

\[ \eta = 1 - \eta_1 \eta_2 \tag{15} \]

where each logical test is one if true or zero if false. If the shadow model yields a value of zero, the point \( \mathbf{p} \) is in the shadow of \( \mathbf{p}_i \) given the parameter values. By generalizing the assumptions of this formulation, the shadow could be defined without logical tests as discussed in section 2.8.

2.3.1 Parameters

Parameter considerations for the shadow model are similar to the reflectance model. However, the shadow model requires the additional specification of a horizontal tolerance \( \tau \). For most applications of the shadow model, the point of interest \( \mathbf{p} \) presumably resides on some continuous surface, the point itself is merely a single coordinate of an infinite number of coordinates that represent the entire surface. Therefore, the surface may obstruct the solar ray
even if there is no \( p_i \) immediately above the solar ray as determined by \( \eta_1 \). Such a case naturally leads to the specification of \( \tau \), as \( p_i \) may not be directly above the solar ray.

The value of \( \tau \) is effectively the radius of a disk in the x-y plane about \( p_i \). If the solar ray through \( p \) crosses this disk when projected onto the x-y plane, then \( \eta_2 \) is true. Hence the value of \( \tau \) should be selected based on the spacing between points in the x-y subspace of the point cloud. Large values will increase the size of the disk relative to small values, resulting in more shadow.
2.4 Image Model

The image model defines the geometry of the camera sensor relative to features on the Earth’s surface. Each pixel in the image is a projection of reflectance and shadow from a portion of a feature. The image model is necessary to transform the reflectance and shadow models to a reflectance image.

An image ray \( \mathbf{v} \in \mathbb{R}^3 \) is a line passing through a point \((i, j)\) in an image. This point corresponds with the location of a pixel on the sensor. In vector form, the image ray can be written as

\[
\mathbf{v} = \mathbf{\lambda} + \tau \mathbf{R} \begin{bmatrix} i - i_0 + \delta(i) \\ j - j_0 + \delta(j) \\ -f \end{bmatrix} = \mathbf{\lambda} + \tau \mathbf{Rg}
\]

where \( \mathbf{\lambda} \) is the projective center of the camera in object space, \( f \) is the camera focal length, \( \mathbf{R} \) is a rotation matrix, \((i_0, j_0)\) is the image principle point and \( \delta \) is camera lens distortion (see Figures 7 and 8). The direction vector is defined by \( \mathbf{Rg} \) and \( \mathbf{\lambda} \) is the position vector in the vector equation of the line. The quantity \( \tau \) is some arbitrary scalar (sometimes referred to as optical depth) that determines the vector location along the line. From (16), the image ray can be expressed as

\[
\mathbf{v} - \mathbf{\lambda} = \tau \mathbf{Rg}. \tag{17}
\]
Figure 7: The effect of the direction vector on the orientation of the focal plane in object space (N is north). Given a fixed focal length, the focal plane without rotation is (a) and a focal plane with rotation is (b) relative to the origin.

Figure 8: The focal plane depicted with square pixels showing the image principle point as (a), a coordinate \((i, j)\) as (b) and a coordinate with additive lens distortion \(\delta\) as (c).
The image model is the distance \( d \) between any point \( p_i \in \mathbb{R}^3 \) and the image ray passing through a point \((i, j)\) in an image. Reflectance at \((i, j)\) can be estimated from the point cloud using weights that are a function of the image model. One possible weight function is defined in section 2.5.

As derived by Lay (2002), the distance \( d \) between any \( p_i \) and a line passing through \( \lambda \) and \( v \), both in the image ray, is

\[
d = \frac{\|(p_i - \lambda) \times (p_i - v)\|}{\|v - \lambda\|} \tag{18}
\]

Then from (16) and (17), the distance can be expressed as

\[
d = \frac{\|(p_i - \lambda) \times (p_i - \lambda - Rg)\|}{\|Rg\|} \tag{19}
\]

for arbitrary \( t = 1 \) which yields the image model.

Camera lens distortion should also be considered, the function for which varies by camera lens. If this distortion is not included in \( g \), then the coordinates of common features in the reflectance image and the image of interest will not match. For some applications, this is trivial. For instance, using the reflectance image to find tie points between an image and features in a point cloud can be accomplished without considering lens distortion.
2.5 Constructing the Reflectance Image

The reflectance image can be assembled a variety of ways using the reflectance, shadow and image models. One possible method to construct the reflectance image uses a kernel function \( k(d) \) on the point cloud such that

\[
    b_{i,j} = \sum_{h \in \mathcal{P}} k(d_h) r_h \eta_h
\]  

(20)

where \( b_{i,j} \in [0,1] \) is a value of pixel \( (i,j) \) in the reflectance image and \( \mathcal{P} = \{1,2,3 \ldots,n\} \) is an index set to the point cloud (other variables previously defined). Clearly this method is computationally expensive on the order of \( O(n^2) \) for each \( b_{i,j} \) which is not desirable from a computation standpoint.

The expense of this method across all \( b_{i,j} \) can be reduced by subsetting \( \mathcal{P} \). This subset could be defined by the extreme corners of the image, based on \( d_h \) for each of these four \( (i,j) \) and taking only those points within some threshold distance. The expense can be further reduced by tiling the point cloud into still smaller subsets corresponding to blocks of pixels. Tiling is complementary to parallel computing.
2.6 Application

The reflectance, shadow and image models were demonstrated at a study site in Mendocino County, California. The objective of this application was to find tie points between features in a color aerial image and the lidar point cloud. Such an application could be useful to image orthorectification or lidar data fusion. This was achieved by modeling reflectance and shadow from the point cloud and interpolating results from these models onto a reflectance image using the image model. The resultant reflectance image achieves this objective as indicated by an accuracy assessment of tie points.

2.6.1 Study Site

The Angelo Coast Range Reserve is on the South Fork of the Eel River with a total size of 3,166 ha (39° 43’ 45”N, 123° 38’ 40”W). A portion of the Reserve that is less than twenty hectares was used for the study site. The Reserve ranges in elevation from 378 to 1,290 meters above sea level and experiences about 216 cm of annual average precipitation. These attributes make for diverse vegetation cover ranging from old growth forest to meadow.

2.6.2 Data

One color image was obtained in March 2005 using a Kodak MS4100 camera mounted onboard an aircraft as a benchmark to compare model results (1920 by 1080 pixels in dimension). Aircraft gyroscopic and positioning information were also obtained for the image as parameters to the image model. Point cloud data were acquired separately from the image using an Optech Aquarius scanner flown by the National Center for Airborne Laser Mapping. The median distance between lidar points in the x-y plane was 0.72 meters. The first principle component of the aerial image was used as an enhanced image to compare the modeled reflectance image to the color image (see Figure 9).
2.6.3 Modeling Reflectance

The reflectance model was applied to an interpolated DSM generated from the point cloud data using computer algorithms to implement the reflectance, shadow and image models. The resolution of the DSM was one meter and the elevation values for the DSM cells were calculated as the maximum z-value from the nearest points to the center of the cell in the x-y plane. The intensity return in the near infrared was interpolated similarly to the elevation values for each DSM cell. Using the interpolated z-values of cells, the slope angle $S$ and aspect $A$ of each cell were calculated using the methods of Hodgson (1998) as implemented by ESRI ArcMap.

The solar position parameters were estimated using the National Oceanic and Atmospheric Administration Solar Calculator provided the latitude and longitude of the Reserve, and the Julian date when the image was obtained (US Department of Commerce n.d.).
The reflectance parameter for each cell of the DSM was estimated by dividing its intensity return by the value of 255, the upper bound on the intensity return provided by the lidar sensor. The reflectance model was applied to each cell of the DSM, presented in grayscale as Figure 10.

![Modeled reflectance for the DSM at the given time and location of the aerial image (north is up).](image)

### 2.6.4 Modeling Shadow

The shadow model was applied to each cell of the DSM to generate a shadow map. The same solar position parameters used in the reflectance model were used in the shadow model. The threshold parameter \( \tau \) was set to the median point density in the x-y plane.

For any particular cell located at \( \mathbf{p}^T = (x, y, z) \), it was determined to be in a shadow by the product of shadow models as
\[ \bar{\eta} = \prod_{i \in \mathcal{D}} \eta_i \]  

where \( \mathcal{D} \) was the set of all DSM cells and \( i \) was an index to a cell.

The shadow map and the reflectance model were applied to each cell of the DSM to construct a pseudo-reflectance map as

\[ \alpha = r^{1-\bar{\eta}} - (1 - \omega)\bar{\eta}r^{\bar{\eta}} \]  

which yielded reflectance when not shadowed and a fraction \( 1 - \omega \) of reflectance when shadowed. The parameter \( \omega \) was defined as the fraction of diffuse irradiance to total irradiance reaching the surface. Although (22) was defined specifically for this study under the assumption that \( \alpha, \omega, r \) and \( \bar{\eta} \) are confined to the unit interval, it can be extended to other applications where reflectance is the product of diffuse and direct irradiance on a surface. For this study, a value of \( \omega = \frac{1}{3} \) was selected because it produced contextual information similar to that observed in the enhanced image. If (22) is adopted in other applications, it may suitable to conduct a sensitivity analysis of the effect of \( \omega \) on study results. For this study, it was only necessary to select \( \omega \) to give reasonable results for tie-point matching (see section 2.7.1). The pseudo-reflectance map is presented in grayscale as Figure 11 and compared to modeled reflectance in Figure 12.
Figure 11: The pseudo-reflectance map in grayscale, the combined shadow map and reflectance models (north is up).

Figure 12: Modeled reflectance values (left) compared to $\alpha$ values (right) in grayscale (north is up).
2.7 Results

The pseudo-reflectance map was a combination of modeled reflectance and shadow values for each DSM cell. Each element of pixel \((i, j)\) in the reflectance image was calculated using this map, the image model and a nearest neighbor kernel \(k_{NN}(d)\) as

\[
b_{i,j} = \sum_{h \in D} k_{NN}(d_h) \alpha
\]

where \(h\) was an index to a DSM cell. Figure 13 compares the enhanced image to the reflectance image. The image coordinates of some features toward the edges of the images differ because camera lens distortion was not considered in the calculation of image model for \(d_h\).

Figure 13: Enhanced image (left) compared to modeled reflectance image (right) in grayscale.
2.7.1 Accuracy Assessment

An accuracy assessment is used to infer if the modeled reflectance image can be used to generate tie points between the aerial image and the point cloud. Because each pixel of the reflectance image can be traced to a specific cell in the DSM and the DSM is generated from the point cloud, it suffices to assess tie points between the reflectance image and the enhanced image.

Identical grids of 2205 points were overlaid on the enhanced aerial image and the reflectance image. On the enhanced image, a window the size of 21 square-pixels was fixed about each grid point. On the reflectance image, a window of the same dimension was initiated about each grid point and allowed to move up to thirty pixels in any direction. The covariance between the pixel values in the window on the enhanced image and the window on the reflectance image was calculated for each grid point as described by Brunelli and Poggio (1999). If the covariance value was less than 0.5 then the window on the reflectance image was moved progressively outward until a covariance value of 0.5 was achieved. If a covariance value of 0.5 was obtained, the grid point was recorded as a tie point. If upon exhausting the search for a particular grid point a desirable covariance value was not obtained, the grid point was not recorded as a tie point.

A large number of tie points between the color image and the point cloud were identified using the reflectance image. A total of 1003 or approximately 45% of grid points were recorded as tie points (see Figures 14 and 15).

Notable in Figure 16, tie points were identified where features were easily discriminated. More tie points were identified at the edges of trees compared to the open field which lacked discernible features. Clearly the results of the accuracy assessment were dependent on the features in the scene and not solely on the results of the reflectance model.
Figure 14: Identified tie points (+) overlaid on the original aerial image.

Figure 15: Identified tie points (+) overlaid on the enhanced aerial image.
Figure 16: Identified tie points (+) overlaid on the DSM presented in grayscale (north is up, darker values are lower elevation than lighter values).
2.8 Discussion

Three models of reflectance, shadow and image geometry are derived in this chapter and together provide a basis for comparison of the point cloud to optical imagery. These models differ from existing radiative transfer models in purpose and application. Firstly, the model is defined for the vector point cloud as opposed to the raster DEM or DSM. Second, the models are projected onto the image plane to produce a reflectance image. This step augments existing methods to the problem of image-lidar fusion where pixel values are fused to the point cloud in the image frame. Finally, the traditional radiative transfer model is simplified for this purpose by focusing on those components of the model that most affect contextual information (location, size, texture and shape) and less on the other components that do not greatly affect this information. Assumptions about diffusion and reflectivity are imposed to simplify the model without the noticeable loss of contextual information.

This chapter also provides an application of these models to identify tie points using feature information in aerial imagery between imagery and the lidar point cloud as a first step toward image-lidar fusion. Because these models are generalized to points in space and points in images, there are many additional applications for these models. The simplifying assumptions on which some of these models are derived can be modified to broaden their applicability.

One example is to the Lambertian assumption of the reflectance model. This common assumption was tested on forest vegetation by Smith, Lin, and Ranson (1980) to infer that reflectivity was not Lambertian. Then to account for non-uniform reflectivity of surfaces, it could be modeled as function of surface thickness and texture. The point cloud provides information about surface thickness and texture in the arrangements of points relative to each other on the exterior of the cloud. Metrics derived from these arrangements could be used to predict how reflectivity changes with the look angle to the camera sensor.

The tolerance parameter $\tau$ can be further modified as well. As defined by (14), this parameter effectively controls the radius of a disk in the $x$-$y$ plane about a point to determine whether that point obstructs the solar ray. When the parameter is defined in $\mathbb{R}^3$ it then controls the shape
dimensions of an ovoid about the point rather than a disk. This definition for \( \tau \) could then account for differences in point density in the x, y and z-dimensions of the point cloud. Further, the tolerance parameter in \( \mathbb{R}^3 \) can be defined by a probability distribution, the parameters of which can be estimated from the lidar point cloud.

The implicit form of the shadow model is reduced under this revision to the tolerance parameter by eliminating the product of logical tests. For instance, assume that \( \tau \sim N(\mu_\tau, \Sigma_\tau) \) in which case the shadow model can be formulated as the probability that the point \( p_i \) obstructs the solar ray to point \( p \)

\[
P(p_i - p_u | \bar{\mu}_\tau, \bar{\Sigma}_\tau)
\]

where \( p_u \) is defined by (10). Then, the shadow model simplifies to

\[
\frac{1}{\eta} = (2\pi)^3 |\Sigma_\tau| \frac{1}{2} e^\frac{1}{2}(p_i-p_u-\bar{\mu}_\tau)^\top \Sigma_\tau^{-1}(p_i-p_u-\bar{\mu}_\tau)
\]

without logical tests. Then given the entire point cloud denoted by \( \mathcal{P} \), the probability that \( p \) is in a shadow becomes

\[
P(p | \mathcal{P}, \bar{\mu}_\tau, \bar{\Sigma}_\tau) = c \sum_{i \in \mathcal{P}} P(p_i - p_u | \bar{\mu}_\tau, \bar{\Sigma}_\tau)
\]

where \( c \) is some normalizing constant. This summation of the shadow model is defined as a summation similar to (20).

Upon specifying a distribution on \( \tau \), the reflectance and shadow models can be applied to solve problems residing within the point cloud in a probabilistic way. An applicable problem is to predict the response of understory plants to light availability over time. Approximately 80\% of incident radiation in the visual spectrum is absorbed by the overstory canopy (Roderick et al.)
The remainder penetrates the canopy as diffuse irradiance and sun flecks. Sun flecks are of particular importance to understory plants (Chazdon & Pearcy 1991) and an application of these models to study the effects of sun flecks on understory plants could yield new ecological findings.

Terrestrial lidar scanners have the capability to acquire high-resolution data of individual leaves and branches. Time integral applications of the reflectance model as presented in (8) and the reformulated shadow model in (26) can be used to construct three dimensional probability maps of light availability in the understory using the high-resolution data. Because the point cloud can be manipulated to simulate differing over story conditions, the effects of changes in light availability can be modeled to predict plant response.

The image model has a variety of applications besides image-lidar fusion. Using a sufficient number of tie points, the trajectory and camera parameters can be estimated for images lacking this auxiliary metadata. These parameters can then be used to produce orthoimagery, eliminating the need for expensive inertial measurement units onboard aircraft imaging platforms. Elements of this approach form the basis of trajectory fusion of time series imagery, discussed in chapter three.

**2.8.1 Contribution**

This chapter provides the following unique contributions:

- Repurposing and simplification of the radiative transfer model to provide contextual information as the basis of comparison between the point cloud and optical imagery;
- Extension of the radiative transfer model from raster data to the vector point cloud; and
- The reflectance image to compare the point cloud and optical imagery directly in the image frame.
Chapter Three: Image Ray Solutions to Camera Pose

This chapter redefines the image ray as a transformation from the object space to the image space, initially defined in Chapter Two. The transformation is shown to be a function of the camera projective center and the flight trajectory parameters. Together, these parameters define the camera pose with Euclidian geometry. The transformation is compared to those presented in the literature with respect to mathematical framework and parametric assumptions. The chapter identifies that the image ray is embedded in existing methods to solve for camera pose. An exploration of data from a study site in Yellowstone National Park infers that a bivariate normal distribution can be used to model the transformation and to estimate camera pose in the presence of error across multiple camera frames. Based on literature review, the formal identity of the image ray in existing methods and exploratory data analysis, a new method for estimating camera pose is introduced. This method transparently accounts for the sources of uncertainty in the image ray using maximum likelihood estimators and an iterative, pseudo-expectation-maximization (EM) method. The method is applied to analyze some convergence properties of the estimators and empirical distributions on camera pose parameters. As specified by the likelihood function, a scale factor is applied to the rotation matrix and optical depth. The presented solution to the scale factor necessarily constrains the rotation matrix and is a novel approach. Applications of these new estimators are far-reaching and of particular importance to photogrammetry for remote-sensing and image-lidar fusion.
3.1 Overview

Camera pose is the absolute location and orientation of a camera at the time of image acquisition. Estimating the location and orientation of the camera frame given an image scene is a fundamental and hard problem in computer vision and photogrammetry. This problem is called Point-n-Perspective (PnP) when the objective is to estimate the relative position between a scene and the frame, and the camera pose problem when the objective is to estimate the absolute position and orientation of the frame from a scene. Solutions to these problems have been derived under varying assumptions and at different levels of complexity. Although the PnP problem is important and similar to the camera pose problem, this chapter focuses on the latter as it relates to determining camera orientation and location in object space using reflectance image models (see Chapter Two).

A good solution to the camera pose problem is important to photogrammetry. In the context of remote sensing, there are at least three prime applications for such estimates: construction of surface models and maps from historical imagery; elimination of onboard Inertial Measurement Units (IMUs) for the rectification of aerial imagery; and precise fusion of images to lidar point clouds. Many other applications exist.

Vast historical images sets have been archived, containing valuable information about the size and location of historical objects. However, accompanying image attributes on camera pose are almost always missing. If the solution provides an accurate estimate of camera pose, then the absolute size and location of objects in historical images can be easily derived. The solution can be applied to an image set to construct historical surface models and maps. If the error in camera pose estimates can be quantified, then the precision of such models and maps can be stated.

The size and cost of IMUs is prohibitive to popularizing modern photogrammetry for remote-sensing. A software solution to camera location and orientation (as opposed to hardware) is an opportunity for economic growth and entrepreneurship. Without the weight of added
hardware, cameras can be mounted on small aircraft with low risk. Without the expense of IMU hardware, photogrammetrists can provide imaging and mapping services at lower cost. Finally, fusing images to lidar data requires accurate estimates of camera pose. At relatively close range with lidar data, small inaccuracies in camera location can result in poor fusion results. When fusing multiple and overlapping images to a single lidar point, it is important that camera pose be accurate to achieve good results.

Several challenges are present in the solution to camera pose from image scenes. An initial challenge is to simply define the problem in a mathematical framework. Generally the literature provide solutions that follow from either distance or transformation-based definitions of the problem (Horaud et al. 1989; Y. Wu & Z. Hu 2005). The mathematical representation of camera orientation is a particular challenge that has been addressed multiple ways including the traditional Euclidian rotation matrix, Rodriguez rotation formula and quaternions because rotations with this later method are quicker to computer than with matrices (Haralick et al. 1989; J. a. Hesch & Roumeliotis 2011; Bill Triggs et al. 2000). For some of these representations, it is necessary to apply constraints in estimation to achieve good estimates of camera orientation (Haralick et al. 1989). Lastly, a final challenge is to define error in the framework. Many solutions are derived from a simplified definition of error which could lead to bias and imprecise estimates of camera pose. The study of such effects has never been published in the mainstream literature.

In this chapter, the camera pose problem is defined as transformation-based, extending the mathematics of the image ray (introduced in Chapter Two) to the object to image space transformation. Because the image ray is defined by a Euclidian rotation matrix, some necessary constraints should be applied in estimation; section 3.5.2.1 introduces a new formulation of these constraints. Further in this chapter, a parametric model is defined and estimators are given in closed form to estimate camera pose. Although many literature solutions fundamentally rely on parametric assumptions, few authors thoroughly consider distributions on error in their formulation of the problem. Such distributions are more fully investigated and modeled in section 3.4.3 using a dataset from Yellowstone National Park.
3.2 Object to Image Space Transformation

The image ray is defined in the object space, a line passing through the projective center of the camera \( \lambda \) and the image plane (See Figure 17). The image ray that is perpendicular to the image plane is known as the camera principle axis. The point at which this line passes through the plane is the principle point. This formulation is equivalent to the projective camera model described in literature.

![Diagram of image ray and image plane in object space](image)

Figure 17: Geometry of the image ray and image plane in the object space where \( \alpha \) is yaw, \( \beta \) is pitch and \( \gamma \) is roll. The principle point is depicted as (a), image frame on the image plane as (b), principle axis as (c) and arbitrary image ray as (d).

In the camera frame, the coordinates of the principle point are \((0,0,-f)\) where the origin of the system is the projective center. The image frame is defined on the image plane where the origin of the image frame coordinate system is \(-\omega = (-i_0, j_0, -f)\) in the camera frame coordinate system (See Figure 18). To estimate camera pose from the image ray, the image ray must be redefined from the image frame in the image plane to the object space.
3.2.1 Definition

The image ray is expressed as a transformation from the camera frame to the object space in Chapter Two, restated in (27) as

\[ \mathbf{v} - \mathbf{\lambda} = \mathbf{\tauRg} \]  (27)

where \( \mathbf{\lambda} \) is the projective center of the camera in object space and \( \mathbf{R} \) is a rotation matrix. The direction vector is defined by \( \mathbf{Rg} \) and \( \mathbf{\lambda} \) is the position vector in the vector equation of the line. The vector \( \mathbf{g} \) is defined as
where \( f \) is the camera focal length, \((i_0, j_0)\) is the image point in the image coordinate system, \( \delta \) is camera lens distortion and \( s \) is the imaging vector, shown in the image frame in Figure 19.

By substituting (28) into (27), the image ray becomes

\[
\mathbf{v} - \lambda = \tau \mathbf{R} (\mathbf{s} - \omega).
\]  

(29)

The solution to the imaging vector is

\[
\mathbf{g} = \begin{bmatrix}
i - i_0 + \delta(i) \\
j - j_0 + \delta(j) \\
-f
\end{bmatrix}
= \begin{bmatrix}
i + \delta(i) \\
j + \delta(j) \\
0
\end{bmatrix} - \begin{bmatrix}
i_0 \\
j_0 \\
f
\end{bmatrix}
= \mathbf{s} - \omega.
\]  

Figure 19: Geometry of the image frame as it relates to (28). The space spanning all \((i, j)\) is the image plane.
where $\mathbf{R}^{-1}$ is the matrix inverse of the rotation matrix. In this form, the imaging vector is a function of the parameter sets $\theta$, $\omega$ and $\lambda$. Note that because $\mathbf{R}$ is symmetric with determinant 1, its inverse is equal to its transpose (see section 3.5.2.1).

### 3.2.2 Derivation

The object to image space transformation seeks to find the imaging vector from $\mathbf{v}$ given the parameter sets. However, because $\mathbf{s}$ is in the image plane, it is not linear in $\mathbf{v}$ and thus the solution must be found in the projection of $\mathbf{g}$ onto the image plane. This problem is equivalent to finding $\tau$.

Let $\mathbf{u} = \mathbf{R}^{-1}(\mathbf{v} - \lambda)$ such that

$$s = \frac{1}{\tau} \mathbf{u} + \omega$$

$$= \frac{1}{\tau} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} + \begin{bmatrix} i_0 \\ j_0 \\ -f \end{bmatrix}$$

which is equivalent to writing the system

$$\begin{cases} i + \delta(i) = \frac{u_1}{\tau} + i_0 \\ j + \delta(j) = \frac{u_2}{\tau} + j_0 \\ 0 = \frac{u_3}{\tau} - f \end{cases}$$

where the solution to $\tau$ is
\[ \tau = \frac{u_3}{f} \]  

(33)

Then given (33), the system becomes

\[
\begin{aligned}
    s_1 &= i + \delta(i) = \frac{fu_1}{u_3} + i_0 \\
    s_2 &= j + \delta(j) = \frac{fu_2}{u_3} + j_0
\end{aligned}
\]  

(34)

in which are the collinear equations of imaging. Thus from (30) and (33), the object to image transformation is finally derived as

\[
s = \frac{f}{u_3} R^{-1}(v - \lambda) + \omega.
\]

(35)

Note that the third element of \( s \) is zero and thus (28) holds:

\[
0 = \frac{f}{u_3} u_3 - f.
\]

(36)

3.2.3 Reparameterization

The object to image space transformation given in (35) can be reparameterized in terms of the vector elements of the rotation matrix and the rotation of the projective center \( \lambda \). This form of the transformation is congruent to the perspective camera model described in literature (Horaud et al. 1989) and simplifies estimation.

The rotation matrix is defined as
\[
\mathbf{R} = \begin{bmatrix}
\cos \alpha \cos \beta & \cos \alpha \sin \beta \sin \gamma - \sin \alpha \cos \gamma & \cos \alpha \sin \beta \cos \gamma + \sin \alpha \sin \gamma \\
\sin \alpha \cos \beta & \sin \alpha \sin \beta \sin \gamma + \cos \alpha \cos \gamma & \sin \alpha \sin \beta \cos \gamma - \cos \alpha \sin \gamma \\
-\sin \beta & \cos \beta \sin \gamma & \cos \beta \cos \gamma
\end{bmatrix}
\] 

(37)

\[
\begin{bmatrix}
a \\
d \\
g
\end{bmatrix}
= \begin{bmatrix}
b & c \\
e & f \\
h & i
\end{bmatrix}
\]

(38)

where \( \alpha \) is yaw, \( \beta \) is pitch and \( \gamma \) is roll (Trainelli & Croce 2004; Slabaugh 2012). The yaw, pitch and roll parameters form the flight trajectory vector \( \mathbf{\theta} \) as described in Chapter Two. Note that the inverse of the rotation matrix is its transpose and defined as

\[
\mathbf{R}^{-1} = \begin{bmatrix}
a & d & g \\
b & e & h \\
c & f & i
\end{bmatrix}
\]

(38)

\[
= \begin{bmatrix}
r_1 \\
r_2 \\
r_3
\end{bmatrix}
\]

and that the rotation matrix must satisfy the following condition

\[
\mathbf{I} = \mathbf{R}^T \mathbf{R}
\]

(39)

where \( \mathbf{I} \) is the identity matrix.

In equation (35), the rotation matrix can be distributed inside the parentheses as

\[
s = \frac{f}{u_3} (\mathbf{R}^{-1} \mathbf{v} - \mathbf{R}^{-1} \lambda) + \mathbf{\omega}
\]

(40)

and by substituting \( \mathbf{\kappa} = \mathbf{R}^{-1} \lambda \) becomes
The reparameterized form of the object to image space transformation is then

\[ s = \frac{f}{u_3} (R^{-1}v - \kappa) + \omega. \]  \hspace{1cm} (41)

The reparameterized form of the object to image space transformation is then

\[ s_1 = f \frac{r_1 v - \kappa_1}{r_3 v - \kappa_3} + i_0 = d_1 + \omega_1 \]  \hspace{1cm} (42)

\[ s_2 = f \frac{r_2 v - \kappa_2}{r_3 v - \kappa_3} + j_0 = d_2 + \omega_2 \]

\[ s_3 = f \frac{r_3 v - \kappa_3}{r_3 v - \kappa_3} - f = d_3 + \omega_3 = 0 \]

\[ s = d + \omega \]

using the vector elements of the rotation matrix as defined by (38) where \( \kappa^T = (\kappa_1, \kappa_2, \kappa_3) \) and \( d^T = (d_1, d_2, f) \).
3.3 Existing Methods

Transformation-based solutions for the Euler angles of the rotation matrix and projective center in the camera pose problems that are widely published and applied. These solutions rely on observing corresponding control points in the object and image space. The problem has been generally solved four ways: directly using the minimal case of three points (Fischler & Bolles 1981), using an over-determined linear system (A. Ansar & K. Daniilidis 2003), iteratively by minimizing a nonlinear least-squares problem (Haralick et al. 1989; Quan & Lan 1999; B. Triggs 1999) and by direct least squares (J. A. Hesch & Roumeliotis 2011). Some of these approaches are sensitive to initialization conditions (Vigueras et al. 2009) and may give multiple or non-optimal solutions (Oberkampf et al. 1996; J. a. Hesch & Roumeliotis 2011). Some authors have presented variations to these approaches to solve for the global optimum using at least six points (Gerald Schweighofer 2008; Lepetit et al. 2008). Pose estimation has been applied to solve numerous problems: architectural measurement, reconstruction of paintings, forensics and traffic accident investigation (Duan et al. 2008; Criminisi 2001).

Perhaps the most cited solution is that provided by Haralick et al. (1989), following from a cost minimization problem. The cost function is specified by a weighted sum of the squared residuals

$$e^2 = \sum_{i=1}^{n} w_i \| R v_i + \kappa - t_i s_i^* \|^2$$

(43)

to account for small observational errors where the weights $w_i$ for each $i = 1, \ldots, n$ identified points are constrained and $s_i^*$ is observed $s$ in the image space.\(^1\) This approach assumes $v$ and the weights are known exactly or can be precisely estimated. If the weights are set to $w_i = 1/n$ then solution is predicated on the assumption that $e^2$ is normally distributed.

\(^1\) Note that this definition is equivalent to that given by Haralick et al. (1989) where $s_i^* = v_n, v_i = y_n, \kappa = t$ and $d_k^m = t_i$. 

52
For robust estimation, the weights are M-estimates determined using the iteratively reweighted least squares (IRLS) method. The M-estimator is implicitly defined as

\[ 0 = \sum_{i=1}^{n} \psi(x_i - \theta) \]  

(44)

where \( x \) is some data, \( \psi \) is a weight function (the derivative of some object function) and \( \theta \) is the weighted mean. Haralick et al. specify the weight function \( \psi \) as Tukey’s bi-weight

\[ w_i = \begin{cases} 
1 - \frac{\|\epsilon_i\|^2}{6s}, & \|\epsilon_i\| \leq 6s; \\
0, & \text{o.w.} 
\end{cases} \]  

(45)

where \( s \) is the median absolute deviation.

Haralick et al. conducted a theoretical experiment by perturbing thousands of transformed point in the three-dimensional camera frame with zero mean Gaussian noise and then projecting these points onto the image plane. No results were given based on real noise in realistic image scenes or for relatively small sample sizes. The distribution of \( \epsilon^2 \) and the i.i.d., zero mean assumptions were not examined.

The effect of estimated weights from small sample sizes in the presence of extreme values has been studied in contexts other than pose estimation. Parameter estimates can be badly affected by IRLS if the underlying distribution is not known (Ryan 2008; Carrol & Ruppert 1988). The consequence of an incorrectly specified model is poor (or even bias) estimates.

Finally, Haralick et al. suggest two iterative methods for estimating \( \tau \) in the 2-D perspective projection to 3-D pose estimation problem. In the first of these methods, its \( k^{th} \) iterate is defined as
\[ \tau^{k+1} = \left( r_3^k \mathbf{v} - \kappa_3^k \right) \frac{\sum_{l=1}^n \| \mathbf{v}_l - \overline{\mathbf{v}} \|^2}{\sum_{l=1}^n \| q_l - \overline{q} \|^2} \] (46)

where \( q = R\mathbf{v} + \kappa \) and overbars indicate the average over all \( n \). In the second, its iterate is defined as

\[ \tau^{k+1} = \frac{s_i^*(r_3^k \mathbf{v} - \kappa_3^k)^T}{s_i^* s_i^{*T}} \] (47)

and it is shown that \( \epsilon^2 \) for the \( k + 1 \) iterate is strictly less than for the \( k^{th} \). Haralick et al. observed parameter convergence under both definitions.
3.4 Exploratory Data Analysis

Chapter Two provides a reflectance image model of direct irradiance, surface reflectance and shadow. The reflectance image model estimates irradiance as it would appear on the camera sensor and resembles a normal grayscale image when a color model is applied to it. The reflectance image provides a unique opportunity study the pure effect of the object to image space transformation on imaging without the presence of camera lens distortion or uncertainties in camera pose. Camera pose can be specified exactly to generate a reflectance image with perfect geometry.

The effect of the transformation on a system’s ability to distinguish unique features in images is quantified as error. Such a system could be traditional photo interpretation or an automated feature recognition algorithm. To estimate camera pose, the coordinates of features must be known in both the object and image spaces. Assuming that we know the coordinates exactly in the object space, then the error is entirely in the image space.

Working in the reflectance image with known camera pose, the error can be quantified and attributed entirely to the transformation without confounding effects.

3.4.1 Study Site

The study site is a square kilometer of Yellowstone National Park along the Lamar River. The area is sparsely populated with trees and other clearly identifiable landmarks for use as control points. The study site contains varying topography from 1,700 to 2,450 meters in elevation. Aerial photography and lidar data were acquired for the study site at separate times.

3.4.2 Yellowstone Data

Imagery was acquired using a Redlake metric frame camera at an average altitude of 3,185 meters. Flight trajectory measurements were obtained for images using aircraft gyroscopic and positioning equipment. Point cloud data were acquired using an Optech Aquarius scanner flown by the National Center for Airborne Laser Mapping.
Data were observed from 31 reflectance image models generated from flight trajectory measurements along an aircraft flight path. The flight path was from north to south over the study site at an altitude between 2500 and 3000 meters. The process described in Chapter Two was applied to generate reflectance image models at 31 exposure points along the flight path from a lidar point cloud covering the study site. The point cloud data were obtained along a different flight path on a different platform. The median density of the point cloud data were 1.8 points per square meter.

The point cloud was interpolated into a Digital Surface Model (DSM) with one-meter square raster cells where each cell value was the maximum elevation of the points that fell within that cell. Missing values were filled in where no points fell within cells using the average of a five-by-five meter square window. The DSM was subsequently used to generate reflectance image models and to identify control points. To identify control points, prominent features in the DSM were mapped in ArcGIS. A total of 164 control points were mapped and labeled with unique numbers.

The exposure points, control points and DSM were printed onto a series of large format maps covering the entire study site. Each of the 31 reflectance image models were printed for comparison to the large format maps. The control points appearing on the large format map were identified and labeled on each printed reflectance image by an independent photo interpreter. The interpreter did not generate the control points or the reflectance image models. The interpreter was trained to recognize features in images corresponding to features on the DSM to identify control points. The interpreted control points were digitized using specially developed software to determine their coordinates in each of the reflectance images. A total of 434 control points were interpreted in the 31 reflectance images. In cases where image scenes overlapped, a single control point was visible in multiple images.

The object to image space transformation was applied to each of the 164 mapped control points from the GIS to obtain their true coordinates in the space of each reflectance image. These coordinates were then paired with their corresponding interpreted coordinates as the
dataset for analysis. For each 434 pairs, the dataset was augmented with several potential covariates. The structure of the dataset is given in Table 1 and described by statistics in Table 2.

To avoid confounding the analysis, lens distortion was not included in the reflectance image models and as a result \( i = i + \delta(i) \).
<table>
<thead>
<tr>
<th>Variable</th>
<th>Covariate</th>
<th>Description (units)</th>
<th>Mathematical Representation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td></td>
<td>A unique label for each data point</td>
<td></td>
</tr>
<tr>
<td>FID</td>
<td></td>
<td>The label of the control point in the GIS</td>
<td></td>
</tr>
<tr>
<td>X</td>
<td></td>
<td>The interpreted $i$ coordinate in the image space (pixels)</td>
<td>$s_1^*$</td>
</tr>
<tr>
<td>Y</td>
<td></td>
<td>The interpreted $j$ coordinate in the image space (pixels)</td>
<td>$s_2^*$</td>
</tr>
<tr>
<td>X0</td>
<td></td>
<td>The transformed $i$ coordinate in the image space (pixels)</td>
<td>$s_1$</td>
</tr>
<tr>
<td>Y0</td>
<td></td>
<td>The transformed $j$ coordinate in the image space (pixels)</td>
<td>$s_2$</td>
</tr>
<tr>
<td>DX</td>
<td></td>
<td>The distance between the transformed and interpreted $i$ coordinates (pixels)</td>
<td>$s_1 - s_1^*$</td>
</tr>
<tr>
<td>DY</td>
<td></td>
<td>The distance between the transformed and interpreted $j$ coordinates (pixels)</td>
<td>$s_2 - s_2^*$</td>
</tr>
<tr>
<td>DVN</td>
<td></td>
<td>The vector normed difference between the transformed and interpreted ($i,j$) coordinates given $\tau$ (pixels)</td>
<td>$\epsilon_{VN}$</td>
</tr>
<tr>
<td>Tau</td>
<td>Yes</td>
<td>Optical depth (meters)</td>
<td>$\tau$</td>
</tr>
<tr>
<td>Yaw</td>
<td>Yes</td>
<td>Rotation parameter (degrees)</td>
<td>$\alpha$</td>
</tr>
<tr>
<td>Pitch</td>
<td>Yes</td>
<td>Rotation parameter (degrees)</td>
<td>$\beta$</td>
</tr>
<tr>
<td>Roll</td>
<td>Yes</td>
<td>Rotation parameter (degrees)</td>
<td>$\gamma$</td>
</tr>
<tr>
<td>DistPPX</td>
<td>Yes</td>
<td>The distance between the principle point and interpreted $i$ coordinates (pixels)</td>
<td>$s_1^* - \omega_1$</td>
</tr>
<tr>
<td>Variable</td>
<td>Yes/No</td>
<td>Description</td>
<td>Formula</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>DistPPY</td>
<td>Yes</td>
<td>The distance between the principle point and interpreted $j$ coordinates (pixels)</td>
<td>$s_2^* - \omega_2$</td>
</tr>
<tr>
<td>AbsDistPPX</td>
<td>Yes</td>
<td>The absolute distance between the principle point and interpreted $i$ coordinates (pixels)</td>
<td>$</td>
</tr>
<tr>
<td>AbsDistPPY</td>
<td>Yes</td>
<td>The absolute distance between the principle point and interpreted $j$ coordinates (pixels)</td>
<td>$</td>
</tr>
<tr>
<td>DistPP</td>
<td>Yes</td>
<td>The absolute distance between the principle point and interpreted $(i, j)$ coordinates (pixels)</td>
<td>$\sqrt{(s_1^* - \omega_1)^2 + (s_2^* - \omega_2)^2}$</td>
</tr>
<tr>
<td>a</td>
<td>Yes</td>
<td>Element of the rotation matrix</td>
<td>$a$</td>
</tr>
<tr>
<td>b</td>
<td>Yes</td>
<td>Element of the rotation matrix</td>
<td>$b$</td>
</tr>
<tr>
<td>c</td>
<td>Yes</td>
<td>Element of the rotation matrix</td>
<td>$c$</td>
</tr>
<tr>
<td>d</td>
<td>Yes</td>
<td>Element of the rotation matrix</td>
<td>$d$</td>
</tr>
<tr>
<td>e</td>
<td>Yes</td>
<td>Element of the rotation matrix</td>
<td>$e$</td>
</tr>
<tr>
<td>f</td>
<td>Yes</td>
<td>Element of the rotation matrix</td>
<td>$f$</td>
</tr>
<tr>
<td>g</td>
<td>Yes</td>
<td>Element of the rotation matrix</td>
<td>$g$</td>
</tr>
<tr>
<td>h</td>
<td>Yes</td>
<td>Element of the rotation matrix</td>
<td>$h$</td>
</tr>
<tr>
<td>i</td>
<td>Yes</td>
<td>Element of the rotation matrix</td>
<td>$i$</td>
</tr>
</tbody>
</table>

Table 1: Variables of the dataset used in exploratory data analysis.
<table>
<thead>
<tr>
<th>Variable (units)</th>
<th>Average</th>
<th>Max</th>
<th>Min</th>
<th>Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>X (pixels)</td>
<td>910.342</td>
<td>1672</td>
<td>177</td>
<td>402.0209</td>
</tr>
<tr>
<td>Y (pixels)</td>
<td>632.4974</td>
<td>1313</td>
<td>19</td>
<td>373.0711</td>
</tr>
<tr>
<td>X0 (pixels)</td>
<td>1075.648</td>
<td>1982.358</td>
<td>206.1299</td>
<td>423.7463</td>
</tr>
<tr>
<td>Y0 (pixels)</td>
<td>633.3043</td>
<td>1317.471</td>
<td>20.06133</td>
<td>375.7791</td>
</tr>
<tr>
<td>DX (pixels)</td>
<td>-3.5E-10</td>
<td>40.4516</td>
<td>-39.893</td>
<td>10.32281</td>
</tr>
<tr>
<td>DY (pixels)</td>
<td>0.806918</td>
<td>19.822</td>
<td>-31.562</td>
<td>6.71155</td>
</tr>
<tr>
<td>DVN (pixels)</td>
<td>9.827564</td>
<td>43.15781</td>
<td>0.407658</td>
<td>7.427677</td>
</tr>
<tr>
<td>Tau (meters)</td>
<td>-42036.5</td>
<td>-40619.8</td>
<td>-43266.8</td>
<td>624.2584</td>
</tr>
<tr>
<td>Yaw (degrees)</td>
<td>189.3024</td>
<td>191.197</td>
<td>187.041</td>
<td>0.651711</td>
</tr>
<tr>
<td>Pitch (degrees)</td>
<td>2.987968</td>
<td>4.4</td>
<td>1.665</td>
<td>0.283466</td>
</tr>
<tr>
<td>Roll (degrees)</td>
<td>-0.4251</td>
<td>2.688</td>
<td>-3.008</td>
<td>1.674106</td>
</tr>
<tr>
<td>DistPPX (pixels)</td>
<td>179.658</td>
<td>913</td>
<td>-582</td>
<td>402.0209</td>
</tr>
<tr>
<td>DistPPY (pixels)</td>
<td>33.00259</td>
<td>646.5</td>
<td>-647.5</td>
<td>373.0711</td>
</tr>
<tr>
<td>AbsDistPPX</td>
<td>385.0052</td>
<td>913</td>
<td>2</td>
<td>212.285</td>
</tr>
<tr>
<td>AbsDistPPY</td>
<td>326.3497</td>
<td>647.5</td>
<td>2.5</td>
<td>182.2582</td>
</tr>
<tr>
<td>DistPP (pixels)</td>
<td>543.5246</td>
<td>1025.31</td>
<td>21.59282</td>
<td>193.3526</td>
</tr>
<tr>
<td>a</td>
<td>-0.98802</td>
<td>-0.9844</td>
<td>-0.99051</td>
<td>0.001733</td>
</tr>
<tr>
<td>b</td>
<td>-0.14915</td>
<td>-0.1322</td>
<td>-0.17209</td>
<td>0.011234</td>
</tr>
<tr>
<td>c</td>
<td>-0.03776</td>
<td>-0.02906</td>
<td>-0.04578</td>
<td>0.004942</td>
</tr>
<tr>
<td>d</td>
<td>0.148295</td>
<td>0.1735</td>
<td>0.130893</td>
<td>0.011966</td>
</tr>
<tr>
<td>e</td>
<td>-0.98814</td>
<td>-0.98404</td>
<td>-0.99077</td>
<td>0.001767</td>
</tr>
<tr>
<td>f</td>
<td>0.024508</td>
<td>0.052435</td>
<td>-0.03963</td>
<td>0.029189</td>
</tr>
</tbody>
</table>
Table 2: Variable statistics of the dataset used in exploratory data analysis.

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>g</td>
<td>-0.04076</td>
<td>-0.02922</td>
<td>-0.04709</td>
<td>0.005083</td>
</tr>
<tr>
<td>h</td>
<td>0.018649</td>
<td>0.045874</td>
<td>-0.04537</td>
<td>0.029454</td>
</tr>
<tr>
<td>i</td>
<td>0.99855</td>
<td>0.999399</td>
<td>0.997858</td>
<td>0.000398</td>
</tr>
</tbody>
</table>

3.4.3 Error Distributions

Error between the interpreted and transformed coordinates was defined two ways: as the vector normed difference and the bivariate difference. The vector normed difference is equivalent to the error defined by Haralick et al. (1989) and is defined as

$$
\epsilon_{VN} = \| q^* + \omega - (R^{-1}v - \kappa + \omega) \|
$$

where \( q^* \) is observed in the three-dimensional camera frame. However, the data were in the two dimensional image plane and not in the camera frame. Following from (30), the relationship between \( q \) and \( s \) on the image plane is

$$
s = \tau^{-1}q + \omega
$$

which implies that

$$
q = \tau(s - \omega).
$$

The vector normed difference then becomes

$$
\epsilon_{VN} = \| \tau(s^* - \omega) - (R^{-1}v - \kappa) \|
$$

in terms of observed \( s^* \) given \( \tau \) which was known exactly (not observed) for each \( s^* \).
The bivariate difference \( \mathbf{e}_{BV} \in \mathbb{R}^2 \) is defined on the image plane as

\[
\mathbf{e}_{BV} = \begin{bmatrix}
  s_1^* - d_1 + \omega_1 \\
  s_2^* - d_2 + \omega_2 
\end{bmatrix}
\]

(52)

where \( s_1^* \) is observed \( s_1 \) from (42).

**3.4.3.1 Vector Normed Difference**

A histogram of \( \mathbf{e}_{VN} \) is presented in Figure 20 and shows a skewed, non-normal distribution given the observed data. The Shapiro-Wilk test statistic for normality infers that \( \mathbf{e}_{VN} \) is not normally distributed \( (p < 0.001) \). A dispersion test infers that the mean and variance of \( \mathbf{e}_{VN} \) are not statistically different \( (p < 0.001) \) which may imply that \( \mathbf{e}_{VN} \) follows a special form of the Gamma or Poisson distributions. As described in section 3.3, many existing solutions to the camera pose problem are formulated as least squares problems and assume data are normally distributed.

![Histogram of DVN](image)

Figure 20: Histogram of vector normed difference in pixels on the horizontal axis (see Table 1 for variable declarations, n=434).
3.4.3.2 Bivariate Difference

Two histograms are presented in Figure 21. These histograms are more symmetric in both dimensions of $\epsilon_{BV}$ than for the histogram of $\epsilon_{VN}$. The skew statistic of Mardia’s multivariate test for normality does not reject the hypothesis that $\epsilon_{BV}$ is normally distributed at the 5% level ($p = 0.064$). However, the kurtosis statistic does not infer normality ($p < 0.001$). Variance stabilizing transformations were not applied to the error data.

![Histogram of DX](image1)

![Histogram of DY](image2)

Figure 21: Histograms of bivariate difference in pixels on the horizontal axis (see Table 1 for variable declarations, n=434).

3.4.4 Model Selection

Because the bivariate error distribution more closely resembles a normal distribution than the vector normed error, candidate models for the bivariate error $\epsilon_{BV}$ were parameterized using the VGAM package for R (Yee 2010). Each of the 144 models were parameterized as a vector generalized linear model (VGLM) described by Yee & Hastie (2003). The linear predictor $\eta$ in candidate models included different linear combinations of covariates and some models were nested in others. Generally, candidate models included a bivariate vector from paired observations (such as DistPPX and DistPPY in Table 1) and combinations of zero to nine additional covariates (nine in the case of the vector elements of the rotation matrix).
Based on inference from section 3.4.3.2, the distribution of $\epsilon_{BV}$ was assumed to be multivariate normal with mean $\eta_{BV}$ and covariance $\Sigma_{BV}^2$. Likewise the canonical link function of the VGLM was specified as the identify function. Model selection was performed using AIC as a measure of the relative goodness of fit.

The selected model is

$$\eta_{BV} = B \begin{bmatrix} s_1^* - \omega_1 \\ s_2^* - \omega_2 \end{bmatrix}$$

(53)

where $B \in \mathbb{R}^{2 \times 2}$ is an effects matrix of the covariates DistPPX and DistPPY (see Table 1) with estimates given in Table 3.

Figure 22 shows plots of the response against each of the covariates in the selected model showing the linear relationship between distance from principle point and error in the dimensions of the image space. The diagonal of $B$ (see Table 3) was statistically significant from zero at the 95% level, inferring that these parameters should be considered in the estimation of camera pose. Generally, quantile plots of the residuals in both dimensions appeared to be normally distributed however with heavy tails at the extremes (see Figure 23). The estimated covariance matrix is shown as Table 4.
<table>
<thead>
<tr>
<th>Element of B (row, column)</th>
<th>Parameter Estimate</th>
<th>Standard Error</th>
<th>p-Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,1</td>
<td>-0.0156711*</td>
<td>0.0012981</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>1,2</td>
<td>-0.0019915</td>
<td>0.0012981</td>
<td>0.575</td>
</tr>
<tr>
<td>2,1</td>
<td>0.0024047*</td>
<td>0.0013988</td>
<td>0.042</td>
</tr>
<tr>
<td>2,2</td>
<td>-0.0071081*</td>
<td>0.0013988</td>
<td>&lt;0.001</td>
</tr>
</tbody>
</table>

Table 3: Statistics of the effects matrix B (* indicates significance at the 95% confidence level).

<table>
<thead>
<tr>
<th>Element of Σ (row, column)</th>
<th>Parameter Estimate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,1</td>
<td>67.21643</td>
</tr>
<tr>
<td>1,2</td>
<td>4.61380</td>
</tr>
<tr>
<td>2,1</td>
<td>4.61380</td>
</tr>
<tr>
<td>2,2</td>
<td>37.61723</td>
</tr>
</tbody>
</table>

Table 4: Statistics of the covariance matrix Σ.
Figure 22: Covariate plots showing general trends in the bivariate error in pixels on the horizontal and vertical axes (see Table 1 for variable declarations).

Figure 23: Quantile plots of residuals showing heavy tails at the extremes in pixels on the horizontal and vertical axes (see Table 1 for variable declarations).
3.5 Estimating Camera Pose

Haralick et al. (1989) and others formulate the camera pose problem in the three-dimensional camera frame based on the projective camera model. The object to image space transformation is shown to be equivalent to the projective camera model in its reparameterized form

\[ s = \frac{1}{\tau} (R^{-1}v - \kappa) + \omega \]  \hspace{1cm} (54)

as is apparent from (41). Upon substitution of (50), it is defined in the camera frame as

\[ q = (R^{-1}v - \kappa). \]  \hspace{1cm} (55)

In this form, both the rotation matrix and the projective center are defined in the camera frame which lends its self to the formulation provided by Haralick et al. However as discussed in section 3.3, this formulation presents a problem as the imaging vector \( s \) is observed in the image frame rather than \( q \) in the camera frame.

3.5.1 A Statistical Model for the Imaging Vector

Based on the inference in section 3.4.3.2, assume that \( \epsilon_{BV} \sim N(\eta_{BV}, \Sigma_{BV}^2) \). The bivariate difference defined in (52) can be specified as an error in the three-dimensional camera frame by substituting (50) as

\[ \epsilon = q^* - q \]

\[ = \tau (s^* - \omega) - \tau (s - \omega) \]

\[ = \tau (s^* - s) \]

where \( \epsilon \sim N(\eta, \Sigma^2) \). Following from equation (54), the error can be expressed in terms of camera pose
\[ \epsilon = \tau s^* - R^{-1}v + \kappa - \tau \omega \] (57)

and further in terms of observed \( d^* \) as

\[ \epsilon = \tau (d^* + \omega) - R^{-1}v + \kappa - \tau \omega \] (58)

\[ = \tau d^* - R^{-1}v + \kappa \]

in the reparameterized form of the transformation specified by (42).

Incorporating inference from section in 3.4.4, the mean in (53) is specified in the camera frame as

\[ \eta = C(s^* - \omega) \] (59)

\[ = Cd^* \]

where the effects matrix is redefined as

\[ C = \begin{bmatrix} b_{1,1} & b_{1,2} & 0 \\ b_{2,1} & b_{2,2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \] (60)

and \( b_{i,j} \) are the respective elements of \( B \). The statistical model for the imaging vector is specified by the density of \( \epsilon \) as

\[ p(\epsilon) = \frac{1}{2\pi^2|\Sigma|^2} e^{-\frac{1}{2}(\tau d^* - R_j^{-1}v + \kappa_j - Cd)^T \Sigma^{-1}(\tau d^* - R_j^{-1}v + \kappa_j - Cd)} \] (61)

where \( R_j^{-1} \) and \( \kappa_j \) are unique to the \( j^{th} \) frame with covariance matrix \( \Sigma^2 \). The model presented in (61) is related to a fixed effects model where the effect is the camera pose for each frame.
3.5.2 Likelihood Function

Define the likelihood function for \( \epsilon \) given observed \( d^* \) as the product of \( p(\epsilon) \) over sampled control points in image frames as

\[
\mathcal{L}(R_1, \ldots, R_m, \kappa_1, \ldots, \kappa_m, \tau_1, \ldots, \tau_m, \Sigma, C|d^*) = \prod_{i=1}^{n=m\times\sum k_j} p(\epsilon)
\]

where \( n_j \) is the number of observed points in the \( j^{th} \) frame. The log likelihood is then

\[
\ell_\epsilon \propto -\frac{n}{2}\ln|\Sigma|
\]

\[
-\frac{1}{2} \sum_{i=1}^{n} \left( \tau_i d_i^* - C d_i^* - R_j^{-1} v_i + \kappa_j \right)^T \Sigma^{-1} \left( \tau_i d_i^* - C d_i^* - R_j^{-1} v_i + \kappa_j \right)
\]

where the notation for \( \ell \) omits the parameters and is designated to show it is over the density of \( \epsilon \). The index notation \( i \) presented for the likelihood function means the \( i^{th} \) control point in the \( j^{th} \) camera frame which could have been written \( i,j \). The notation was selected to minimize subscript variables and simplify readability.

3.5.2.1 Constraints on the Rotation Matrix

As expressed in (39), the rotation matrix must be orthogonal with determinant 1. Various constraints have been developed to ensure this condition by Haralick et al. (1989), Quan & Lan (1999), B. Triggs (1999) and J. A. Hesch & Roumeliotis (2011).

The most popular method is proposed by Haralick et al. (1989) who define six constraints for this condition:

\[
r_1^T r_1 = 1
\]

\[
r_2^T r_2 = 1
\]

69
\[ r_3^T r_3 = 1 \]
\[ 2r_1^T r_2 = 0 \]
\[ 2r_2^T r_3 = 0 \]
\[ 2r_3^T r_1 = 0 \]

which are expressed by Lagrange multipliers \( m \) as a linear combination

\[
\sum_{k=1}^{3} m_k (r_k^T r_k - 1) + 2m_4 r_1^T r_2 + 2m_5 r_1^T r_3 + 2m_6 r_2^T r_3. \tag{65}
\]

A different approach is to allow the term \( R_j^{-1} v_i \) to vary linearly in a rotation scale factor \( \alpha_j \) unique to each frame such that the image vector for the \( i^{th} \) control point in the \( j^{th} \) image frame is

\[
s = \frac{1}{\alpha_j \tau_i} R_j^{-1} v_i - \kappa_j + \omega. \tag{66}
\]

Using this formulation, the likelihood function then becomes
\[ \ell_\varepsilon \propto -\frac{n}{2} \ln |\Sigma| \]  

\[
-\frac{1}{2} \sum_{i=1}^{n} \left( \alpha_j \tau_i d^*_i - C d^*_i - \alpha_j^{-1} R_j^{-1} \nu_i + \kappa_j \right)^T \Sigma^{-1} \left( \alpha_j \tau_i d^*_i - C d^*_i - \alpha_j^{-1} R_j^{-1} \nu_i + \kappa_j \right) = -\frac{n}{2} \ln |\Sigma| 
\]

\[
-\frac{1}{2} \sum_{i=1}^{n} \left( \tau_j d^*_i - C d^*_i - R_j^{-1} \nu_i + \kappa_j \right)^T \Sigma^{-1} \left( \tau_j d^*_i - C d^*_i - R_j^{-1} \nu_i + \kappa_j \right) 
\]

where

\[ \tau_j = \alpha_j \tau_i \]  

and

\[ R_j^{-1} = \alpha_j^{-1} R_j^{-1}, \]  

the underbars denoting that the parameters are scaled by \( \alpha_j \).

Upon finding the maximum likelihood estimates (MLEs) for \( \tau_j \) and \( R_j^{-1} \), then the scale parameter can be found by solving for \( \alpha_j \) such that the equation in (39) holds using (90). This approach immediately eliminates the constraint in the maximum likelihood estimators for each parameter but later enforces the constraint after the scaled MLEs have been found as a final step (see section 3.6.1 for solution to \( \alpha_j \)).

### 3.5.3 Maximum Likelihood Estimators

The maximum likelihood estimates (MLEs) are found by taking the derivative of \( \ell_\varepsilon \) with respect to each parameter and solving the zero equations at their maximums. The estimators derived in
the subsequent sections are not profiled: profiled likelihoods were reviewed but found to give unstable solutions during iteration: singularity in $\mathbf{R}_j$ and poor convergence.

### 3.5.3.1 Effects matrix

Using the chain rule, the derivative of the log likelihood with respect to $\mathbf{C}$ is

$$
\frac{\partial \ell_{\epsilon+\delta}}{\partial \mathbf{C}} \propto \sum_{i=1}^{n} \mathbf{\Sigma}^{-1}(\tau_j \mathbf{d}_i^* - \mathbf{C} \mathbf{d}_i^* - \mathbf{R}_j^{-1} \mathbf{v}_i + \kappa_j) \mathbf{d}_i^{*T}.
$$

Evaluated at its maximum, its estimate is

$$
0 = \left. \frac{\partial \ell_{\epsilon}}{\partial \mathbf{C}} \right|_{\mathbf{C} = \hat{\mathbf{C}}}
$$

$$
= \sum_{i=1}^{n} \mathbf{\Sigma}^{-1}(\tau_j \mathbf{d}_i^* - \hat{\mathbf{C}} \mathbf{d}_i^* - \mathbf{R}_j^{-1} \mathbf{v}_i + \kappa_j) \mathbf{d}_i^{*T}
$$

$$
= \mathbf{\Sigma}^{-1} \sum_{i=1}^{n} (\tau_j \mathbf{d}_i^* - \hat{\mathbf{C}} \mathbf{d}_i^* - \mathbf{R}_j^{-1} \mathbf{v}_i + \kappa_j) \mathbf{d}_i^{*T}
$$

$$
= \sum_{i=1}^{n} (\tau_j \mathbf{d}_i^* - \mathbf{R}_j^{-1} \mathbf{v}_i + \kappa_j) \mathbf{d}_i^{*T} - \hat{\mathbf{C}} \sum_{i=1}^{n} \mathbf{d}_i^* \mathbf{d}_i^{*T}
$$

$$
\hat{\mathbf{C}} \sum_{i=1}^{n} \mathbf{d}_i^* \mathbf{d}_i^{*T} = \sum_{i=1}^{n} (\tau_j \mathbf{d}_i^* - \mathbf{R}_j^{-1} \mathbf{v}_i + \kappa_j) \mathbf{d}_i^{*T}
$$

$$
\hat{\mathbf{C}} \mathbf{D} = \sum_{i=1}^{n} (\tau_j \mathbf{d}_i^* - \mathbf{R}_j^{-1} \mathbf{v}_i + \kappa_j) \mathbf{d}_i^{*T}
$$

where
\[ D = \sum_{i=1}^{n} d_i^* d_i^{*T} = \left( \sum_{i=1}^{n} d_i^* \right) \sum_{i=1}^{n} d_i^{*T} \]  

is a square matrix. Assuming \( D \) is invertible, it follows that

\[ \hat{C} = \sum_{i=1}^{n} (\tau_i d_i^* - R_j^{-1} v_i + \kappa_j) d_i^{*T} D^{-1}. \]  

### 3.5.3.2 Covariance Matrix

The derivative with respect to the covariance matrix is

\[ \frac{\partial \ell_e}{\partial \Sigma} \propto \frac{n}{2} \Sigma - \frac{n}{2} \sum_{i=1}^{n} (\tau_i d_i^* - C d_i^* - R_j^{-1} v_i + \kappa_j)(\tau_i d_i^* - C d_i^* - R_j^{-1} v_i + \kappa_j)^T. \]

Assuming \( \hat{\Sigma} \) is invertible, the estimate is evaluated at its maximum as

\[ 0 = \frac{\partial \ell_e}{\partial \Sigma} \bigg|_{\Sigma = \hat{\Sigma}} \]

\[ = \frac{n}{2} \hat{\Sigma} - \frac{n}{2} \sum_{i=1}^{n} (\tau_i d_i^* - C d_i^* - R_j^{-1} v_i + \kappa_j)(\tau_i d_i^* - C d_i^* - R_j^{-1} v_i + \kappa_j)^T \]

\[ \hat{\Sigma} = \sum_{i=1}^{n} (\tau_i d_i^* - C d_i^* - R_j^{-1} v_i + \kappa_j)(\tau_i d_i^* - C d_i^* - R_j^{-1} v_i + \kappa_j)^T. \]

### 3.5.3.3 Projective Center

The derivative with respect to the projective center is
Evaluated at its maximum, its estimate is

$$0 = \frac{\partial \ell_{\epsilon}}{\partial \kappa_j} \bigg|_{\kappa_j = \hat{\kappa}_j}$$

$$= \sum_{h=1}^{n_j} \Sigma^{-1} \left( \tau_h d_h^* - Cd_h^* - R_j^{-1} v_h + \hat{\kappa}_j \right)$$

$$= \Sigma^{-1} \sum_{h=1}^{n_j} \left( \tau_h d_h^* - Cd_h^* - R_j^{-1} v_h + \hat{\kappa}_j \right)$$

$$= n_j \hat{\kappa}_j + \sum_{h=1}^{n_j} \left( \tau_h d_h^* - Cd_h^* - R_j^{-1} v_h \right)$$

$$\hat{\kappa}_j = \frac{1}{n_j} \sum_{h=1}^{n_j} \left( \tau_h d_h^* - Cd_h^* - R_j^{-1} v_h \right).$$

Note that the $\hat{\kappa}_j$ is a function of $C$ and $R_j$.

### 3.5.3.4 Scaled Rotation Matrix

The derivative of $\ell_{\epsilon}$ with respect to $R_j$ is

$$\frac{\partial \ell_{\epsilon}}{\partial R_j} \propto \sum_{i=1}^{n} \Sigma^{-1} \left( \tau_i d_i^* - Cd_i^* - R_j^{-1} v_i + \kappa_j \right) v_i^T.$$

Evaluated at its maximum, the estimate of $R_j^{-1}$ is determined by
where

\[ \mathbf{v}_j = \sum_{h=1}^{n_j} \mathbf{v}_h \mathbf{v}_h^T \]  

(80)

is a sufficient statistic of \( \mathbf{v} \). Assuming that \( \mathbf{V}_j \) is invertible, the MLE of \( \mathbf{R}_j \) from (79) is
\[
\hat{R}_j^{-1} = \sum_{h=1}^{n_j} (\tau_{h} d_{h}^* - C d_{h}^* + \kappa_j) v_{h}^T v_{j}^{-1}.
\]

### 3.5.4 Scaled Optical Depth

Haralick et al. (1989) refer to \( \tau_i \) as the optical depth in the camera frame. As discussed in section 3.3, several solutions have been proposed. Motivated by the likelihood function, another solution can be obtained as a derivative of the likelihood function:

\[
\frac{\partial \ell_e}{\partial \tau_i} \propto \sum_{i=1}^{n} d_i^T \Sigma^{-1} (\tau_i d_i^* - C d_i^* - R_j^{-1} v_i + \kappa_j).
\]

Evaluated at its maximum, the estimate of the optical depth is

\[
0 = \frac{\partial \ell_e}{\partial \tau_i} \bigg|_{\tau_i = \hat{\tau}_i} = d_i^T \Sigma^{-1} (\hat{\tau}_i d_i^* - C d_i^* - R_j^{-1} v_i + \kappa_j)
\]

\[
= \hat{\tau}_i d_i^T \Sigma^{-1} d_i^* - d_i^T \Sigma^{-1} (C d_i^* - R_j^{-1} v_i + \kappa_j)
\]

\[
\hat{\tau}_i d_i^T \Sigma^{-1} d_i^* = d_i^T \Sigma^{-1} (C d_i^* - R_j^{-1} v_i + \kappa_j).
\]

The term \( d_i^T \Sigma^{-1} d_i^* \) is a scalar and if it is non-zero, then from (83) the estimate is

\[
\hat{\tau}_i = \frac{d_i^T \Sigma^{-1} (C d_i^* - R_j^{-1} v_i + \kappa_j)}{d_i^T \Sigma^{-1} d_i^*}.
\]

Note that this estimate has similar form to (47).
### 3.5.5 Finding the Trajectory Parameters

Upon convergence of the rotation matrix as described in section 3.5.7, it can be deconstructed to find $\hat{\Theta}_j$ (Slabaugh 2012). As defined in (37), the rotation matrix yields the following system

\[
\begin{aligned}
\begin{cases}
\frac{d}{a} = \tan \hat{\alpha} \\
g = -\sin \hat{\beta} \\
h \frac{1}{i} = \tan \hat{\gamma}
\end{cases}
\end{aligned}
\]  

(85)

From this system, two solutions exist for $\hat{\beta}$ as $\hat{\beta}_1 = -\sin^{-1} g$ and $\hat{\beta}_2 = \pi - \sin^{-1} g$ where $\pi$ is the quantity. Following from these solutions the estimates are

\[
\begin{aligned}
\hat{\alpha}_1 &= \tan^{-1} \left( \frac{d}{\cos \beta_1}, \frac{a}{\cos \beta_1} \right) \\
\hat{\alpha}_2 &= \tan^{-1} \left( \frac{d}{\cos \beta_2}, \frac{a}{\cos \beta_2} \right) \\
\hat{\beta}_1 &= -\sin^{-1} g \\
\hat{\beta}_2 &= \pi - \sin^{-1} g \\
\hat{\gamma}_1 &= \tan^{-1} \left( \frac{h}{\cos \beta_1}, \frac{i}{\cos \beta_1} \right) \\
\hat{\gamma}_2 &= \tan^{-1} \left( \frac{h}{\cos \beta_2}, \frac{i}{\cos \beta_2} \right)
\end{aligned}
\]

(86)

when $g \neq 1$, otherwise the rotation matrix is in Gimbal lock (Slabaugh 2012) where $\hat{\gamma}$ is the real line.

### 3.5.6 Finding the Projective Center

Upon convergence of $\hat{\kappa}_j$ and $\hat{R}_j$ as described in section 3.5.7, they can be used to find $\hat{\lambda}_j$. From (41) the projective center is simply
\[ \hat{\lambda} = \hat{R}\hat{\kappa}. \] (87)

### 3.5.7 Iterative Method

The convergence function of a parameter \( p \) between iterations \( k \) and \( k + 1 \) is

\[
\delta(p^k, p^{k+1}) = \frac{\|p^{k+1} - p^k\|}{2\|p^{k+1} + p^k\|} \tag{88}
\]

The convergence function indicates when the likelihood function for some parameter value is maximized. Upon maximization, the estimate does not significantly change upon iteration. Other convergence functions can be defined to indicate maximum likelihood.

Because many of the parameters presented in sections 3.5.3 and 3.5.4 are nested within each other’s estimators, the model solution to the pose problem is not linear in the data. Hence the solution must be found using an iterative algorithm based on conditional parameter estimates. One such algorithm to estimate the model is:

(A) Initialize iteration \( l = 0 \) as no effects
   (a) Initialize \( \hat{C}^l = I \) as the identity matrix
   (b) Initialize \( \hat{S}^l = I \) as the identity matrix

(B) Initialize iteration \( k = 0 \) for each \( j^{th} \) camera frame \( 1, ..., m \)
   (a) Initialize \( \hat{\theta}_j^k \) as a zero vector
   (b) Initialize \( \hat{R}_j^k \) from \( \hat{\theta}_j^k \) as the identity matrix per (37)
   (c) Initialize the camera projective center as

\[
\hat{k}^k = \begin{bmatrix} 0 \\ 0 \\ r_{\lambda} \end{bmatrix} + \frac{1}{n_j} \sum_{i=1}^{n_j} \mathbf{v}_i
\]

where \( r_{\lambda} \) is an initial guess at the altitude of the aircraft relative to the ground,
(d) Solve for each $\hat{\tau}_i^k$ per (84)
(C) Frame step: for each $j^{th}$ camera frame 1, ..., $m$
   (a) Let $\hat{\mathbf{C}} = \hat{\mathbf{C}}^l$
   (b) Let $\hat{\mathbf{S}} = \hat{\mathbf{S}}^l$
   (c) Lambda step: iterate $k$ until $\delta(\hat{\lambda}^k, \hat{\lambda}^{k+1}) < 0.01$
      (i) Fix $\hat{\tau}_i^{k+1} = \hat{\tau}_i^k$
      (ii) Solve for $\hat{\mathbf{R}}_j^{k+1}$ per (81) where $\kappa_j = \hat{\kappa}_j^k$ and $\alpha_j \tau_i = \hat{\tau}_i^k$
      (iii) Solve for $\hat{\kappa}_j^{k+1}$ per (77) where $\mathbf{R}_j = \hat{\mathbf{R}}_j^{k+1}$ and $\alpha_j \tau_i = \hat{\tau}_i^k$
   (d) Tau step: iterate $k$ until $\delta(\hat{\lambda}^k, \hat{\lambda}^{k+1}) < 0.01$
      (i) Solve for each $\hat{\tau}_i^{k+1}$ per (84) where $\kappa_j = \hat{\kappa}_j^k$ and $\mathbf{R}_j = \hat{\mathbf{R}}_j^k$
      (ii) Let $\hat{\mathbf{R}}_j^{k+1} = \hat{\mathbf{R}}_j^k$
      (iii) Let $\hat{\kappa}_j^{k+1} = \hat{\kappa}_j^k$
      (iv) Repeat lambda step (a) above
   (e) Alpha normalization step: iterate $k$ once
      (i) Solve for $\alpha_j$ in (69) such that $|\alpha_j^{-1} \hat{\mathbf{R}}_j^k| = 1$ and $\alpha_j^{-1} \hat{\mathbf{R}}_j^k$ is orthogonal by

$$
1 = \alpha_j^{-3}(aei + bfg + cdh - ceg - bdi - afh) \quad (90)
$$

$$
\alpha_j = (aei + bfg + cdh - ceg - bdi - afh)^{1/3}
$$

where the variables $a, b, ..., i$ are defined by (38) in $\hat{\mathbf{R}}_j^k$
      (i) Solve for each $\hat{\tau}_i^{k+1} = \hat{\tau}_i^k$ using (68)
      (ii) Let $\alpha_j = 1$
      (iii) Repeat lambda step (a) above
(D) Model step: iterate $l$ repeating until $\delta(|\hat{\mathbf{C}}^l|, |\hat{\mathbf{C}}^{l+1}|) < 0.01$
   (a) Let $\mathbf{R}_j = \hat{\mathbf{R}}_j^k$
   (b) Let $\kappa_j = \hat{\kappa}_j^k$
   (c) Let $\alpha_j = 1$
   (d) Let $\tau_i = \hat{\tau}_i^k$
   (e) Solve for $\hat{\mathbf{C}}^{l+1}$ per (91)
   (f) Solve for $\hat{\mathbf{S}}^{l+1}$ per (75)
   (g) Repeat the frame step (C) above
This algorithm specifies two iterations: the first is within each camera frame where sequential iterates are denoted by \( k \) and \( k + 1 \), and the second is across camera frames denoted by \( l \) and \( l + 1 \) where the first iteration is nested in the second.

The algorithm can be viewed as a variant of the expectation maximization method. The MLEs are achieved by searching the parameter space using conditional likelihood functions that alternately restrict the search to certain subspaces. The MLE of each parameter is repeatedly estimated until its conditional likelihood reaches some maximum as indicated by the convergence function. Two likelihood functions are iterated in the lambda step conditional on other parameters,

\[
\mathcal{L}(R_{1}^{k+1}, ..., R_{m}^{k+1}| \Sigma^{l}, C^{l}, \kappa_{1}^{k}, ..., \kappa_{m}^{k}, \tau_{1}, ..., \tau_{n}, d^{*})
\]

\[
\mathcal{L}(\kappa_{1}^{k}, ..., \kappa_{m}^{k}| \Sigma^{l}, C^{l}, R_{1}^{k+1}, ..., R_{m}^{k+1}, \tau_{1}, ..., \tau_{n}, d^{*}).
\]

effectively restricting the search to the six-dimensional parameter subspace spanning \( \theta \) and \( \lambda \) by fixing \( \Sigma^{l}, C^{l} \) and \( \tau_{1}, ..., \tau_{n} \). Convergence in the lambda step is realized in \( \hat{\lambda} \) which is the matrix product of the two MLEs defined by (87).

The tau step is similar to the lambda step but seeks to maximize all \( \tau_{1}, ..., \tau_{n} \) conditional on all the other parameters. Likewise the model step seeks to maximize the effect and covariance matrices conditional on all other parameters. However, the alpha normalization step does not seek to maximize the likelihood of \( \alpha_{j} \). Rather it seeks to normalize \( \hat{t}_{1}, ..., \hat{t}_{n} \) and \( \hat{R}_{j} \) which are linear in \( \alpha_{j} \) due to the linearity assumption of the constraint formulation in (67).
3.6 Results

Camera pose and model parameters were estimated to analyze the basic convergence properties of the iterative method and to study the distributions of estimated camera pose parameters. The first of these is useful to confirm that the iterative method converges to reasonable estimates and to analyze the effect of sample size on estimate accuracy. The latter of these is useful to infer the expected precision of estimates in application.

Frame-based results were derived from a trivial dataset that was constructed by randomly generating 400 control points in a volume that measured 800 meters in the X-dimension, 800 meters in the Y-dimension and 200 meters in the Z-dimension about the origin. Subsets of the data were randomly selected to reduce the sample sizes \( n \) of frames during analysis. Model-based results were derived from ten trivial frames, each frame constructed as described above for frame-based results.

The image ray and the iterative method were implemented in C# computer code and analyses were conducted in R, a statistical computing program.

3.6.1 Convergence Properties

Convergence properties of estimates were studied in both the frame step for camera pose parameters and the model step for model parameters of the iterative method.

3.6.1.1 Frame Step

To analyze the effect of sample size on the accuracy of camera pose estimates in a single frame, the data were randomly subsampled to achieve subsets of cardinality \( n = 8, 40, 80 \) and 400. The object to image space transformation was applied to the data using (30). The transformed data were not perturbed by (59) or by any Gaussian noise. The true parameter values were purposely selected to provide an oblique look angle into the data.

The frame step of the iterative method was then applied to each subset to observe the convergence paths of the estimated camera pose parameters. It was initialized with \( r_2 = 800 \)
meters, an altitude far higher than the true value of 200 meters (see Table 5). The convergence path of the estimated projective center $\hat{\lambda}$ was observed about the region of the known true parameter value in the object space (see Appendix A).

Figures A.1 through A.10 show that the estimates jumps wildly early in the method and then approach the true values. Figures A.11 through A.14 show that the estimates converge with fewer iterations at smaller sample sizes and that the convergence function $\delta(\hat{\lambda}^k, \hat{\lambda}^{k+1})$ spikes immediately before convergence in all $n > 8$ subsets.

The frame-based results in Table 5 indicate that the accuracy of parameter estimates improves with sample size. In the case of these data, the estimates are fairly good using a sample size of 40.

<table>
<thead>
<tr>
<th></th>
<th>$x_\lambda$ (m)</th>
<th>$y_\lambda$ (m)</th>
<th>$z_\lambda$ (m)</th>
<th>$\alpha$ (yaw$^\circ$)</th>
<th>$\beta$ (pitch$^\circ$)</th>
<th>$\gamma$ (roll$^\circ$)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>True</strong></td>
<td>125</td>
<td>-60</td>
<td>200</td>
<td>122</td>
<td>14</td>
<td>-17</td>
</tr>
<tr>
<td><strong>Values</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>n=8</strong></td>
<td>41.32</td>
<td>-20.1</td>
<td>23.83</td>
<td>120.15</td>
<td>9.71</td>
<td>-16.6</td>
</tr>
<tr>
<td></td>
<td>(-83.68,</td>
<td>(39.9, -</td>
<td>(-176.17,</td>
<td>(-1.85, -</td>
<td>(-4.29,</td>
<td>(0.4, -2.3%)</td>
</tr>
<tr>
<td></td>
<td>-66.9%)</td>
<td></td>
<td>-88.1%)</td>
<td>1.5%)</td>
<td>-30.7%)</td>
<td></td>
</tr>
<tr>
<td><strong>n=40</strong></td>
<td>122.87</td>
<td>-59.12</td>
<td>193.57</td>
<td>121.96</td>
<td>13.88</td>
<td>-16.95</td>
</tr>
<tr>
<td></td>
<td>(-2.13,</td>
<td>(0.88, -</td>
<td>(-6.43, -</td>
<td>(-0.04, 0%)</td>
<td>(-0.12,</td>
<td>(0.05, -</td>
</tr>
<tr>
<td></td>
<td>1.7%)</td>
<td></td>
<td>3.2%)</td>
<td></td>
<td>0.8%)</td>
<td>0.3%)</td>
</tr>
<tr>
<td><strong>n=80</strong></td>
<td>124.89</td>
<td>-59.96</td>
<td>199.67</td>
<td>122</td>
<td>14</td>
<td>-17</td>
</tr>
<tr>
<td></td>
<td>(-0.11,</td>
<td>(0.04, -</td>
<td>(-0.33, -</td>
<td>(0, 0%)</td>
<td>(0, 0%)</td>
<td>(0, 0%)</td>
</tr>
<tr>
<td></td>
<td>0.1%)</td>
<td></td>
<td>0.2%)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>n=400</strong></td>
<td>125.14</td>
<td>-60.08</td>
<td>201.01</td>
<td>121.99</td>
<td>13.96</td>
<td>-16.93</td>
</tr>
<tr>
<td></td>
<td>(0.14, 0.1%)</td>
<td>(-0.08,</td>
<td>(1.01, 0.5%)</td>
<td>(-0.01, 0%)</td>
<td>(-0.04,</td>
<td>(0.07, -</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.1%)</td>
<td></td>
<td></td>
<td>0.3%</td>
<td>0.4%</td>
</tr>
</tbody>
</table>
Table 5: True parameter values and estimates based on varying sample sizes (estimate error of the true known parameter in meters or degrees, and as percentage of true parameter).

3.6.1.2 Model Step

Based on results from the frame step analysis, a sample size of \( n = 48 \) was selected to analyze the parameter estimates of the effects matrix \( C \). The data were randomly subsampled to achieve 20 subsets of 48 data each that corresponded to 20 camera frames. The object to image space transformation was applied to the data in each frame using (30). The transformed data were perturbed by (59) where the elements of the effects matrix in (60) were assigned the values from Table 3. The transformed data were further perturbed with bivariate Gaussian noise where the elements of \( \Sigma \) were assigned values from Table 4, otherwise zeros in the third row and column.

For each frame, a set of true camera pose parameter values was purposely selected to provide an oblique look angle into the data. These values were determined by adding a random distance between -20 and 20 meters to the true projective center values in Table 5 and a random degree between -2 and 2 degrees to the true rotation values in Table 5, using the uniform distribution.

The iterative method was then applied to the data across all frames to estimate model parameters. The frame step for each frame was initialized with \( r_\lambda = 800 \) meters. The convergence function \( \delta(\mid \hat{\gamma}^i \mid, \mid \hat{\gamma}^{i+1} \mid) \) quickly approached the threshold after two iterations as shown by the model-based results in Figure 24. The parameter estimates of the effects matrix are shown in Table 6. The parameter estimates show some error, however none of the observed errors were statistically significant from zero given the estimated standard errors in Table 3. Errors and parameter estimates in the third row and third column of the effects matrix are the same because the true values are zero; any estimate other than zero is entirely an error.
<table>
<thead>
<tr>
<th>Element of ( \hat{C} ) (row, column)</th>
<th>Parameter Estimate</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,1</td>
<td>0.24441</td>
<td>0.26008</td>
</tr>
<tr>
<td>1,2</td>
<td>0.03929</td>
<td>0.04128</td>
</tr>
<tr>
<td>1,3</td>
<td>-0.00081</td>
<td>-0.00081</td>
</tr>
<tr>
<td>2,1</td>
<td>0.03963</td>
<td>0.03723</td>
</tr>
<tr>
<td>2,2</td>
<td>0.27277</td>
<td>0.27988</td>
</tr>
<tr>
<td>2,3</td>
<td>0.0014</td>
<td>0.0014</td>
</tr>
<tr>
<td>3,1</td>
<td>-0.06188</td>
<td>-0.06188</td>
</tr>
<tr>
<td>3,2</td>
<td>-0.03813</td>
<td>-0.03813</td>
</tr>
<tr>
<td>3,3</td>
<td>0.00043</td>
<td>0.00043</td>
</tr>
</tbody>
</table>

Table 6: Parameter estimates of the effects matrix and observed error.
3.6.2 Yellowstone Data

The iterative method was applied to the Yellowstone data presented in section 3.4.2 to analyze the convergence error in estimated parameters. Errors were examined in the estimates of camera pose parameters and the effects matrix in Tables 7 and 8, respectively. Errors and parameter estimates in the third row and third column of the effects matrix are the same because the true values are zero; any estimate other than zero is entirely an error.
Based on review of IMU measurements, the flight trajectory of the aircraft was uniform with fairly constant heading (yaw). Pitch and roll of the aircraft was minor, indicating that the look of the camera was nearly nadir.

Upon estimation, yaw was consistently over estimated while all other parameters were consistently underestimated based on the values of the medians errors. Possible causes of this observed bias are discussed in section 3.7.

<table>
<thead>
<tr>
<th>Image</th>
<th>n_j</th>
<th>x_λ (m)</th>
<th>y_λ (m)</th>
<th>z_λ (m)</th>
<th>α (yaw°)</th>
<th>β (pitch°)</th>
<th>γ (roll°)</th>
</tr>
</thead>
<tbody>
<tr>
<td>90010</td>
<td>28</td>
<td>573747.2</td>
<td>4980934.4</td>
<td>3336.8</td>
<td>188.4</td>
<td>3.1</td>
<td>-2.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(-18.6)</td>
<td>(-20.4)</td>
<td>(-29)</td>
<td>(0.2)</td>
<td>(-1.4)</td>
<td>(-0.5)</td>
</tr>
<tr>
<td>90011</td>
<td>24</td>
<td>573739.5</td>
<td>4980857.8</td>
<td>3321</td>
<td>187.9</td>
<td>3.8</td>
<td>-2.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(-22.6)</td>
<td>(-10.8)</td>
<td>(-15.2)</td>
<td>(0.3)</td>
<td>(-1.8)</td>
<td>(-0.4)</td>
</tr>
<tr>
<td>90012</td>
<td>21</td>
<td>573728.1</td>
<td>4980809.4</td>
<td>3313.9</td>
<td>188.6</td>
<td>4.3</td>
<td>-2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(-22.2)</td>
<td>(-28.4)</td>
<td>(-10)</td>
<td>(0.2)</td>
<td>(-2.1)</td>
<td>(-1)</td>
</tr>
<tr>
<td>90013</td>
<td>21</td>
<td>573721.4</td>
<td>4980722.7</td>
<td>3330.5</td>
<td>188.4</td>
<td>3.6</td>
<td>-0.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(-26.2)</td>
<td>(-7.7)</td>
<td>(-28)</td>
<td>(0)</td>
<td>(-1.3)</td>
<td>(-0.9)</td>
</tr>
<tr>
<td>90014</td>
<td>18</td>
<td>573703.9</td>
<td>4980668.1</td>
<td>3328.9</td>
<td>188</td>
<td>4.1</td>
<td>-0.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(-19)</td>
<td>(-20)</td>
<td>(-27.6)</td>
<td>(0.1)</td>
<td>(-1.5)</td>
<td>(-0.3)</td>
</tr>
<tr>
<td>90015</td>
<td>15</td>
<td>573700.2</td>
<td>4980590.6</td>
<td>3326.9</td>
<td>189.7</td>
<td>4.6</td>
<td>0.9</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(-25.4)</td>
<td>(-8.6)</td>
<td>(-26.6)</td>
<td>(0)</td>
<td>(-2.1)</td>
<td>(-0.6)</td>
</tr>
<tr>
<td>90016</td>
<td>12</td>
<td>573679.8</td>
<td>4980529.6</td>
<td>3318.6</td>
<td>188.6</td>
<td>3.9</td>
<td>2.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(-15.4)</td>
<td>(-13.6)</td>
<td>(-19.4)</td>
<td>(0.3)</td>
<td>(-1.4)</td>
<td>(-0.7)</td>
</tr>
<tr>
<td>90017</td>
<td>11</td>
<td>573677.2</td>
<td>4980469</td>
<td>3308.6</td>
<td>189.9</td>
<td>3.5</td>
<td>3.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(-23.4)</td>
<td>(-19)</td>
<td>(-10.2)</td>
<td>(0)</td>
<td>(-1.4)</td>
<td>(-1.2)</td>
</tr>
<tr>
<td>90018</td>
<td>12</td>
<td>573666.5</td>
<td>4980416</td>
<td>3317.4</td>
<td>189.1</td>
<td>3.2</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(-23.8)</td>
<td>(-32)</td>
<td>(-19.8)</td>
<td>(0.2)</td>
<td>(-1.3)</td>
<td>(-1.5)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----</td>
<td>----</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>------</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td></td>
<td>17</td>
<td>573660.7</td>
<td>4980332.2</td>
<td>3310.4</td>
<td>187.4</td>
<td>3.9</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>573634.1</td>
<td>4980273.3</td>
<td>3315.7</td>
<td>187.6</td>
<td>3.1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>573638.6</td>
<td>4980206.8</td>
<td>3318.7</td>
<td>187.7</td>
<td>4.2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>22</td>
<td>573628.5</td>
<td>4980138</td>
<td>3315.4</td>
<td>187</td>
<td>4.9</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>24</td>
<td>573605.6</td>
<td>4980076.4</td>
<td>3312.9</td>
<td>188.6</td>
<td>4.7</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>573605.3</td>
<td>4980099.4</td>
<td>3319.3</td>
<td>189.2</td>
<td>5.6</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>17</td>
<td>573585</td>
<td>4979939.7</td>
<td>3311.4</td>
<td>190.5</td>
<td>4.5</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>573576.3</td>
<td>4979873.1</td>
<td>3310.5</td>
<td>190.7</td>
<td>4.3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>21</td>
<td>573568.6</td>
<td>4979803</td>
<td>3306.6</td>
<td>188.8</td>
<td>4.3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>21</td>
<td>573549.4</td>
<td>4979742.6</td>
<td>3306.5</td>
<td>189.4</td>
<td>5.3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>19</td>
<td>573552</td>
<td>4979680</td>
<td>3313.3</td>
<td>189.6</td>
<td>4.7</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>573536.4</td>
<td>4979630.6</td>
<td>3312</td>
<td>188.8</td>
<td>5.7</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>27</td>
<td>573533.5</td>
<td>4979570.8</td>
<td>3307.3</td>
<td>188.5</td>
<td>5.1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>28</td>
<td>573524.7</td>
<td>4979488.2</td>
<td>3301.6</td>
<td>189</td>
<td>4.6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Sample sizes and estimates of camera pose parameters for camera frames in the Yellowstone data (error relative to GPS/IMU measurement in meters or degrees).</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(-28.6)</td>
<td>(-11.2)</td>
<td>(-12.8)</td>
<td>(0.2)</td>
<td>(-1.3)</td>
<td>(0.1)</td>
<td></td>
</tr>
<tr>
<td>90033</td>
<td>31</td>
<td>573497.9</td>
<td>4979422.7</td>
<td>3314.7</td>
<td>189</td>
<td>5.2</td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td>(-11.8)</td>
<td>(-9.7)</td>
<td>(-26)</td>
<td>(0.1)</td>
<td>(-1.7)</td>
<td>(0.2)</td>
<td></td>
</tr>
<tr>
<td>90034</td>
<td>25</td>
<td>573488.7</td>
<td>4979377</td>
<td>3318.8</td>
<td>189.8</td>
<td>4.9</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>(-12.6)</td>
<td>(-27)</td>
<td>(-30)</td>
<td>(0.3)</td>
<td>(-1.2)</td>
<td>(-0.7)</td>
<td></td>
</tr>
<tr>
<td>90035</td>
<td>23</td>
<td>573490.3</td>
<td>4979304.8</td>
<td>3314.5</td>
<td>190.5</td>
<td>5.9</td>
<td>-1</td>
</tr>
<tr>
<td></td>
<td>(-24.2)</td>
<td>(-17.8)</td>
<td>(-25.6)</td>
<td>(0.1)</td>
<td>(-2.3)</td>
<td>(0)</td>
<td></td>
</tr>
<tr>
<td>90036</td>
<td>17</td>
<td>573471.4</td>
<td>4979234.5</td>
<td>3314.4</td>
<td>190</td>
<td>5.5</td>
<td>0.6</td>
</tr>
<tr>
<td></td>
<td>(-15.4)</td>
<td>(-10.5)</td>
<td>(-25.4)</td>
<td>(0.1)</td>
<td>(-1.9)</td>
<td>(-0.7)</td>
<td></td>
</tr>
<tr>
<td>90037</td>
<td>19</td>
<td>573462.3</td>
<td>4979183.6</td>
<td>3304.8</td>
<td>190.6</td>
<td>5.5</td>
<td>1.7</td>
</tr>
<tr>
<td></td>
<td>(-16.4)</td>
<td>(-22.6)</td>
<td>(-16)</td>
<td>(0.3)</td>
<td>(-1.6)</td>
<td>(-1.2)</td>
<td></td>
</tr>
<tr>
<td>90038</td>
<td>22</td>
<td>573447.4</td>
<td>4979123.6</td>
<td>3296.9</td>
<td>190.2</td>
<td>6.7</td>
<td>1.3</td>
</tr>
<tr>
<td></td>
<td>(-12)</td>
<td>(-25.6)</td>
<td>(-8.8)</td>
<td>(0.3)</td>
<td>(-2.3)</td>
<td>(-1)</td>
<td></td>
</tr>
<tr>
<td>90039</td>
<td>21</td>
<td>573434.6</td>
<td>4979068</td>
<td>3305.3</td>
<td>190.9</td>
<td>6.2</td>
<td>-1.1</td>
</tr>
<tr>
<td></td>
<td>(-9.8)</td>
<td>(-32)</td>
<td>(-17.6)</td>
<td>(0.3)</td>
<td>(-1.9)</td>
<td>(-0.3)</td>
<td></td>
</tr>
<tr>
<td>90040</td>
<td>19</td>
<td>573427.1</td>
<td>4978995.4</td>
<td>3301.3</td>
<td>190.8</td>
<td>5.7</td>
<td>2.3</td>
</tr>
<tr>
<td></td>
<td>(-13)</td>
<td>(-21.4)</td>
<td>(-13.4)</td>
<td>(0.3)</td>
<td>(-1.8)</td>
<td>(-1.1)</td>
<td></td>
</tr>
<tr>
<td>Median</td>
<td>573585</td>
<td>4979939.7</td>
<td>3313.9</td>
<td>189</td>
<td>4.6</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(-19.6)</td>
<td>(-18.4)</td>
<td>(-19.6)</td>
<td>(0.2)</td>
<td>(-1.6)</td>
<td>(-0.6)</td>
<td></td>
</tr>
<tr>
<td>Element of $\mathbf{\hat{C}}$ (row, column)</td>
<td>Parameter Estimate</td>
<td>Difference*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>-------------------</td>
<td>-------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1,1</td>
<td>0.450655</td>
<td>0.466326</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1,2</td>
<td>0.075427</td>
<td>0.077418</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1,3</td>
<td>-0.001</td>
<td>-0.001</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2,1</td>
<td>0.068408</td>
<td>0.066003</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2,2</td>
<td>0.395013</td>
<td>0.402121</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2,3</td>
<td>0.003801</td>
<td>0.003801</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3,1</td>
<td>-0.07473</td>
<td>-0.07473</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3,2</td>
<td>-0.05498</td>
<td>-0.05498</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3,3</td>
<td>0.000647</td>
<td>0.000647</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 8: Parameter estimates of the effects matrix and their differences from those in Table 3 (* difference between parameter estimates and estimates in Table 3).
3.6.3 Parameter Distributions

To identify the potential statistical distributions of the estimated camera pose parameters, the data were randomly subsampled to achieve subsets of cardinality \( n = 24, 48 \) and 64. These sample sizes were selected based on the results in section 3.6.1.1 that indicate a sample size around 40 gives reasonable results. The data in each subsample were bootstrapped 5000 times to create bootstrapped subsamples for each of the three sample sizes.

The object to image space transformation was applied to the bootstrapped subsamples using (30). The transformed data were not perturbed by (59) or by any Gaussian noise. The frame step was then applied to each bootstrapped subsample. The true parameter values were purposely selected to provide an oblique look angle into the data.

Histograms of the estimated camera pose parameters from each bootstrapped subsample were created for each of the three sample sizes. These histograms shown in Figures 25, 26 and 27 indicate that parameter estimates may be biased, however that the bias may decrease in some estimates and increase in others as sample size increases. Possible causes of this observed bias are discussed in section 3.7. The histograms also indicate that the variance of parameter estimates may decrease as sample size increases.
Figure 25: Histograms of estimated camera pose parameters by bootstrap with n=24 (X, Y and Z in meters; Yaw, Pitch and Roll in degrees; dashed lines indicate true values).
Figure 26: Histograms of estimated camera pose parameters by bootstrap with n=48 (X, Y and Z in meters; Yaw, Pitch and Roll in degrees; dashed lines indicate true values).
Figure 27: Histograms of estimated camera pose parameters by bootstrap with n=64 (X, Y and Z in meters; Yaw, Pitch and Roll in degrees; dashed lines indicate true values).
3.7 Discussion

Although many solutions to the camera pose problem have been provided in the literature, none examine the parametric assumptions in their formulation. Further, no models have been published for error in observed control points in images. As shown in section 3.4.3, depending on how the error term is defined and the dataset, different distributions may exist. Dimensions of the error may have non-zero covariance and follow a trend in the image space, as demonstrated in section 3.4.4. These findings should motivate the close examination of these assumptions when formulating solutions to the camera pose problem.

One such solution is presented in section 3.5 as a statistical model for the imaging vector and shows that estimates of camera pose in this model are closed-form functions. However the model can be improved in several ways. Foremost, the rotation matrix is not directly constrained in its maximum likelihood estimator. However the estimator does provide an unconstrained solution for the rotation matrix based on the assumption that the constrained rotation matrix is linear in a rotation scale factor. Although a novel solution to the constraint problem, the reasonableness and robustness of the scale factor assumption have not been validated. Other constrained solutions are provided by Haralick et al. (1989) and others under specific parametric assumptions about the distribution of the vector normed difference (see section 3.4.3.1).

Like the rotation matrix, constraints should be formulated for the effects matrix. As specified by (53), the effects matrix should be constrained to the (two dimensional) image plane. This specification is supported by exploratory data analysis (EDA) which does not infer any significant effect of the optical depth \( \tau \) on error (see section 3.4.4). If error were present in the third dimension of the camera frame, then effect of \( \tau \) on error would have been statistically significant as \( q \) in the camera frame is only linear in \( \tau \) as shown in (50) (see results in section 3.4.4).

Because of the model specified in (61), the effects matrix \( C \) necessarily spans the entire three dimensions of the camera frame (the effects matrix is defined in (60)). One possible approach
is to constrain the likelihood function in (63) using Lagrange multipliers as suggested by (65). However this approach was attempted without successfully finding closed form estimators of the rotation or effect matrices. The mathematical work of a Lagrangian specification is not presented but is definitely an important area for further research.

Exploratory analysis in section 3.4 indicates that bivariate error distributions may be symmetric and approximately Gaussian for certain datasets. This analysis also infers that error is positively affected by distance from the principle point on the image plane. Upon accounting for this effect, the distribution of residual error exhibits heavy tails; to account for these heavy tails, further EDA could be performed to find a parametric distribution that better fits the data and to reformulate the likelihood function in section 3.5.2. Alternatively, variance stabilizing transformations could be applied to induce normality, if any exist.

The iterative method presented in section 3.6.1 seeks to find parameter MLEs by maximizing the likelihood function. This approach is similar to expectation maximization (EM) methods in that it relies on iteratively maximizing conditional likelihoods with the overall objective of maximizing the full likelihood. A true EM method considers the expected value of the conditional likelihoods during iteration (T. Hastie et al. 2003). But unlike a true EM method, the proposed algorithm does not consider expectation which may result in bias parameter estimates. Although the bias may be small within each step of the proposed iterative method, the effect can compound after thousands of iterations. This compounded effect could be a contributing factor to the observed bias of the results in section 3.6. Both frame-based and (more-so) model-based results appear to suffer from consistent and measureable bias. As an area of further research, intermediate expectation steps could be added to the algorithm to reduce the possible effect of compound bias.

Each of the maximization steps in the various parameter subspaces – subspaces associated with the conditional likelihoods – iteratively seek to find a global maximum. However, it is understood that EM methods are not guaranteed to converge to a global maximum (T. Hastie
et al. 2003). Further research should examine the conditions that ensure the algorithm finds the global maximum, conditions which include how the algorithm initialized (Haralick et al. 1989).

Savings in compute time can be realized in the frame step of the iterative method as the frame step can be implemented in parallel processes across camera frames. This is possible because the step uses conditional likelihoods which are conditionally independent of all other model parameters, including those expressed in (92). Therefore, all frames can be computed at once in parallel during the frame step, between model steps.

Although point estimates are provided in section 3.5 and parameter distributions in section 3.6.3, no closed form estimates of confidence regions are derived. Following from large sample theory, this effort requires that the Fisher information be derived from the likelihood function. Because many estimates are based on conditional likelihoods, this is a challenging mathematical endeavor. Although closed form solutions may exist, it may be more pragmatic to use the resampling methods described in section 3.6.3 to estimate confidence regions rather than closed form estimates.

Finally, section 3.6.2 is an analysis of an estimated model compared to the true (known) model based on the Yellowstone data described in section 3.4.2. Some additional work could be undertaken to study the relative accuracy of the estimated model compared to models estimated using other methods in the literature, and to analyze the results based on other datasets. The convergence qualities and initialization conditions that are proposed in section 3.5.7 could be further examined, in far more detail than presented in section 3.6.1: equivalent solutions to the eight from the maximization of the traditional cost function could be identified (J. A. Hesch & Roumeliotis 2011). The cause of the interesting spikes observed in Figures A.11 though A.14 is also a topic for research (see Appendix A).
3.7.1 Contribution

This chapter provides the following unique contributions:

- Based on identities of the image ray in some existing methods to estimate camera pose, the image ray is presented as a unifying element of these methods;
- A statistical review of the treatment of error in existing methods to estimate camera pose;
- A model for observed error in control points to estimate camera pose;
- A statistical method for the simultaneous estimation of camera pose and error across multiple camera frames; and
- A new scale factor to constrain the rotation matrix in estimation.
Chapter Four: Probability Model for Image-Lidar Fusion

This chapter presents the problem of image-lidar fusion, establishes a theoretical framework for the problem to maximize information in fusion data and provides measures on uncertainty, and proposes the probability fusion model as a solution. Fusion reduction and duplicity are identified as the cruxes of the problem; reduction is the occlusion of the point cloud by itself while duplicity is caused by overlapping camera images. The solution is unified by the relative geometry of camera images and the point cloud, derived as a statistical estimator. The estimator is shown to be unbiased with efficiency equivalent to or greater than the sample mean. Likewise the standard error of the probability fusion model is derived. An optical depth covariance (ODC) ratio is defined to measure the effect of the variances of the point cloud and camera projective center on the transformation probability, a component of the model. Finally, resampling methods are used to estimate an approximate density function for the transformation probability under the condition that the ODC ratio is small. Several corollaries are identified, including generalities about the Horvitz-Thompson weights and Inverse Distance Weighting (IDW) as they relate to prior probabilities on estimates.
4.1 Overview

Presently, the integration of multiple sensors is a relatively new and widely publicized topic in scientific literature (for example see recent work by Albarelli et al. 2013; Farooque et al. 2013; Villa et al. 2013). Advancing technologies and retreating costs spur their adoption: digital cameras are commonplace and laser scanners are being used to assemble expansive datasets. Despite the proliferation of these sensor data, though, they are often disparate in their applications; operations of camera sensors are largely independent of laser scanners. Referred to as sensor fusion in the literature, integrating these sensors is fundamental to improving application performance, especially in object discrimination and classification.

The fusion of pixel data from camera sensors and lidar point clouds from laser scanners is important for three reasons. First, the sensors are complementary as the camera measures surface irradiance of objects while the laser scanner measures physical points on the surfaces of objects. The sensors provide unique information about objects in a common scene that when unified may improve object discrimination and classification as a result of added information. The importance of sensor fusion in this regard is demonstrated by Popescu & Wynne (2004) who improve individual tree measurements from lidar data by discriminating between forest types using fusion data.

Second, object discrimination of two-dimensional pixels on a grid is generally an easier problem to solve than for lidar points because pixels have uniform topology. Often an artifact of sparse objects in a three-dimensional scene, the spatial associations between points in a point cloud are highly irregular compared to those of pixels on a grid. The fusion of these data creates new topologies between lidar points and pixels which can be exploited in discrimination. For example, Strom et al. (2010) use topology to segment fusion data by extending camera-only graphical methods. The added dimension of optical depth alone is shown to improve discrimination of pixels into objects (Zhang et al. 2010).
Third, repeated measurements by camera sensors integrated with a point cloud provide a time-series of pixel data for objects in a scene. The point cloud data provide physical context to objects while time-series pixel data provide temporal and spectral context to objects. The added spectral-time dimensions from repeated measurements represent new information for object discrimination and classification.

Although image-lidar fusion is important for these multiple reasons, it is a difficult problem to solve. Difficulty arises from the geometric transformations between the multiple coordinate systems of the pixels and the point cloud. The problem is underscored when the camera and laser scanner are separate on independent platforms and measurements of a scene are acquired under different conditions (Neumann et al. 2003). For example, time-series pixel data of a scene may be acquired over the course of several years while the point cloud is acquired at a single point in time. Objects in the scene may change location throughout the time-series. Other problems arise as well, including imagery with overlapping fields of view and object occlusion. The problem of occlusion in image-lidar fusion are documented (Sohn & Dowman 2007; L. Chen et al. 2005; Zhang et al. 2010; Lim 2011; Ramalingam & Taguchi 2012). Image-lidar fusion is formally presented and discussed in section 4.3, as are the multiple problems and applications of image-lidar fusion.

Generally, published research on image-lidar fusion is in three applications: surface modeling of objects, object detection and object measurement. Common in the literature is the reconstruction of building surfaces from fusion data (L. Chen et al. 2008; L. Chen et al. 2004; Huber et al. n.d.; Kokkas & Dowman 2006; Lim 2011) and building detection (Rottensteiner et al. 2005; Sohn & Dowman 2007; Fraser et al. 2002; Vu et al. 2009). In robotics, obstacle detection and classification are applications of fusion data (Nguyen 2010; Lim & Suter 2008; Lim 2011; Posner et al. 2009; Blanc et al. 2004; Adams 2010). In resource assessment, camera and laser scanner data are fused to identify and measure attributes of trees and forests (Z. Wang et al. 2007; Suárez et al. 2005; Kellndorfer et al. 2010; L. Chen et al. 2005; Popescu & Wynne 2004; Koetz et al. 2007).
These applications vary in approach and offer solutions for specific formulations of the general problem: sensor fusion from a shared platform or from an orthographic view. Section 4.3.2 summarizes these approaches and describes related work in the literature. Section 4.4 generalizes the application-specific literature on the problem in the contexts of uncertainties and probabilities. In these contexts, a new estimator is proposed as the probability fusion model in sections 4.5, 4.6 and 4.6.3.
4.2 Vector Space Definitions

This chapter presents a probabilistic model for the fusion of camera imagery to a lidar point cloud. In order to effectively communicate this model, it is necessary to define images, the point cloud and their geometric relationships with mathematical notation. The probability fusion model presented in this chapter follows from a theoretical framework of the image-lidar fusion problem.

The problem is defined in three vector spaces: object, image and parameter. As discussed in section 4.4, component probabilities of the model are measured by integrals over regions in these spaces.

As specified by the image ray in Chapter Two, the object and image spaces are closely related; image space is a plane in object space onto which object space is projected. Let object space be that which spans the lidar point cloud, the point cloud matrix (sometimes referred to simply as the point cloud) denoted by \( P \in \mathbb{R}^{3 \times N} = [p_1 \ p_2 \ p_3 \ \ldots \ p_N] \) in an absolute coordinate system. In a point cloud, a single lidar point \( p^T = (x, y, z) \) represents the measured coordinate where an energy pulse reflects from the surface of an object. The total number of points in a cloud is denoted by \( N \). It is important to note that the point cloud matrix is denoted in capital-bold while the probability measure \( P \) is denoted in capital-italic; these two entities are different.

Likewise, image space spans the image pixels, the pixels denoted by the pixel matrix \( S \in \mathbb{R}^{2 \times M} = [s_1 \ s_2 \ s_3 \ \ldots \ s_M] \) where \( s^T = (i, j) \) is the coordinate of a pixel on the image plane. The image frame is a specific coordinate system with origin in the region of the image plane that contains the entire pixel matrix. Complementary to the image space, the attribute space is that which spans the attributes of each pixel in the pixel matrix. The pixel attribute matrix is \( B \in \mathbb{R}^{L \times M} = [b_1 \ b_2 \ b_3 \ \ldots \ b_M] \) where a pixel attribute \( b^T = (b_1, b_2, b_3, \ldots, b_L) \) is of dimension \( L \).
The parameter space spans the possible values of the pose vector \( \psi^T = (\alpha, \beta, \gamma, x_\lambda, y_\lambda, z_\lambda) \) where the values of \( \theta^T = (\alpha, \beta, \gamma) \) correspond to yaw, pitch and roll, respectively; and, where \( \lambda^T = (x_\lambda, y_\lambda, z_\lambda) \) is the location of the projective center of the camera in object space. The pose vector, in addition to internal camera parameters, defines the image frame and the projective geometry of image and object spaces relative to each other. The elements of the pose vector are described in Chapter Three.
4.3 Image-Lidar Fusion

Image-lidar fusion is the integration of pixel attributes and lidar points to generate a new, informative dataset usually with higher dimension or greater topology than an image matrix or point cloud alone. The resultant data are called fusion data which are a pair \( (\hat{b}, \hat{p}) \) for each lidar point \( p \) in the point cloud. Applications of image-lidar fusion include attributing spectral information from pixel attributes to lidar points; classifying lidar points into discrete objects as recognized by their context in imagery; and, constructing surface models from lidar points based on the analysis of surface continuation in imagery (such as image segmentation).

The process of image-lidar fusion must consider the relative geometry of the camera and lidar sensors. This geometry is defined by the location and orientation of the camera in object space, called camera pose (see Chapter Three for a discussion of camera pose). It can be described as a transformation of coordinate systems from the relative system of the image to the absolute system of the point cloud.

One such transformation is the image ray

\[
\nu = \lambda + \tau Rg
\]

initially defined in Chapter Two where \( \lambda \) and \( R \) are the projective center and orientation of the camera sensor, respectively. The elements of the rotation matrix \( R \) are non-linear combinations of \( \theta \). These projective center and orientation parameters are sometimes measured by using a Global Positioning System (GPS) and Inertial Measurement Unit (IMU), respectively. In the absence of measurement, these parameters are estimated by methods such as those presented in Chapter Three, among others (Fischler & Bolles 1981; Adnan Ansar & Kostas Daniilidis 2003; Haralick et al. 1989; Quan & Lan 1999; B. Triggs 1999; J. a. Hesch & Roumeliotis 2011). Together, the projective center and orientation parameters are denoted by the pose vector \( \psi \) (see section 4.2). To maintain consistency with Chapter Three, the measured or estimated pose vector is denoted as \( \hat{\psi} \).
4.3.1 Motivation

A practical model for image-lidar fusion should account for uncertainty in fusion data; it should maximize and quantify the precision of fusion data for subsequent inference. Inference is especially important in the case of object detection and collision avoidance in robotic systems (Blanc et al. 2004). When measured by the GPS and IMU, the pose vector is subject to error in these measurement devices. Likewise the pose vector is subject to sampling error when estimated from the imagery and lidar data. Even minor errors in the pose vector can have a substantial effect on fusion results when $\tau$ is large. From this perspective, the measured or estimated pose vector $\hat{\psi}$ presents a source of transformation uncertainty in the external geometry of the camera.

Another source of uncertainty is the spatial association between lidar points and an image ray. The image ray passing through any lidar point $p$ could be partially (or completely) occluded by other points in $P$ (see Figure 28). This problem is called fusion reduction and affects the precision of fusion data because image rays are occluded. If the model is incorrectly specified without the effect of fusion reduction, it may produce bias and inaccurate fusion data. Further, the effect of fusion reduction is compounded by transformation uncertainty if $\hat{\psi}$ and $\tau$ is large (see Figure 29 and discussion in section 4.5.3).
Figure 28: Potential occlusion of the (a) image ray passing through a point $p$ in the two-dimensional object subspace (empty dots represent points in the cloud, arbitrary units on axes).
Figure 29: Uncertainty region about the (a) image ray passing through a point $p$ in the two-dimensional object subspace (empty dots represent points in the cloud, dotted lines are bounds on the uncertainty region, arbitrary units on axes).
Yet another source of uncertainty is the effect of multiple image rays on the fusion data. This problem is called *fusion duplicity* and arises when a point $\mathbf{p}$ is observed in more than one image frame. Geometrically, camera frames are regions on planes in object space where each plane is defined by $\hat{\mathbf{u}}$ which can take an infinite number of values. Hence there can be an infinite number of images of $\mathbf{p}$.
Often this is the case with aerial imagery along a flight line when two or more images overlap with each other as illustrated in Figure 30. In some cases the solution to fusion duplicity is trivial because the attribute of interest in each image of \( p \) is identical. However the solution is non-trivial when the attribute in each image of \( p \) is a function of camera pose \( \hat{\psi} \), has different values in different images or if \( p \) is occluded in some image frames.

To illustrate the problem of fusion duplicity, consider a scalar attribute \( b \) of a point \( p \) that can be observed in an image frame. For example the attribute could be object class, such as “road” in a classified camera image. If the attribute is independent of camera pose, then it should appear to be the same in each image. In this case the equality \( b = b_i \) should be true for the \( i^{th} \) image of point’s attribute and thus \( b_i = b_k \) for its \( k^{th} \) image. However if we don’t know \( b \) and only observe \( b_i \neq b_k \), what is the true attribute \( b \)? The estimate \( \hat{b} \) is affected by the problem of fusion duplicity.

Further consider an attribute \( b(\hat{\psi}) \) of a point \( p \) that can be observed in imagery where the attribute is a function of the pose vector. For example the attribute could be surface irradiance that varies depending on the camera’s location (projective center) and orientation looking at the point. In this case the observed attribute \( b(\hat{\psi}_i) \neq b(\hat{\psi}_j) \) is likely for the \( i^{th} \) and \( k^{th} \) images and hence any estimate of \( b(\hat{\psi}) \) is uncertain because of fusion duplicity.

Finally, uncertainty is inherent in the point cloud and often in the attribute matrix. The point cloud is measured by the laser scanner, and in many cases its measurements are affected by atmospheric transmission and surface reflectance. Likewise depending on the source of the attribute matrix, the pixel attributes may contain measurement error. For example, the pixel attribute may be irradiance as measured by a camera sensor. Like the laser scanner to the point cloud, the camera sensor contributes to measurement error in the pixel attributes.

**4.3.2 Related Work**

Generally, image-lidar fusion has been applied to building detection, building reconstruction, robotics and resource assessment problems. Application objectives vary depending on
accuracy and precision requirements. For example, the objective in some robotics applications is to discriminate and classify obstacles to robotic vehicular movement with high accuracy, such as in Hwang et al. (2007). In this case, one camera and one laser sensor are fixed to a single platform. The transformation of coordinate systems between the camera and lidar data is simplified by fixing the geometric relationship between the sensors, eliminating some uncertainty in the resultant fusion data. Here, the fusion problem is also simplified by eliminating duplication as there is only one camera. However by using a single camera, occlusion in the fusion data becomes a source of uncertainty which may compromise the objective of obstacle detection. A trade-off exists between duplication and reduction: an object can be duplicated from multiple camera views to eliminate the net effect of occlusion in the fusion data.

In problems of building detection and resource assessment, as in robotics, image-lidar fusion is generally simplified by attempting to eliminate duplication and transformation uncertainty as well. For example, all cited literature applications eliminate duplication by transforming multiple images into a single mosaic or by using a single orthoimage (H. Li et al. 2000; Kokkas & Dowman 2006; Y. Chen et al. 2009; L. Chen et al. 2005; Geerling et al. 2007; Huber et al. n.d.; Z. Wang et al. 2007; L. Chen et al. 2004; Schenk & Csatho 2002; Rottensteiner et al. 2005; Sohn & Dowman 2007). Likewise these applications partially eliminate transformation uncertainty by co-registering imagery on an orthographic plane in the object space. However as evidenced by accuracy assessment in these literatures, omission and commission errors result from inaccuracies in the fusion data caused by misregistration of images or regions of images relative to objects in the point cloud (L. Chen et al. 2004). These errors are most apparent at the boundaries of objects, such as the footprints of buildings or rooflines, where image transformations to the orthographic plane fail to account for object occlusion and object depth relative to the look angle of the camera. Unlike the natural environment, fusion applications in building detection and resource assessment may be easier because buildings are static compared to features in the natural environment.
Based on the approaches presented in these various applications, image-lidar fusion in the literature can be classified into two specific cases: fusion from a shared platform (such as in obstacle detection) and fusion from an orthographic plane. These solutions to the problem of image-lidar fusion mitigate uncertainty in duplication and at the expense of information loss. In the case of duplicity, multiple look angles overcome the problem of occlusion and reduction that may be apparent in a single image. In the case of transformation, fusion data lose information about the sides of objects not visible from the orthographic plane that would otherwise be visible in duplicated, oblique imagery. Likewise, projections of sides of objects onto the orthographic plane may cause registration errors and confuse discrimination and classification results. Furthermore, current methods fail to provide measures of uncertainty in the resultant fusion data which are important when application objectives have high-accuracy and precision requirements. To maximize information in fusion data and provide measures on uncertainty, a theoretical context for the problem of image-lidar fusion is constructed in section 4.4.
4.4 Theory

Image-lidar fusion can be divided into three related parts: fusion transformation, duplication and interpolation. *Fusion transformation* is a function under which a lidar point is an imaging vector in an image plane defined by the pose vector. As the pose vector \( \hat{\Psi} \) can take an infinite number of values, there can be an infinite number of images of a single lidar point under the fusion transformation.

*Fusion duplication* is a process by which the pose vector is sampled. The process may be stochastic in the case where a pose vector is selected using a chance device and a probability distribution \( P(v_{\Psi}) \) where \( v_{\Psi} \) is a random draw from that distribution. If the sampled set of pose vectors \( \{\hat{\Psi}_1, \hat{\Psi}_2, \ldots, \hat{\Psi}_K\} \) has size \( K > 1 \), then the problem of duplicity arises. The process may not generate unique samples, for instance if it is with replacement. Because the pose vector can take an infinite number of values, the sampled set is from an infinite population. However, the sample is necessarily finite as the sample space cannot be completely enumerated.

The problem of duplicity effects all values that are conditional on \( \hat{\Psi} \). For instance the attribute matrix \( B \) is defined for pixels in an image frame. The image frame is defined by \( \hat{\Psi} \) and thus the attribute matrix is conditional on \( \hat{\Psi} \). When \( K > 1 \), then there is a sampled set of attribute matrices \( \{B_1, B_2, \ldots, B_K\} \) corresponding to the sampled set of pose vectors.

*Fusion interpolation* is an estimator of an attribute given the fusion transformation and duplication. An example for the case when \( K = 1 \) in the absence of duplicity, interpolation is to estimate the attribute of a lidar point provided the imaging vector of the lidar point in the image frame containing attributes associated with pixels. The imaging vector may differ from the exact coordinate of any single pixel, so its attribution may be a function of multiple pixels and pixel attributes in the image frame. Likewise if \( K > 1 \), then the it may be a function of multiple image frames.
From a statistical perspective when fusion duplicity is a stochastic process and $K > 1$, pose vectors are sampling units from which pixel attributes are observed. As the estimator of an attribute of a lidar point, fusion interpolation is conditional on the observed attributes of all pixels in the sample.

This section develops a theoretical context for the problem of image-lidar fusion by describing sources of uncertainty as probabilistic events (section 4.4.1) and events as they relate to fusion interpolation (section 4.4.2).

4.4.1 Sources of Uncertainty as Events

To perform reasonable inference on fusion data, it is necessary to measure the uncertainty in the fusion data: the multiple sources of uncertainty should be unified in theoretical context. As presented in section 4.3, sources of uncertainty include:

1. The pose vector from measurement error or estimation;
2. Duplicity of the pose vector;
3. Reduction in the fusion transformation;
4. The attribute matrix from measurement error or estimation; and
5. The point cloud.

With regard to duplicity of the fusion data, consider again the statistical perspective of image frames as a sample (see section 4.4). If fusion duplicity is a stochastic process, then a probability distribution is used to select the image frames via the pose vectors and the subsequently the pixel attributes in the image frames. Uncertainty in the duplicity of the pose vector arises from the probability distribution on the selection of the sampling units.

This formulation of image-lidar fusion is in the context of probabilities; the uncertainty of the fusion should be measured as the probability of observing the fusion data conditional on the multiple sources of uncertainty. The probability of observing these multiple sources of uncertainty can be measured by a joint probability distribution on events with uncertain outcomes:
1. That the measured or estimated pose vector is within some region of parameter space denoted by \( \hat{\Psi} \);
2. That a pixel is selected in an image frame denoted by \( s^* \);
3. That the lidar point is not occluded denoted by \( \delta \);
4. That a measured or estimated attribute is within some region of attribute space denoted by \( b \);
5. That a lidar point is within some region of object space in the scene denoted by \( p \).

The pose vector is denoted with a “hat” for consistency with Chapter Three where it is estimated from data. Regions about these events may be defined a variety of ways, including by threshold on the event. For example, the region for the first event \( \hat{\Psi} \) listed above could be specified as \( \pm \) one degree. The probability of the event is measured by taking the integral of the density over the region. The boundaries of the regions should be specified by an acceptable level of allowable error; the smaller the region, likely smaller the probability.

To define the joint probability of these events, consider the case of \( K = 1 \) where there is a single pose vector \( \hat{\Psi} \) and a single pixel \( s \) with attribute \( b \). In this case there is only one sampling unit, the pose vector \( \hat{\Psi} \) and thus the only selection event \( \nu_{\Psi} \). Ignoring the uncertainty in the point cloud, the joint probability of observing all events is \( P(\hat{\Psi}, \delta, b, p, s, \nu_{\Psi}) \). In practice, this joint probability is difficult to compute because its density may be very complex or unknown in closed form. However using Bayesian conditional independence, the joint probability can be expressed as the product of conditional probabilities.

Restated by the conditional independence of the pose vector, the joint probability becomes

\[
P(\hat{\Psi}, \delta, b, p, s^*, \nu_{\Psi}) = P(\hat{\Psi}|\delta_0, b, p, s^*, \nu_{\Psi})P(\delta_0, b, p, s^*, \nu_{\Psi}).
\]  

(94)

The pose vector is not conditional any other events beside the selection event, so again by conditional independence
\[ P(\hat{\psi}, \delta, b, p, s^*, u_\psi) = P(\hat{\psi}|u_\psi)P(\delta|\hat{\psi}, p, u_\psi)P(b, p, s^*, u_\psi) \]

\[ = P(\hat{\psi}|u_\psi)P(\delta|\hat{\psi}, p, u_\psi)P(b, p, s^*, u_\psi). \]

The event that the lidar point is occluded depends on whether the lidar point is of the true surface of an object, the pose vector and that the pose vector is selected. The joint probability then becomes

\[ P(\hat{\psi}, \delta, b, p, s^*, u_\psi) = P(\hat{\psi}|u_\psi)P(\delta|\hat{\psi}, p, u_\psi)P(b, p, s^*, u_\psi) \]

\[ = P(\hat{\psi}|u_\psi)P(\delta|\hat{\psi}, p, u_\psi)P(b, p, s^*, u_\psi)P(p, s^*, u_\psi). \]

The pixel attribute is independent of all events except selected pixel\(^2\), so

\[ P(\hat{\psi}, \delta, b, p, s^*, u_\psi) = P(\hat{\psi}|u_\psi)P(\delta|\hat{\psi}, p, u_\psi)P(b|s^*, u_\psi)P(p, s^*, u_\psi) \]

\[ = P(\hat{\psi}|u_\psi)P(\delta|\hat{\psi}, p, u_\psi)P(b|s^*, u_\psi)P(p|s^*, u_\psi)P(s^*, u_\psi) \]

and the lidar point is independent of the selected pose vector assuming that the camera and lidar sensors are on independent platforms:

\[ P(\hat{\psi}, \delta, b, p, s^*, u_\psi) = P(\hat{\psi}|u_\psi)P(\delta|\hat{\psi}, p, u_\psi)P(b|s^*, u_\psi)P(p)P(s^*, u_\psi) \]

\[ = P(\hat{\psi}|u_\psi)P(\delta|\hat{\psi}, p, u_\psi)P(b|s^*, u_\psi)P(p)P(s^*|u_\psi)P(u_\psi). \]

\(^2\) In reality, the pixel attributes are not independent of each other as there likely exists some autocorrelation in the domain of the image plane. Independence is assumed to simplify the problem.
From (98), the conditional probabilities of the events associated with the sources of uncertainty are apparent:

1. $P(\hat{v}\mid v_\psi)$;
2. $P(s^*\mid v_\psi)$;
3. $P(\delta\mid \hat{v}, P, v_\psi)$;
4. $P(b\mid s^*, v_\psi)$;
5. $P(p)$.

Recall from section 4.2 that $P$ is the point cloud matrix. These events are not necessarily independent of each other. For example, the event that the lidar point is not occluded depends on the event that lidar points are of the true surfaces of objects; if a lidar point is uncertain, then there is the possibility that it is indeed occluded. Likewise, many of these events are conditional on the event that a pose vector or pixel is selected. This is reflected by the selection probability $P(v_\psi)$ in these statements. From a sampling perspective, $P(v_\psi)$ is the selection probability of the selection event which is always observed given that a sample is selected. The selection probability is discussed further in section 4.4.2.

This joint probability is expressed as a Directed Acyclic Graph (DAG) in Figure 31. A similar DAG expresses the case of fusion duplicity in Figure 32 where $v_\psi_i$ is the event that the $i^{th}$ pose vector is selected and $b_{i,j}$ is the observed attribute of the $j^{th}$ pixel in the $i^{th}$ image frame. Ultimately, the joint probability should be expressed as the probability of the estimated fusion data. As formulated in section 4.4.2, uncertainty in the fusion data is measured by the fusion probability which naturally effects the fusion interpolation.
4.4.2 Fusion Probability

Five events that correspond to sources of uncertainty in image-lidar fusion are presented in section 4.4.1. Uncertainty in each source is measured by the probability of observing an outcome of its event. To unify the theory of image-lidar fusion in the context of probability, it is
necessary to define a single measure of probability on the event that the estimated fusion data are the true fusion data.

When \( K > 1 \) and there are multiple pixels in the image frame \((m > 1)\), the marginal probability of observing all events for the sampling unit corresponding to the \( j^{th} \) pixel in the \( i^{th} \) image frame is

\[
P\left(\tilde{\psi}_i, \delta_i, b_{i,j}, P, s_{i,j}^*, u_{\psi_i}\right) = P\left(\tilde{\psi}_i \mid u_{\psi_i}\right)P\left(\delta_i \mid \tilde{\psi}_i, u_{\psi_i}, P\right)P\left(b_{i,j} \mid s_{i,j}^*, u_{\psi_i}\right)P\left(P\right)P\left(s_{i,j}^* \mid u_{\psi_i}\right)P\left(u_{\psi_i}\right).
\]

where \( u_{\psi_i} \) is the event that the \( i^{th} \) pose vector is selected and \( b_{i,j} \) is the observed attribute of the \( j^{th} \) pixel in the \( i^{th} \) image frame (see Figure 32). It is important to note that the \( b_{i,j} \) may not be independent of each other in some cases for which (99) should be revised to reflect this condition. From a sampling perspective in a probability context, selection of the sampling unit means we observe the outcome of the selection event \( u_{\psi_i} \). Hence by Baye’s rule, the joint probability of the marginal becomes

\[
P\left(\tilde{\psi}_i, \delta_i, b_{i,j}, P, s_{i,j}^*, u_{\psi_i}\right) = \frac{P\left(u_{\psi_i} \mid \tilde{\psi}_i, \delta_i, b_{i,j}\right)P\left(\tilde{\psi}_i\right)P\left(\delta_i \mid \tilde{\psi}_i, P\right)P\left(b_{i,j} \mid s_{i,j}^*, \tilde{\psi}_i\right)P\left(P\right)P\left(s_{i,j}^* \mid u_{\psi_i}\right)P\left(u_{\psi_i}\right)}{P\left(u_{\psi_i}\right)}
\]

which is denoted as the fusion probability. In the fusion probability denoted in (100), the term 
\( P\left(u_{\psi_i} \mid \tilde{\psi}_i, s_{i,j}^*, \delta_i, b_{i,j}\right) = 1 \) because given the sampling unit, the outcome of the selection event and the pixel is known and disappears from the equation. The fusion probability is for events conditional on observing the selection event.
The fusion data are the estimated attributes for each lidar point in the point cloud and the point cloud itself. When \( K > 1 \) and there are multiple pixels in the image frame, the marginal uncertainty in the fusion data pair \((\hat{b}_i, p)\) associated with observing the sampling unit is measured by

\[
P(\hat{b}_i, p|\Psi_i, \delta_i, b_{i,j}, P, s^*_{i,j}, u_{\psi_i}) = P(\hat{b}_i|\Psi_i, \delta_i, b_{i,j}, P, s^*_{i,j}, u_{\psi_i})P(p)
\]

which is the probability of observing the pair conditional on the sources of uncertainty in the sampling unit as illustrated by (100). The probability of observing the estimated attribute is independent of the lidar point and the lidar point is independent of all other events.

To understand this event, consider again the statistical perspective of image frames defined by pose vectors as a sample (see section 4.4.1). For example, if the only source of uncertainty in the fusion data were the chance device used to select the pose vector and the \( m \) pixels in the corresponding image frame, then the statement in (100) becomes \( P(\hat{b}_i|u_{\psi_i}) \). The probability distribution used to select a simple random sample (SRS) is uniform and each selection event is independent of all other selection events. Under SRS, the fusion probability is then

\[
P(\Psi_i, \delta_i, b_{i,j}, P, s^*_{i,j}|u_{\psi_i}) = \frac{1}{P(u_{\psi_i})}.
\]

In this example, \( b_{i,j} \) is known exactly and is not a random variable.

A simple estimator for the attribute is the sample mean

\[
\bar{b} = \frac{1}{Km} \sum_{i=1}^{K} \sum_{j=1}^{m} b_{i,j}
\]
where each of \( K \) selected image frames contain \( m \) selected pixels. In this example, Lohr (2010) shows that the sample mean is unbiased and consistent. This estimator can be re-parameterized in a variant form of a Horvitz-Thompson estimator (Horvitz & Thompson 1952) by a weight \( w_{i,j} \) and a normalizing constant \( c \) as

\[
\bar{b} = \frac{1}{c} \sum_{i=1}^{K} \sum_{j=1}^{m} w_{i,j} b_{i,j}
\]

where the weight is

\[
w_{i,j} = \frac{1}{P(v_{\psi_i})},
\]

and normalizing constant is

\[
c = \sum_{i=1}^{K} \sum_{j=1}^{m} w_{i,j}
\]

\[
= \sum_{i=1}^{K} \sum_{j=1}^{m} \frac{1}{P(v_{\psi_i})}
\]

\[
= \frac{Km}{P(v_{\psi_i})}.
\]

If \( b_{i,j} \) is not known exactly and is in fact a random variable, then the fusion probability is

\[
P(\bar{\psi}_i, \delta_i, b_{i,j}, \mathbf{P}, s^*_{i,j}|v_{\psi_i}) = \frac{P(b_{i,j}|s^*_{i,j})}{P(v_{\psi_i})}
\]

The weight becomes

\[
120
\]
Substituting the weight from (108) into the estimator gives

\[ w_{i,j} = \frac{P(b_{i,j} | s_{i,j}^*)}{P(v_{\Psi_i})} \]  

which is an Inverse Distance Weight (IDW) estimator where the weight is the probability measure on the pixel attributes with power parameter equal to one (Shepard 1968). Because \( P(b_{i,j} | s_{i,j}^*) \leq 1 \), increasing the power parameter effectively “sharpens” the distribution of \( b_{i,j} \) and gives more weight to those pixel attributes with relatively high probability (see Figure 33).
Some alternate forms of IDW define the weight as the Euclidean distance between a prediction point \(\hat{s}\) and a pixel with a known attribute (Babak & Deutsch 2008) on the image plane. In this form, the weight \(P(b_{i,j} | s_{i,j}^*) = P(b_{i,j} | s_{i,j}^*, \hat{s})\) is defined as being proportional to the magnitude of the vector difference between the prediction point and sampled pixel on the image plane in all directions; it assumes the attribute is isotropic (Babak & Deutsch 2008). Increasing the power parameter in this form effectively gives more weight to those pixels that are near the prediction point.

Because the specific estimator in (109) is not conditional on a prediction point, it is not predicated on any assumptions of stationarity in the attribute on the image plane; observing a pixel attribute conditional on the selected pixel implies that all observed attributes are
independent of each other. This result derives from SRS sampling theory that given an independent selection event, the sampling units are independent of each other (Lohr 2010). Conditional independence of attributes is an important feature of the fusion probability.

Following from (109), the IDW estimator for the full fusion probability specified by (100) is

\[
\hat{b} = \frac{\sum_{i=1}^{K} \sum_{j=1}^{m} P(\hat{\psi}_i) P(s_{i,j}^*) P(\delta_i|\hat{\psi}_i, p, P) P(b_{i,j}|s_{i,j}^*) P(P)b_{i,j}}{\sum_{i=1}^{K} \sum_{j=1}^{m} P(\hat{\psi}_i) P(s_{i,j}^*) P(\delta_i|\hat{\psi}_i, p, P) P(b_{i,j}|s_{i,j}^*) P(P)}
\]

(110)

\[
= \frac{\sum_{i=1}^{K} \sum_{j=1}^{m} P(\hat{\psi}_i) P(\delta_i|\hat{\psi}_i, p, P) P(b_{i,j}|s_{i,j}^*) b_{i,j}}{\sum_{i=1}^{K} \sum_{j=1}^{m} P(\hat{\psi}_i) P(\delta_i|\hat{\psi}_i, p, P) P(b_{i,j}|s_{i,j}^*)}
\]

under the assumption that the pixel is without error so that \(P(s_{i,j}^*) = 1\).

The estimator is intuitive, as it down weights attributes where the fusion probability is low. It also shows that the probability of observing the point cloud does not affect the estimate as \(P(P)\) is in both the denominator and numerator, and is independent of the sample. Under SRS, the resultant weights are the product of three probabilities: the pose probability \(P(\hat{\psi}_i)\), reduction probability \(P(\delta_i|\hat{\psi}_i, p, P)\) and attribute probability \(P(b_{i,j}|s_{i,j}^*)\). The problem of image-lidar fusion can be narrowed to the occlusion probability under SRS upon assumption that the pose vector and attribute matrix are without error.

Likewise, the problem can be expanded to account for un-equal probability samples in contrast to the SRS sampling scheme. For instance, the probability of selection may be conditional on the trajectory vector of the camera platform.

Finally, the fusion probability can be applied to estimators other than the sample mean. Other estimators, such as the probability fusion model discussed in Section 4.5, may be more efficient than the sample mean.
4.5 Probability Fusion Model

The probability fusion model derives from the fusion probability presented in section 4.4. The model is an estimator of the attribute \( \hat{b} \) of the fusion data pair \( \hat{b}, p \) for a lidar point \( p \) in the presence of duplicity. The sample mean in (14) is defined so that the observed pixel attributes \( b_{i,j} \) are independent of each other when conditioned on the pixels \( s_{i,j}^* \) in the image frame.

However unlike the sample mean, the probability fusion model exploits the spatial relationship of each \( s_{i,j}^* \) relative to a prediction point \( \hat{s}_i \) on each image plane. The prediction point is the image of \( p \) given the pose vector \( \hat{\psi}_i \) for the \( i^{th} \) image frame.

Specifically, the relationship is expressed as the vector difference between the pixel and prediction point as

\[
\hat{s}_{i,j} = s_{i,j}^* - \hat{s}_i
\]

where \( s_{i,j}^* \) is known without error as initially assumed in the derivation of the sample mean in section 4.4.2. Substituting the imaging vector from the object to image space transformation for the prediction point, the difference becomes

\[
\hat{s}_{i,j} = s_{i,j}^* - \frac{1}{t} \hat{R}_i^{-1}(p - \hat{\lambda}_i) + \omega
\]

which shows it is conditional on the pose vector \( \hat{\psi}_i \) and the lidar point \( p \). Recall from Chapter Three that the rotation matrix and projective center are defined by the pose vector. The probability of observing this vector difference is the transformation probability \( P(\hat{s}_{i,j}|\hat{\psi}_i, p) \).

As specified in the probability fusion model, the marginal uncertainty in the fusion data pair \( \hat{b}, p \) associated with observing the sampling unit is measured by
\[ P(\hat{\mathbf{b}}, \mathbf{p} | \hat{\psi}_i, \delta_i, \mathbf{b}_{i,j}, \mathbf{P}, \mathbf{s}_{i,j}^*, \hat{s}_{i,j}, \mathbf{v}_{\psi_i}) = P(\hat{\mathbf{b}} | \hat{\psi}_i, \delta_i, \mathbf{b}_{i,j}, \mathbf{P}, \mathbf{s}_{i,j}^*, \hat{s}_{i,j}, \mathbf{v}_{\psi_i}) P(\mathbf{p}) \]  

which is conditional on the specific fusion probability

\[ P(\hat{\psi}_i, \delta_i, \mathbf{b}_{i,j}, \mathbf{P}, \mathbf{s}_{i,j}^*, \hat{s}_{i,j} | \mathbf{v}_{\psi_i}) \]

\[ = \frac{P(\hat{\psi}_i) P(\delta_i | \hat{\psi}_i, \mathbf{P}) P(\mathbf{b}_{i,j} | \mathbf{s}_{i,j}^*) P(\mathbf{p}) P(\mathbf{s}_{i,j}^*) P(\hat{\psi}_i, \mathbf{p}) P(\mathbf{v}_{\psi_i})}{P(\mathbf{v}_{\psi_i})} \]

represented as a DAG in Figure 34, defined similarly to (100).

The probability fusion model is the attribute estimator

\[ \hat{\mathbf{b}} = \frac{\sum_{i=1}^{K} \sum_{j=1}^{m} P(\hat{\psi}_i) P(\delta_i | \hat{\psi}_i, \mathbf{P}) P(\mathbf{b}_{i,j} | \mathbf{s}_{i,j}^*) P(\mathbf{s}_{i,j}^*) P(\hat{\psi}_i, \mathbf{p}) \mathbf{b}_{i,j}}{\sum_{i=1}^{K} \sum_{j=1}^{m} P(\hat{\psi}_i) P(\delta_i | \hat{\psi}_i, \mathbf{P}) P(\mathbf{b}_{i,j} | \mathbf{s}_{i,j}^*) P(\mathbf{s}_{i,j}^*) P(\hat{\psi}_i, \mathbf{p})} \]

where \( P(\mathbf{s}_{i,j}^*) = 1 \) because the pixel is without error. The probability fusion model is weighted by the pose probability \( P(\hat{\psi}_i) \), reduction probability \( P(\delta_i | \hat{\psi}_i, \mathbf{p}, \mathbf{P}) \), attribute probability \( P(\mathbf{b}_{i,j} | \mathbf{s}_{i,j}^*) \) and transformation probability \( P(\hat{s}_{i,j} | \hat{\psi}_i, \mathbf{p}) \). Component probabilities are described in sections 4.5.2, 4.5.3 and 4.5.4, and estimator properties are discussed in section 4.5.5.
4.5.1 Pose Probability

The pose probability $P(\hat{\psi}_i)$ is a measure on the event that an estimated or measured pose vector is the true pose vector of an image frame. There are many possible methods for measuring the pose probability. In the case when $\hat{\psi}_i$ is measured, the GPS and IMU devices often give an estimate of associated standard error. Under some parametric assumptions, the pose probability can be estimated. For example, the pose vector can be assumed to have zero-mean, additive Gaussian noise as

$$\hat{\psi}_i = \psi_i + \epsilon_{\psi_i}$$  \hspace{1cm} (116)

where $\epsilon_{\psi_i} \sim \mathcal{N}(0, \Sigma_{\psi_i})$ and the variance-covariance matrix $\Sigma_{\psi_i}$ with the variance as the squared-standard error on its diagonal and assumed covariance in $\epsilon_{\psi_i}$ elsewhere. Parametric assumptions of distribution and covariance could be confirmed using experimental design or simulation study.
In the case when \( \hat{\psi}_i \) is estimated, non-parametric resampling methods such as those presented in Chapter Three can be used to estimate the pose probability. For example, the distribution of \( \hat{\psi}_i \) can be observed by resampling the tie points used to estimate the pose vector. To estimate the probability, the observed mass for \( \hat{\psi}_i \) can be divided by the sum of all observed mass.

The probability fusion model gives more weight to pose vectors with higher probability (lower uncertainty) than lower probability. If all pose probabilities are identical, then the model gives equal weight to all data and the pose probabilities can be ignored. When working with a large dataset of image frames, it may be reasonable to ignore the pose probabilities if a subsample of pose probabilities infers their identity.

### 4.5.2 Attribute Probability

The attribute probability \( P(b_{i,j} | s^*_{i,j}) \) is a measure on the event that the estimated or measured attribute is the true attribute of a pixel. In some cases of image-lidar fusion, the estimated attribute may be assumed to be the true attribute of the pixel. This may be a reasonable assumption if the measurement error in the attribute is negligible or if inference on the fusion data is without consideration to the uncertainty in pixel attributes.

In other cases, the attribute probability is important and should not be ignored. For example, the inference on the fusion data might be whether a lidar point is a class of object. Object class is discriminated as an attribute of a pixel and \( P(b_{i,j} | s^*_{i,j}) \) is the probability that the pixel attribute is of the object class. Then to infer whether the lidar point is a class of object, the significance of its fusion data pair \((\bar{b}, p)\) is tested by \( P(\bar{b}, p) \). The probability fusion model gives more weight to pixel attributes with higher probability (lower uncertainty) than lower probability (higher uncertainty).

### 4.5.3 Transformation Probability

The transformation probability \( P(\hat{s}_{i,j} | \hat{\psi}_i, p) \) is a measure on the event that the image \( \hat{s}_i \) of a lidar point \( p \) is the sampled pixel \( s^*_{i,j} \) given the pose vector \( \hat{\psi}_i \). As defined in (112), this can be interpreted using the conditional distribution of the imaging vector \( \hat{s}_i \) in the \( i^{th} \) image frame.
relative to each $s_{i,j}^r$. Note that $\hat{s}_{i,j} = 0$ when $\hat{s}_i = s_{i,j}^*$ and therefore the distribution of $\hat{s}_{i,j}$ is assumed to have a zero mean. Figure 35 shows levels of an example distribution in an image frame relative to selected pixels. The estimator assigns more weight to attributes with pixels at relatively high levels of this distribution.

![Probability Levels in an Image Frame](image)

Figure 35: Probability levels of an example distribution on $\hat{s}_i$ in an image frame relative to pixels (+).

The relationship between the imaging vector and $(\hat{\psi}_i, p)$ is the object to image space transformation as defined in (112). Through this transformation and under certain regularity conditions discussed below, the transformation probability can be defined largely by the uncertainty in rotation of the image plane in the object space, ignoring the uncertainty in the camera projective center and the point cloud.
Assuming that the distribution of $\hat{s}_i$ has finite second-order moment, the distribution and probability levels of $\hat{s}_{i,j}$ are defined by the transformation variance

$$ \text{var}(\hat{s}_{i,j}) = \text{var} \left( \hat{s}_{i,j}^* - \frac{1}{\tau} \hat{R}_i^{-1}(p - \hat{\lambda}_i) + \omega \right) $$

$$ = \text{var} \left( \frac{1}{\tau} \hat{R}_i^{-1}(p - \hat{\lambda}_i) \right) $$

$$ = \text{var} \left( \hat{R}_i^{-1}(p - \hat{\lambda}_i) \right) \tau^{-2} $$

because the pixel and $\hat{s}_{i,j}^*$ and internal camera parameters $\omega$ are assumed fixed and not random. Note the transformation is that from the object to image space defined by (35).

The transformation variance defined in (117) is complex as it includes those sources from the projective center $\hat{\lambda}_i$, flight trajectory parameters $\hat{\theta}_i$ and point cloud. The significance of these sources may differ, and under certain conditions some sources could be ignored because they are relatively insignificant to the other sources. To simplify the transformation variance, it can be divided into two components as

$$ \text{var}(\hat{s}_{i,j}) = \text{var} \left( w_A \hat{R}_i^{-1}(p - \hat{\lambda}_i) + w_B \hat{R}_i^{-1}(p - \hat{\lambda}_i) \right) \tau^{-2} $$

$$ = \text{var} \left( \hat{R}_i^{-1}(p - \hat{\lambda}_i) \right) w_A^2 \tau^{-2} + \text{var} \left( \hat{R}_i^{-1}(p - \hat{\lambda}_i) \right) w_B^2 \tau^{-2} $$

where the weights $w_A + w_B = 1$. This formulation is mathematically equivalent to (117). Then considering the case when the variance of the flight trajectory parameters $\hat{\theta}_i$ can be quantified in the first term and the other sources $(p - \hat{\lambda}_i)$ in the second term, the transformation variance becomes
\[ \text{var}(\hat{s}_{i,j}) = \text{var}(\hat{R}_i^{-1} f_i)w_A^2 \tau^{-2} + \text{var}(\hat{R}_i^{-1}(p - \hat{\lambda}_i))w_B^2 \tau^{-2} \]  

(119)

where in the first term the variable \( f_i = p - \hat{\lambda}_i \) specially denotes that its value is fixed as opposed to \( p - \hat{\lambda}_i \) generally which is random and in the second term only the quantity of \( p - \hat{\lambda}_i \) is random.

Proceeding from (119), the transformation variance becomes

\[
\text{var}(\hat{s}_{i,j}) = \text{var}(\hat{R}_i^{-1} f_i)w_A^2 \tau^{-2} + \text{var}(\hat{R}_i^{-1}(p - \hat{\lambda}_i))w_B^2 \tau^{-2} \\
= \text{var}(\hat{R}_i^{-1} f_i)w_A^2 \tau^{-2} + w_B^2 \tau^{-2} \hat{R}_i^{-1} \text{var}(p - \hat{\lambda}_i)(\hat{R}_i^{-1})^T \\
\text{var}(\hat{R}_i^{-1} f_i)w_A^2 \tau^{-2} + w_B^2 \tau^{-2} \hat{R}_i^{-1} \text{var}(p - \hat{\lambda}_i)R_i
\]

because \( \hat{R}_i^{-1} = R_i^T \) is orthogonal with determinant one.

As expressed by (120), the relative significance of the rotational variance \( \text{var}(\hat{R}_i^{-1} f_i) \) and distance variance \( \text{var}(p - \hat{\lambda}_i) \) are partially a function of the weights \( w_A \) and \( w_B \). By separating the variance of the flight trajectory parameters from the other sources, the transformation variance is redefined as a linear combination of the rotational variance and the distance variance.

As demonstrated in section 4.5.3.1, if \( \tau \) is large relative to \( \text{var}(p - \hat{\lambda}_i) \), then the final term in (120) is small. The effect of \( \tau \) on the probability levels in the image frame as a sole function of the distance variance is illustrated in Figure 36 where \( \text{var}(p - \hat{\lambda}_i) \) is fixed and \( \tau \) varies: the lidar point \( p \) appears further away from the image plane as \( \tau \) increases. Thus, if \( \tau \) is sufficiently large, distance variance can be ignored in the computation of the transformation probability.
Conversely, simplifying the transformation variance as a linear combination of the rotational variance and the distance variance may not be reasonable under certain conditions, such as when the optical depth $\tau$ is small relative to the magnitude of the distance variance.

![Figure 36: Example probability levels as a function of distance variance by increasing optical depth (a-d) in image frames relative to $\hat{s}_i$ (+).](image)

With regard to rotational variance, the elements of the rotation matrix, defined in Chapter Three, are not linear in $\hat{\theta}_i$ and therefore the variance of this term is difficult to express in closed form. Further, the rotation matrix expresses $\hat{\theta}_i$ as combinations of periodic functions, sine and
cosine. Periodicity in the rotation matrix prohibits the formulation of variance approximations by such methods as Dahlquist & Bjorck (2007) as derivatives of the periodic functions themselves are periodic.

As with the pose probability in section 4.5.1, one approach to measuring the transformation probability is to use non-parametric methods. In this approach, the probability mass of each $\hat{s}_{i,j}$ is estimated by sampling the distributions of $\hat{\psi}_i$ and $p$. Alternatively, an approximate density function can be used to measure the transformation probability when the distance variance can be ignored, its second moment defined only by the rotational variance. Such an approximate density function is explored in section 4.6.1

Upon computation of the rotational variance term, a pivotal quantity can be employed to test $P(\hat{s}_{i,j}|\hat{\psi}_i, p)$. If $\hat{s}_{i,j}$ is approximately normal and $\text{var}(\hat{R}_i^{-1}f_i)$ is approximately Wishart on the image plane, one such test is the two Student’s t-test (Hotelling 1931).

4.5.3.1 Optical Depth Covariance Ratio

As identified in section 4.5.3, the transformation probability can be measured by the distribution of $\hat{s}_{i,j}$ on the image plane. This distribution is assumed to have zero mean and finite second-order moment. Under some conditions, it is reasonable to define the second moment of this distribution solely by the rotational variance of $p$ if the inverse-squared optical depth $\tau^{-2}$ is large relative to the variance $\text{var}(p - \hat{\lambda}_i)$ for a lidar point $p$ and camera projective center $\hat{\lambda}_i$.

One possible measure of this condition is the optical depth covariance (ODC) ratio for a lidar point $p$ relative to the $i^{th}$ image plane

$$\rho_\tau = \frac{f^2|\text{var}(p) + \text{var}(\hat{\lambda}_i)|}{\|p - \hat{\lambda}_i\|^2} \geq 0$$  \hspace{1cm} (121)

---

3 The Wishart distribution is often used to estimate covariance matrices.
where \(\text{var}(\mathbf{p}) \in \mathbb{R}^{3 \times 3}\) and \(\text{var}(\hat{\lambda}_i) \in \mathbb{R}^{3 \times 3}\) are covariance matrices of \(\mathbf{p}\) and \(\hat{\lambda}_i\), respectively.

The ODC ratio is obtained from the determinant of the distance variance term in (120) as

\[
\left| w_B^2 \tau^{-2} R_i^{-1} \text{var}(\mathbf{p} - \hat{\lambda}_i) R_i \right| = w_B^2 \tau^{-2} \left| R_i^{-1} \text{var}(\mathbf{p} - \hat{\lambda}_i) R_i \right|
\]

\[
= w_B^2 \tau^{-2} \left| \text{var}(\mathbf{p} - \hat{\lambda}_i) \right| \geq 0
\]

because \(|R_i| = 1\) and \(\text{var}(\mathbf{p} - \hat{\lambda}_i)\) are necessarily symmetric, positive semi-definite. Applying the assumption of independence presented in section 4.5.3, \(P(\hat{\lambda}_i, \mathbf{p}) = P(\hat{\lambda}_i)P(\mathbf{p})\) gives

\[
\rho_t = w_B^2 \tau^{-2} \left| \text{var}(\mathbf{p} - \hat{\lambda}_i) \right|
\]

\[
= w_B^2 \tau^{-2} \left| \text{var}(\mathbf{p}) + \text{var}(\hat{\lambda}_i) \right|.
\]

As defined in section 4.5.3, the weight \(w_B \leq 1\) and thus \(w_B^2 \leq 1\); the only effect of the weight is to reduce the quantity on the right-hand side of (123). Assuming the extreme case when \(w_B = 1\), the ODC ratio is

\[
\rho_t = \tau^{-2} \left| \text{var}(\mathbf{p}) + \text{var}(\hat{\lambda}_i) \right|
\]

which potentially over-measures the ODC ratio if in reality \(w_B < 1\).

As defined in Chapter Three, the optical depth is

\[
\tau = \frac{r_3(\mathbf{p} - \hat{\lambda}_i)}{f},
\]

where \(r_3\) is a vector element of \(R_i^{-1}\) and \(f\) is the camera focal length. Taking an approximation in the numerator of (125) gives
that when substituted into (124) gives (121). The ODC ratio is minimized when the focal length is short and \( \| p - \hat{\lambda}_i \| \) is large.

If the covariance structures \( \text{var}(p) \) and \( \text{var}(\hat{\lambda}_i) \) are known \textit{a-priori} to the acquisition of camera imagery, then the focal length and projective center of the camera can be selected to minimize the ODC ratio. Upon minimizing the ODC ratio, the transformation variance can be reasonably approximated by the rotational variance. Under this condition, an approximate density function based on rotational variance is explored in section 4.6.1.

4.5.4 Reduction Probability

The reduction probability \( P(\delta_i | \hat{\psi}_i, P) \) is a measure on the event that a lidar point \( p \in P \) is not occluded by the point cloud matrix \( P \) given the pose vector \( \hat{\psi}_i \). For a lidar point to be occluded, two events must occur:

1) A lidar point \( p_k \neq p \in P \) is in the image ray passing through \( p \); and
2) The same lidar point \( p_k \) is between the projective center \( \hat{\lambda}_i \) and \( p \) in object space.

These events can be expressed in terms of their complements as

1) A lidar point \( p_k \) is \textit{not} in the image ray passing through \( p \), denoted as \( \bar{\delta}_{i,k} \); and
2) The same lidar point \( p_k \) is \textit{not} between the projective center \( \hat{\lambda}_i \) and \( p \) in object space, denoted as \( \bar{\delta}_{i,k} \).

In terms of these complements, the reduction probability is the joint probability of observing these two complementary events for all \( N - 1 \) lidar points in the point cloud.
for the set of lidar points $\mathcal{P}_p = \{p_k \in \mathcal{P}: p_k \neq p\}$ assuming the events are independent of each other. However, this may not be a reasonable assumption as the events of observing two different points $p_k \neq p_j$ may be conditional on a latent random variable. An example of this condition is when the two points are observed as measurements of a highly reflective surface. In this case, the measurements may be conditional on the surface which could be a common source of uncertainty and hence induces conditional dependence between the events.

The event $\bar{\delta}_{i,k}$ occurs when the magnitude of the vector difference between a lidar point $p_k$ and the point closest to the lidar point $v_k$ in the image ray is

$$\|p_k - v_k\| \neq 0$$

such that $p_k = (x_k, y_k, z_k)^T$ is not in the image ray passing through $p = (x, y, z)^T$, as illustrated in Figure 37. Therefore, the probability of observing this event is

$$P(\bar{\delta}_{i,k}) = P(\|p_k - v_k\| \neq 0)$$

$$= 1 - P(\|p_k - v_k\| = 0).$$

The equation for $\|p_k - v_k\|$ is derived from Lay (2002) in Chapter Two as

$$\|p_k - v_k\| = \frac{\|(p_k - \hat{\lambda}_i) \times (p_k - \hat{\lambda}_i)\|}{\|p - \hat{\lambda}_i\|}$$

which implies that the probability is only conditional on $(\hat{\psi}_i, p_k, p)$. From (130), the probability of observing the event $\bar{\delta}_{i,k}$ becomes
Likewise, the event $\delta_{i,k}$ occurs when

$$\|p - \hat{\lambda}_i\| \neq \|p - v_k\| + \|v_k - \hat{\lambda}_i\|$$  \hspace{1cm} (132)$$

where the probability of observing this event is

$$P(\delta_{i,k} | \tilde{\Psi}_i, p_k, p) = 1 - P\left(\frac{\| (p_k - \hat{\lambda}_i) \times (p_k - \hat{\lambda}_i) \|}{\| p - \hat{\lambda}_i \|} = 0 \right).$$  \hspace{1cm} (131)$$
\[ P(\tilde{\beta}_{i,k}|\tilde{\psi}_{i}, p_{k}, p) = P(\|p - v_{k}\| + \|v_{k} - \tilde{\lambda}_{i}\| - \|p - \tilde{\lambda}_{i}\| \neq 0) \]
\[ = 1 - P(\|p - v_{k}\| + \|v_{k} - \tilde{\lambda}_{i}\| - \|p - \tilde{\lambda}_{i}\| = 0). \] (133)

To find the equation of \( v_{k} \) in terms of \((\tilde{\psi}_{b}, p_{k}, p)\), consider the vector equation of the line passing between \( \tilde{\lambda}_{i} = (x_{\tilde{\lambda}_{i}}, y_{\tilde{\lambda}_{i}}, z_{\tilde{\lambda}_{i}})^{T} \) and \( p \) as illustrated in Figure 37:
\[ v = \tilde{\lambda}_{i} + \tau(p - \tilde{\lambda}_{i}) \] (134)

where \((p - v_{k})\) is the direction vector, \( \tilde{\lambda}_{i} \) is the position vector \( \tau \) the position scalar. Also consider the vector differences \((p_{k} - v)\) and \((p - \tilde{\lambda}_{i})\) which are orthogonal when \( v = v_{k} \) such that their dot product is
\[ (p_{k} - v) \cdot (p - \tilde{\lambda}_{i}) = [p_{k} - \tilde{\lambda}_{i} - \tau(p - \tilde{\lambda}_{i})] \cdot (p - \tilde{\lambda}_{i}) = 0. \] (135)

Then finding \( v_{k} \) is equivalent to finding \( \tau = \tau_{k} \), the solution to which is
\[ 0 = [x_{k} - x_{\tilde{\lambda}_{i}} - \tau_{k}(x - x_{\tilde{\lambda}_{i}})](x - x_{\tilde{\lambda}_{i}}) + [y_{k} - y_{\tilde{\lambda}_{i}} - \tau_{k}(y - y_{\tilde{\lambda}_{i}})](y - y_{\tilde{\lambda}_{i}}) \]
\[ + [z_{k} - z_{\tilde{\lambda}_{i}} - \tau_{k}(z - z_{\tilde{\lambda}_{i}})](z - z_{\tilde{\lambda}_{i}}) \]
\[ = (x_{k} - x_{\tilde{\lambda}_{i}})(x - x_{\tilde{\lambda}_{i}}) + (y_{k} - y_{\tilde{\lambda}_{i}})(y - y_{\tilde{\lambda}_{i}}) + (z_{k} - z_{\tilde{\lambda}_{i}})(z - z_{\tilde{\lambda}_{i}}) \]
\[ - \tau_{k}(x - x_{\tilde{\lambda}_{i}})(x - x_{\tilde{\lambda}_{i}}) - \tau_{k}(y - y_{\tilde{\lambda}_{i}})(y - y_{\tilde{\lambda}_{i}}) \]
\[ - \tau_{k}(z - z_{\tilde{\lambda}_{i}})(z - z_{\tilde{\lambda}_{i}}) \]
\[ \Rightarrow \tau_{k}(x - x_{\tilde{\lambda}_{i}})^{2} + \tau_{k}(y - y_{\tilde{\lambda}_{i}})^{2} + \tau_{k}(z - z_{\tilde{\lambda}_{i}})^{2} \]
\[ = (x_{k} - x_{\tilde{\lambda}_{i}})(x - x_{\tilde{\lambda}_{i}}) + (y_{k} - y_{\tilde{\lambda}_{i}})(y - y_{\tilde{\lambda}_{i}}) + (z_{k} - z_{\tilde{\lambda}_{i}})(z - z_{\tilde{\lambda}_{i}}) \]
\[
\Rightarrow \tau_k = \frac{(p_k - \hat{\lambda}_i) \cdot (p - \hat{\lambda}_i)}{\|p_k - \hat{\lambda}_i\|^2}
\]

and the solution to \(v_k\) is

\[
v_k = \hat{\lambda}_i + \tau_k (p - \hat{\lambda}_i) \tag{137}
\]

\[
= \hat{\lambda}_i + (p - \hat{\lambda}_i) \frac{(p_k - \hat{\lambda}_i) \cdot (p - \hat{\lambda}_i)}{\|p_k - \hat{\lambda}_i\|^2}.
\]

From (137), the probability of observing the event \(\tilde{\delta}_{l,k}\) in (133) becomes

\[
P(\delta_{l,k} | \tilde{\Psi}_l, p_k, p) \tag{138}
\]

\[
= 1 - P \left( \left\| p - \hat{\lambda}_i + (p - \hat{\lambda}_i) \frac{(p_k - \hat{\lambda}_i) \cdot (p - \hat{\lambda}_i)}{\|p_k - \hat{\lambda}_i\|^2} \right\| \right)
\]

\[
+ \left\| (p - \hat{\lambda}_i) \frac{(p_k - \hat{\lambda}_i) \cdot (p - \hat{\lambda}_i)}{\|p_k - \hat{\lambda}_i\|^2} \right\| - \|p - \hat{\lambda}_i\| = 0 \right).
\]

From (131) and (138), the joint probability of observing these two events \(\delta_{l,k}\) and \(\tilde{\delta}_{l,k}\) for the \(k^{th}\) lidar point \(p_k\) in the point cloud is
\[ P(\tilde{d}_{i,k}, \tilde{d}_{i,k} | \tilde{\psi}_l, p_k, p) \]

\[ = \left[ 1 - P \left( \left\| \frac{(p_k - \hat{\lambda}_i) \times (p_k - \hat{\lambda}_i)}{\| p - \hat{\lambda}_i \|} = 0 \right\| \right) \right]^1 \]

\[ - P \left( \left\| p - \hat{\lambda}_i + \left( p - \hat{\lambda}_i \right) \frac{\left( p_k - \hat{\lambda}_i \right) \cdot (p - \hat{\lambda}_i)}{\| p_k - \hat{\lambda}_i \|^2} \right\| \right) \]

\[ + \left( \left\| \left( p - \hat{\lambda}_i \right) \frac{\left( p_k - \hat{\lambda}_i \right) \cdot (p - \hat{\lambda}_i)}{\| p_k - \hat{\lambda}_i \|^2} \right\| - \| p - \hat{\lambda}_i \| = 0 \right) \]

\[ = 1 + P \left( \left\| \frac{(p_k - \hat{\lambda}_i) \times (p_k - \hat{\lambda}_i)}{\| p - \hat{\lambda}_i \|} \right\| \right) \]

\[ = 0 \right) P \left( \left\| p - \hat{\lambda}_i + \left( p - \hat{\lambda}_i \right) \frac{\left( p_k - \hat{\lambda}_i \right) \cdot (p - \hat{\lambda}_i)}{\| p_k - \hat{\lambda}_i \|^2} \right\| \right) \]

\[ + \left( \left\| \left( p - \hat{\lambda}_i \right) \frac{\left( p_k - \hat{\lambda}_i \right) \cdot (p - \hat{\lambda}_i)}{\| p_k - \hat{\lambda}_i \|^2} \right\| - \| p - \hat{\lambda}_i \| = 0 \right) \]

\[ - P \left( \left\| \frac{(p_k - \hat{\lambda}_i) \times (p_k - \hat{\lambda}_i)}{\| p - \hat{\lambda}_i \|} = 0 \right\| \right) \]

\[ - P \left( \left\| p - \hat{\lambda}_i + \left( p - \hat{\lambda}_i \right) \frac{\left( p_k - \hat{\lambda}_i \right) \cdot (p - \hat{\lambda}_i)}{\| p_k - \hat{\lambda}_i \|^2} \right\| \right) \]

\[ + \left( \left\| \left( p - \hat{\lambda}_i \right) \frac{\left( p_k - \hat{\lambda}_i \right) \cdot (p - \hat{\lambda}_i)}{\| p_k - \hat{\lambda}_i \|^2} \right\| - \| p - \hat{\lambda}_i \| = 0 \right) \]

The reduction probability becomes the product of (139) over the set \( \mathcal{P}_p \) based on the assumption of independence exploited in (127) as

139
Provided the algebraic complexity of the reduction probability, it may difficult to compute in closed form. This is necessarily the case if the joint distribution on $\tilde{\Phi}_l, P$ cannot be expressed in closed form. Therefore, as with the pose and transformation probabilities, it may be necessary to use non-parametric methods or approximate the density function of $\delta_{l,k}$ to compute this probability. One possible approximation is the product of Bernoulli distributions on $\delta_{l,k}$ and $\tilde{\delta}_{l,k}$. A threshold can be set on outcomes of (128) and (132) where the probability is one if less than the threshold or zero otherwise.

In certain cases when the application of the probability fusion model is known a-priori to avoid occlusion, the reduction probability can be ignored entirely. For example, if the lidar point $p$ is
purposefully imaged to avoid occlusion, then the reduction probability is necessarily one and does not need to be computed.

4.5.5 Statistical Properties

The probability fusion model is defined by (115) and estimates the attribute of the fusion data pair \((\hat{b}, p)\). If the pose vector is sampled using SRS, then the model is unbiased and as efficient as the sample mean. These properties are demonstrated in sections 4.5.5.1 and 4.5.5.3. Section 4.5.5.2 gives the estimator for the standard error of the model. The statistical properties assume that the component probabilities of the model are known given that the pose vector and pixels are selected.

4.5.5.1 Bias

The probability fusion model is unbiased if the pixel attributes \(b_{i,j}\) are unbiased. That is, it is unbiased if \(\text{bias}(b_{i,j}) = 0\) such that \(E(b_{i,j}) = b\) where \(E(b_{i,j})\) is the expectation of the pixel attribute and \(b\) is the true but unknown attribute. The bias of the probability fusion model specified by (115) is

\[
bias(\hat{b}) = E(\hat{b}) - b
\]

\[
= E \left[ \sum_{i=1}^{K} \sum_{j=1}^{m} P(\hat{\psi}_i)P(\delta_i|\hat{\psi}_i, p)P(b_{i,j}|s_{i,j}^*)P(\hat{s}_{i,j}|\hat{\psi}_i, p)b_{i,j} \right] - b
\]

\[
= \frac{\sum_{i=1}^{K} \sum_{j=1}^{m} P(\hat{\psi}_i)P(\delta_i|\hat{\psi}_i, p)P(b_{i,j}|s_{i,j}^*)P(\hat{s}_{i,j}|\hat{\psi}_i, p)E(b_{i,j})}{\sum_{i=1}^{K} \sum_{j=1}^{m} P(\hat{\psi}_i)P(\delta_i|\hat{\psi}_i, p)P(b_{i,j}|s_{i,j}^*)P(\hat{s}_{i,j}|\hat{\psi}_i, p)} - b
\]

\[
= \frac{\sum_{i=1}^{K} \sum_{j=1}^{m} P(\hat{\psi}_i)P(\delta_i|\hat{\psi}_i, p)P(b_{i,j}|s_{i,j}^*)P(\hat{s}_{i,j}|\hat{\psi}_i, p)b}{\sum_{i=1}^{K} \sum_{j=1}^{m} P(\hat{\psi}_i)P(\delta_i|\hat{\psi}_i, p)P(b_{i,j}|s_{i,j}^*)P(\hat{s}_{i,j}|\hat{\psi}_i, p)} - b
\]

\[
= b - b = 0.
\]
4.5.5.2 Standard Error

Assuming the asymptotic distribution of $\hat{b}$ is Gaussian under SRS, it is defined by the mean $\hat{b}$ and variance $\text{var}(\hat{b})$ parameters and thus the probability of the fusion data pair $P(\hat{b}, p)$ can be estimated using the variance of the probability fusion model. Finding the standard error of the model is equivalent to finding the variance of the probability fusion model and is given by

$$\text{var}(\hat{b}) = \text{var} \left[ \frac{\sum_{i=1}^{K} \sum_{j=1}^{m} P(\hat{\Psi}_i)P(\delta_i|\hat{\Psi}_i, p) P(b_{i,j}|s_{i,j}^*) P(s_{i,j}|\hat{\Psi}_i, p) b_{i,j}}{\sum_{i=1}^{K} \sum_{j=1}^{m} P(\hat{\Psi}_i)P(\delta_i|\hat{\Psi}_i, p) P(b_{i,j}|s_{i,j}^*) P(s_{i,j}|\hat{\Psi}_i, p)} \right]$$

$$= \frac{\text{var} \left[ \sum_{i=1}^{K} \sum_{j=1}^{m} P(\hat{\Psi}_i)P(\delta_i|\hat{\Psi}_i, p) P(b_{i,j}|s_{i,j}^*) P(s_{i,j}|\hat{\Psi}_i, p) b_{i,j} \right]}{\left[ \sum_{i=1}^{K} \sum_{j=1}^{m} P(\hat{\Psi}_i)P(\delta_i|\hat{\Psi}_i, p) P(b_{i,j}|s_{i,j}^*) P(s_{i,j}|\hat{\Psi}_i, p) \right]^2}$$

$$= \frac{\sum_{i=1}^{K} \sum_{j=1}^{m} P(\hat{\Psi}_i)^2 P(\delta_i|\hat{\Psi}_i, p)^2 P(b_{i,j}|s_{i,j}^*)^2 P(s_{i,j}|\hat{\Psi}_i, p)^2 \text{var}(b_{i,j})}{\left[ \sum_{i=1}^{K} \sum_{j=1}^{m} P(\hat{\Psi}_i)P(\delta_i|\hat{\Psi}_i, p) P(b_{i,j}|s_{i,j}^*) P(s_{i,j}|\hat{\Psi}_i, p) \right]^2}$$

In practice, the variance of the pixel attribute may not be known. In this case, the variance of the pixel attribute can be estimated by

$$\text{var}(b_{i,j}) = \sum_{i=1}^{K} \sum_{j=1}^{m} (b_{i,j} - \bar{b})(b_{i,j} - \bar{b})^T$$

because $\hat{b}$ is unbiased (Mardia et al. 1980).

4.5.5.3 Relative Efficiency

The Asymptotic Relative Efficiency (ARE) of two unbiased estimators is the ratio of the determinants of their variances (Serfling 1980). If the ARE is greater than one, then the first estimator is more efficient than the second. From (103), the variance of the sample mean $\bar{b}$ is given by
\[
\text{var}(\tilde{b}) = \text{var}\left(\frac{1}{Km} \sum_{i=1}^{K} \sum_{j=1}^{m} b_{i,j}\right) \\
= \frac{1}{K^2m^2} \sum_{i=1}^{K} \sum_{j=1}^{m} \text{var}(b_{i,j}) \\
= \frac{\text{var}(b_{i,j})}{Km}
\]

and the ARE is then

\[
\frac{|\text{var}(\tilde{b})|}{|\text{var}(\hat{b})|} = \frac{\left[\Sigma_{i=1}^{K} \sum_{j=1}^{m} P(\tilde{\psi}_{i}) P(\delta_i|\tilde{\psi}_{i}, \mathbf{p})^2 P(b_{i,j}|\mathbf{s}_{i,j}^*)^2 P(\hat{\delta}_{i,j}|\hat{\psi}_{i}, \mathbf{p})^2 \right]}{\left[\Sigma_{i=1}^{K} \sum_{j=1}^{m} P(\tilde{\psi}_{i}) P(\delta_i|\tilde{\psi}_{i}, \mathbf{p})^2 P(b_{i,j}|\mathbf{s}_{i,j}^*)^2 P(\hat{\delta}_{i,j}|\hat{\psi}_{i}, \mathbf{p})^2 \right]} \\
\geq 1.
\]

From (145), it is clear that the two estimators are equally efficient when (114) is uniform across the marginal uncertainties in the fusion data pair. Hence, efficiency is gained when the marginal uncertainties in the fusion data pair deviate from each other.
4.6 Simulation

The probability fusion model presented in section 4.5 is an estimator of the fusion data pair specified as a linear combination of normalized probability weights on observed pixel attributes in multiple image frames. Each weight is the product of four component probabilities: pose, attribute, transformation and reduction probabilities. The pose and attributes probabilities are directly measured by either parametric or non-parametric methods. These probabilities are not conditional on any additional sources of uncertainty.

Conversely, the transformation and reduction probabilities are conditional on the pose vector and the point cloud. Therefore these probabilities cannot be directly measured without considering the probability distributions on these random variables. The probabilities may be difficult to compute in closed form because the complex mathematical relationship between these variables illustrated in sections 4.5.3 and 4.5.4.

As discussed in sections 4.5.3 and 4.5.4, the transformation and reduction probabilities can be approximated, respectively. A Bernoulli distribution may be suitable for the reduction probability as the occlusion event is binary: the image ray passing through the lidar point is or is not occluded. However an approximation to the transformation probability is more complex because its domain is in the image space taking values in $\mathbb{R}^2$.

In this section, the von Mises-Fisher distribution is proposed as an approximation to the transformation probability. Its density function is defined by mean and concentration parameters. The mean is explicitly defined in section 4.6.1 and a model for the concentration parameter is estimated in 4.6.2.

4.6.1 Approximate Density Function for Fusion Transformation

As identified in section 4.5.3, the transformation probability $P(\hat{s}_{i,j} | \hat{\psi}_i, p)$ is measured by the distribution of $\hat{s}_{i,j}$ on the image plane. This distribution is assumed to have zero mean and finite second-order moment. As discussed in section 4.5.3.1, a condition of the ODC ratio
implies that the transformation probability can be defined solely by the uncertainty in the rotation matrix, ignoring the uncertainty in the camera projective center and the point cloud.

The transformation probability is the probability of observing the vector difference $\hat{s}_{i,j}$ between the $j^{th}$ pixel $s^*_{i,j}$ and prediction point $\hat{s}_i$ in the $i^{th}$ image frame (or one more extreme), defined in (112). Thus, we wish to find a distribution for $\hat{s}_{i,j}$ such that the probability of observing this vector difference is zero can be tested. It is also desirable to simplify the computation of the probability by defining this distribution without having to solve for $\hat{s}_i$ or $\tau$.

Such a simplified distribution can be defined using image rays in the object space as opposed to the image space in which $\hat{s}_{i,j}$ is exists. As demonstrated in Chapter Three, the image ray for $s^*_{i,j}$ is conditional on the pose vector $\hat{\psi}_i$, a quantity that is estimated or measured during image-lidar fusion. Likewise the image ray of $\hat{s}_i$ can be defined by a line independent of $\hat{\psi}_i$ using the lidar point in the fusion data pair as

$$\hat{v}_i = \lambda_i + d(p - \hat{\lambda}_i)$$

(146)

where $p - \hat{\lambda}_i$ is the direction vector of the line and arbitrary $d$ determines the vector position along the line. Using the original definition presented in Chapter Two, the image ray of $s^*_{i,j}$ is

$$v^*_{i,j} = \lambda_i + \tau \hat{R}_i(s^*_{i,j} + \omega)$$

(147)

where $\hat{R}_i(s^*_{i,j} + \omega)$ is the direction vector and the optical depth $\tau$ determines the vector position along the line. Because the magnitudes of the direction vectors are different in (146) and (147), the position vectors along the lines cannot be determined by a common scalar as $d \neq \tau$ necessarily. However a common scalar $\tau$ is found by normalizing the direction vectors of (146) and (147) as
\[ \hat{v}_i = \lambda_i + \frac{\tau(p - \hat{\lambda}_i)}{\|p - \hat{\lambda}_i\|} \]  

and

\[ v_{i,j}^* = \lambda_i + \frac{\tau \hat{R}_i(s_{i,j}^* + \omega)}{\|\hat{R}_i(s_{i,j}^* + \omega)\|} \]  

Respectively so that \( d = \tau \). Likewise \( \hat{v}_i \) and \( v_{i,j}^* \) can be translated to the origin of \( \mathbb{R}^3 \) as

\[ \hat{v}_i = \frac{\tau(p - \hat{\lambda}_i)}{\|p - \hat{\lambda}_i\|} \]  

and

\[ v_{i,j}^* = \frac{\tau \hat{R}_i(s_{i,j}^* + \omega)}{\|\hat{R}_i(s_{i,j}^* + \omega)\|} \]

respectively, without effecting their relative geometry. As illustrated by Figure 38, the vectors \( \hat{v}_i \) and \( v_{i,j}^* \) lay on a sphere of radius \( \tau \) in \( \mathbb{R}^3 \). By approximating \( \hat{s}_{i,j} \) with \( \hat{v}_i \) and \( v_{i,j}^* \), the value of \( \hat{s}_i \) does not need to be computed.
Upon $\tau = 1$ the sphere in $\mathbb{R}^3$ is of unit radius which gives rise to several possible density functions for circular data. Such densities functions included those in the class densities on $\mathbb{R}^3$ wrapped to the sphere and the class of Fisher densities on the sphere (Mardia & Jupp 1999). This later class includes the von Mises-Fisher distribution which is the sphere-analog of the Gaussian distribution and is often used to model circular data on a sphere in $\mathbb{R}^3$ (Fisher 1995).

Formally letting $\tau = 1$ as described above, $\hat{\nu}_i$ and $\nu_{i,j}^*$ then become

$$\hat{\nu}_i = \frac{p - \hat{\lambda}_i}{||p - \hat{\lambda}_i||}$$  \hspace{1cm} (152)

and
respectively. The approximation is then \( \mathbf{v}^*_{i,j} \sim \nu MF(\hat{\mathbf{v}}_i, \kappa) \) where \( \nu MF(\hat{\mathbf{v}}_i, \kappa) \) is the von Mises-Fisher distribution on a sphere in \( \mathbb{R}^3 \) and an approximate density function for fusion transformation to measure the transformation probability \( P(\hat{s}_{i,j}|\hat{\mathbf{v}}_i, \mathbf{p}) \). The density is centered at \( \hat{\mathbf{v}}_i \) with concentration parameter \( \kappa \). As \( \kappa \) increases the probability mass becomes centered at \( \hat{\mathbf{v}}_i \) on the sphere. In this sense, \( \kappa \) is the second moment of the distribution.

Although \( \nu MF(\hat{\mathbf{v}}_i, \kappa) \) may be a reasonable approximation in some cases, the concentration parameter \( \kappa \) is never actually measured or estimated. Ideally, \( \kappa \) should be defined as a function of the variance of trajectory parameters \( \mathbf{\theta}_i \) if the ODC ratio presented in section 4.5.3.1 is sufficiently small. The rotational variance is a function of the variance of trajectory parameters \( \mathbf{\theta}_i \) in the pose vector \( \hat{\mathbf{v}}_i \).

### 4.6.2 Methods

To estimate a model for the concentration parameter \( \kappa \) of the approximate von Mises-Fisher distribution, trajectory parameters \( \mathbf{\theta} \) were randomly generated and transformed to the unit sphere. A normal distribution with mean \( \mathbf{\phi} \) and diagonal covariance \( \Sigma_{\mathbf{\theta}} = c \mathbf{I} \) was used to generate \( \mathbf{\theta} \) where \( c \) is a scalar and \( \mathbf{I} \) is a \( 3 \times 3 \) identity matrix. The value of \( c \) was fixed periodically in the interval \([0.01, 10]\) every one-hundredth of a degree to generate 1,000 values of \( c \).

For each value of \( c \), the covariance matrix \( \Sigma_{\mathbf{\theta}} \) was calculated and 100 samples of \( \mathbf{\theta} \sim N(\mathbf{\phi}, \Sigma_{\mathbf{\theta}}) \) were generated where \( \mathbf{\phi} = (120, 6, 3)^T \) degrees. The choice of \( \mathbf{\phi} \) is arbitrary because the von Mises-Fisher distribution is invariant under rotation (Mardia & Jupp 1999). A rotation matrix \( \mathbf{R}_\mathbf{\theta} \) was formulated for each \( \mathbf{\theta} \). The data were then generated on the unit sphere as
\[ \mathbf{v}_c = \mathbf{R}_\theta (0,0,1)^T \]  

(154)

where \((0,0,1)^T\) is the “up vector” or the north pole. Again, the choice of the up vector is arbitrary because the von Mises-Fisher distribution is invariant under rotation.

For each value of \(c\), the concentration parameter \(\kappa_c\) was estimated as \(\hat{\kappa}_c\) using maximum likelihood where \(\mathbf{v}_c \sim \nu MF(\mathbf{R}_\theta (0,0,1)^T, \kappa_c)\). The data were assembled across the values of \(c\) in a matrix with 1,000 rows. Each row was a vector \((\hat{\kappa}_c, |\mathbf{\Sigma}_\theta|)\) where \(\hat{\kappa}_c\) was considered a response and \(|\mathbf{\Sigma}_\theta|\) was considered a covariate. A log-log plot of the data is presented in Figure 39.
Figure 39: Log-log plot of $(\hat{\kappa}, |\Sigma_\theta|)$.

The plot shows the data are approximately linear in the log domain which implies the following model form for $\kappa$ as a function of $|\Sigma_\theta|$:

$$\kappa = a \ |\Sigma_\theta|^{b}$$  \hspace{1cm} (155)
where $a$ and $b$ are parameters of the model. The parameters were estimated using ordinary least squares by taking the logarithmic transform of (155) as

$$
\ln(\kappa) = \ln(a) + b \ln(|\Sigma_\theta|).
$$

\[156\]

### 4.6.3 Results

The parameterized model (156) infers a strong relationship between the concentration parameter $\kappa$ and $|\Sigma_\theta|$. Parameter estimates $\hat{a}$ and $\hat{b}$ are statistically significant (see Table 9). As illustrated by Figure 40, a quantile plot of the model residuals shows they are normally distributed.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Estimate</th>
<th>Significance (p-value)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\ln(a)$</td>
<td>1.00114</td>
<td>$&lt;0.001$</td>
</tr>
<tr>
<td>$b$</td>
<td>-3.14186</td>
<td>$&lt;0.001$</td>
</tr>
</tbody>
</table>

Table 9: Parameter estimates of the model for the concentration parameter $\kappa$ as a function of $|\Sigma_\theta|$. 
Rerunning the model fitting procedure described in section 4.6.2 gives parameter estimates that fluctuate around $\log(\hat{a}) = 1$ and $\hat{b} = \pi$ which empirically suggests the following model

$$\kappa = e^{\mid\Sigma_{\theta}\mid^{\pi}}$$

(157)
where \( e \approx 2.718 \) and \( \pi \approx 3.142 \).

Based on the significance of parameter estimates in Table 9 and their values relative to \( e \) and \( \pi \), the following model is proposed as an approximation to the transformation probability:

\[
P(\mathbf{s}_{i,j} | \mathbf{\hat{p}}, \mathbf{p}) \approx \frac{e^{\frac{\mathbf{p} \cdot (\mathbf{p} - \lambda_i)}{\|\mathbf{p} - \lambda_i\|}}}{2\pi (e^{\|\mathbf{p}\|^2} - e^{-\|\mathbf{p}\|^2})} \tag{158}
\]

which is the von Mises-Fisher distribution with mean (152) and concentration parameter (157) on the vector (153). Note this approximation is conditional on \( (\mathbf{\hat{p}}, \mathbf{p}) \) as expected and that the imaging point \( \mathbf{s}_i \) of \( \mathbf{p} \) does not need to be computed.
4.7 Discussion

The probability fusion model is nested in a larger, theoretical framework for the fusion of image and lidar data. The model, like its extending framework, surmounts fusion reduction and duplication by mathematically parsing these problems into component probabilities. It also addresses the multiple sources of uncertainty in image-lidar fusion in a probabilistic context. By formulating the model in this context, it is shown to be unbiased and efficient. However working in this context necessitates some complex calculations of probabilities, some of which can be allayed through approximation under certain conditions.

As discussed in Chapter Three and section 4.5.1, the measurement of the pose probability requires some non-parametric methods or some strong parametric assumptions. If estimated using the methods provided in Chapter Three, it may be difficult to identify a reasonable density function for the pose vector. In some other cases, such as when the pose vector is measured using an IMU and GPS, less information about the density function may be available than when estimated by methods in Chapter Three. One may arbitrarily choose the Gaussian density in these cases. Though when the density function is arbitrarily specified, the reasonableness of the selected function should be assessed whenever possible. One area of further research is how to assess parametric assumptions of camera pose. Without such an assessment, it is preferable to use non-parametric methods if possible.

In many instances of image-lidar fusion, the attribute matrix may be considered fixed and non-random. For example, the attribute matrix could be binary observations of object presence in a scene; it is obvious that the object is or is not present and thus the attribute matrix can be considered non-random. However in most instances, the attribute matrix contains some measurement or classification error and thus cannot be considered non-random. As noted in section 4.4.1, the attribute probability specified in section 4.5.2 does not account for the possible autocorrelation of attributes in the image frame (or between image frames). In individual applications of the probability fusion model, the covariance structure of the attribute matrix should be explored and accounted for in the estimate of the fusion data pair.
The reduction probability presented in section 4.5.4 is the product of probabilities over the entire point cloud matrix. In practice, this computation is $O\left(2(N - 1)\right)$ for a fusion data pair where $N$ is the number of points in the cloud. To reduce the size of this computation, the lidar points in the point cloud matrix can be sorted by location relative to each other. Using an index on the matrix defined by this relation, then only those points closest to $p$ in the fusion data pair need to be assessed. This computation is further simplified by taking an approximation to the reduction probability, perhaps as the product of Bernoulli densities as described in section 4.5.4.

The most complex of all component probabilities in the probability fusion model is the transformation probability. As an alternative to resampling methods, the approximation in section 4.6.1 can be used under the condition that the ODC ratio is small. The approximation is based on the von Mises-Fisher distribution and defined by a single concentration parameter. The concentration parameter is estimated by a model as a function of the determinant of the covariance matrix of the pose vector. As described in section 4.6.2, the model is parameterized by scaling the diagonal elements of sample covariance matrices by a common factor. This method generates covariance matrices of distributions which are symmetric about the mean.

In reality, the covariance structure of the pose vector may not be symmetric.

To improve upon the approximation to the transformation probability as specified in section 4.6.3, an alternative distribution could be selected. The Kent distribution is defined by two additional parameters which allow the concentration of the distribution on the sphere to vary about two axes (Mardia & Jupp 1999). This distribution may better approximate the non-symmetry in the distribution of some pose vectors.

The framework of the probability fusion model gives rise to an interesting corollary that warrants further research. The fusion probability in section 4.4.2 shows that the estimated fusion data pair is conditional on the sources of uncertainty. This fact by itself is obvious, but what is more interesting is that the estimator for the fusion data pair is the sum of products of probabilities. Sections 4.5.1, 4.5.2, 4.5.3 and 4.5.4 attempt to measure these probabilities to
arrive at a point estimate of the fusion data pair. However from a Bayesian perspective when the probabilities are defined only by their distributions and not computed, these component probabilities act as prior densities on the fusion data pair. This perspective suggests that the sources of uncertainty are priors on the fusion data pair, and that generally, any estimator can viewed the same way by simply writing out the estimator as conditional on random variables and expressing its conditional probability statement using Bayes rule.

4.7.1 Contribution
This chapter provides the following unique contributions:

- A theoretical framework for the fusion of image and lidar data;
- The probability fusion model for image-lidar fusion;
- The optical depth covariance ratio (ODC) to identify when the transformation probability on the image plane can be measured by rotational variance;
- An approximate density function for the transformation probability; and
- Corollaries that Horvitz-Thompson weights and Inverse Distance Weighting can be viewed in terms of prior probabilities.
Chapter Five: Conclusion

This dissertation is concerned with identifying and estimating the geometric relationships of camera imagery using point clouds, estimating radiometry from point clouds as a basis of similarity and generalizing image-lidar fusion in the context of accuracy and precision. Chapter Two proposes the radiometric image model to compare the point cloud with camera imagery. Chapter Three builds upon the model to solve the 2D perspective point problem using the point cloud in a statistical manor. And finally, Chapter Four explores the sources of uncertainty in image-lidar fusion and proposes the probability fusion model as efficient estimator of fusion data as a specific case in a general probabilistic context of the theory of image-lidar fusion.

Each of the chapters is an application of lidar to photogrammetry, and together, these chapters unify the two data through a common geometric relationship and from a statistical perspective. This common geometry of the two data types is first presented in Chapter Two as the image ray. The image ray can be expressed in a world coordinate system, the camera frame and the image frame which gives the necessary utility to applications of imagery and lidar.

The image ray defines the object to image space transformation in Chapter Three through which the reflectance image model proposed in Chapter One is mathematically connected to the image space. This transformation also provides the basis for estimating camera pose from lidar data. For the fusion of imagery and lidar data, an estimate or measurement of camera pose is necessary.

Using an estimate of camera pose from Chapter Three, the image ray is used to fuse pixel attributes to lidar points through the probability fusion model in Chapter Four. The probability fusion model accounts for the uncertainty in the image ray and the problems of fusion reduction and duplication.

Throughout this dissertation, the multiple applications of imagery and lidar data are defined in a statistical manor. Chapter Two identifies a maximum likelihood estimator for camera pose in
the presence of measurement error of tie points in the image frame. The chapter also explores the parametric assumptions of some other common approaches to estimating camera pose. The theory of image-lidar fusion is developed in Chapter Three entirely from the perspective of multiple sources of uncertainty. By formulating the problem of image-lidar fusion in a statistical manor, important qualities of the estimator are identified, including unbiasedness and efficiency.

The result of Chapter Two is the reflectance image. The utility of the model is demonstrated by automatically identifying tie points between the pixels of an image and points in the cloud using common features in imagery and the reflectance image. The reflectance image provides a basis for comparison of the point cloud with imagery which is further demonstrated in Chapter Three.

Chapter Three provides an estimator for camera pose using tie points identified through the reflectance image. The chapter identifies non-parametric methods to quantify the distribution of camera pose. Although estimates are slightly biased, they converge for data from a study site in Yellowstone National Park. Camera pose is central to image-lidar fusion as described in Chapter Three.

Finally, Chapter Three identifies and quantifies the various sources of uncertainty in image-lidar fusion. The result of this chapter is a theoretical framework for fusion and a specific model for image-lidar fusion. Various considerations are also provided in Chapter Three to guide the simplification and complication of future variations of the probability fusion model.
5.1 Future Works

The various applications provided in the chapters of this dissertation lend further research and new application. Consideration and direction are given in the discussion sections of each chapter. These endeavors include time-integral applications of the reflectance image, further exploration of measurement error in tie points, refinement of the estimator for camera pose, new approximations to the transformation probability in image-lidar fusion and theoretical research on prior distributions to sample-based estimates.

Another area of direct research flows from the progression of the dissertation: from beginning to end, it ultimately gives the fusion data for subsequent inference. If the methods presented in this dissertation are encapsulated in a workflow, then the production of fusion data is likely semi-automatic. Such a system leads to the proliferation of fusion data, the applications of which are yet to be realized.

Future work should be conducted on the fusion system and applications of fusion data. Such applications may include 3D object discrimination and classification, model building and simulation, and 3D change detection.
References


Ansar, Adnan & Daniilidis, Kostas, 2003. Linear Pose Estimation from Points or Lines Linear Pose Estimation from Points or Lines.


Brunelli, R. & Poggio, T., 1999. Template Matching : Matched Spatial Filters and beyond,


Von Gruber, O., 1924. *Einfache und Doppelpunkteinschaltung im Raume*,


164


167


Appendix A: Convergence Paths for Camera Pose Estimation

Figure A.1: Convergence path of lambda in X-Y subspace for n=400: (a) starting point, (b) point of convergence, X denotes true value.
Figure A.2: Convergence path of lambda in X-Y subspace for n=80: (a) starting point, (b) point of convergence, X denotes true value.
Figure A.3: Convergence path of lambda in X-Y subspace for n=40: (a) starting point, (b) point of convergence, X denotes true value.
Figure A.4: Convergence path of lambda in X-Y subspace for n=8: (a) starting point, (b) point of convergence, X denotes true value.
Figure A.5: Convergence path of lambda in X-Z subspace for n=400: (a) starting point, (b) point of convergence, X denotes true value.
Figure A.41: Convergence path of lambda in X-Z subspace for n=80: (a) starting point, (b) point of convergence, X denotes true value.
Figure A.6: Convergence path of lambda in X-Z subspace for n=40: (a) starting point, (b) point of convergence, X denotes true value.
Figure A.7: Convergence path of lambda in X-Z subspace for n=8: (a) starting point, (b) point of convergence, X denotes true value.
Figure A.42: Convergence path of lambda in Y-Z subspace for n=8: (a) starting point, (b) point of convergence, X denotes true value.
Figure A.8: Convergence path of lambda in Y-Z subspace for n=40: (a) starting point, (b) point of convergence, X denotes true value.
Figure A.9: Convergence path of lambda in Y-Z subspace for n=80: (a) starting point, (b) point of convergence, X denotes true value.
Figure A.10: Convergence path of lambda in Y-Z subspace for n=400: (a) starting point, (b) point of convergence, X denotes true value.
Figure A.11: Convergence in lambda by iteration for n=8.
Figure A.12: Convergence in lambda by iteration for n=40.
Figure A.13: Convergence in lambda by iteration for n=80.
Figure A.14: Convergence in lambda by iteration for n=400.
## Appendix B: Key to Variables

<table>
<thead>
<tr>
<th>Notation</th>
<th>Definition</th>
<th>Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>$D$</td>
<td>Direct irradiance</td>
<td>2.2</td>
</tr>
<tr>
<td>$\psi$</td>
<td>Cosine of the angle of incidence or pose vector</td>
<td>2.2</td>
</tr>
<tr>
<td>$I$</td>
<td>Direct normal irradiance</td>
<td>2.2</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>Fractional absorption</td>
<td>2.2</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>Fractional transmission</td>
<td>2.2</td>
</tr>
<tr>
<td>$\rho$</td>
<td>Fractional reflectance</td>
<td>2.2</td>
</tr>
<tr>
<td>$R$</td>
<td>Reflected irradiance</td>
<td>2.2</td>
</tr>
<tr>
<td>$r$</td>
<td>Reflectance model</td>
<td>2.2</td>
</tr>
<tr>
<td>$\phi$</td>
<td>Solar zenith angle</td>
<td>2.2</td>
</tr>
<tr>
<td>$\theta$</td>
<td>Solar azimuth or trajectory parameters</td>
<td>2.2</td>
</tr>
<tr>
<td>$S$</td>
<td>Surface slope angle</td>
<td>2.2</td>
</tr>
<tr>
<td>$A$</td>
<td>Surface slope azimuth</td>
<td>2.2</td>
</tr>
<tr>
<td>$p'$</td>
<td>Arbitrary point of interest</td>
<td>2.2.1.1</td>
</tr>
<tr>
<td>$t$</td>
<td>Time</td>
<td>2.2.2</td>
</tr>
<tr>
<td>$c$</td>
<td>Normalizing constant</td>
<td>2.2.2</td>
</tr>
<tr>
<td>$u$</td>
<td>Solar ray</td>
<td>2.3</td>
</tr>
<tr>
<td>$s_r$</td>
<td>Direction vector</td>
<td>2.3</td>
</tr>
<tr>
<td>$p$</td>
<td>Position vector or lidar point</td>
<td>2.3</td>
</tr>
<tr>
<td>$t$</td>
<td>Arbitrary scalar as defined by the implicit form of a vector equation</td>
<td>2.3</td>
</tr>
<tr>
<td>$\tau$</td>
<td>Horizontal tolerance or optical depth</td>
<td>2.3</td>
</tr>
<tr>
<td>$v$</td>
<td>Point along a line or image ray</td>
<td>2.4</td>
</tr>
<tr>
<td>$\lambda$</td>
<td>Projective center of the camera in object space</td>
<td>2.4</td>
</tr>
<tr>
<td>$f$</td>
<td>Camera focal length</td>
<td>2.4</td>
</tr>
<tr>
<td>$R$</td>
<td>Rotation matrix</td>
<td>2.4</td>
</tr>
</tbody>
</table>
\((i_0,j_0)\) Image principle point 2.4
\(\delta\) Camera lens distortion 2.4
\(\mathbf{R}\) Direction vector 2.4
\(\lambda\) Position vector in the vector equation of the line 2.4
\(d\) Distance 2.4
\(k(d)\) A kernel function 2.5
\(b_{i,j} \in [0,1]\) Value of pixel (i,j) in the reflectance image 2.5
\(\mathcal{P}\) Index set to the point cloud 2.5
\(\mathcal{D}\) Set of all DSM cells 2.6.4
\(i\) Index to a cell 2.6.4
\(\omega\) The fraction of diffuse irradiance to total irradiance reaching the surface 2.6.4
\(h\) Index to a DSM cell 2.7
\(s\) Imaging vector 3.2.1
\(\mathbf{R}^{-1}\) Matrix inverse of the rotation matrix 3.2.1
\(\alpha\) Yaw 3.2.3
\(\beta\) Pitch 3.2.3
\(\gamma\) Roll 3.2.3
\(\mathbf{\theta}\) Flight trajectory vector 3.2.3
\(\mathbf{I}\) Identity matrix 3.2.3
\(w_i\) Weights 3.3
\(s^*\) Observed imaging vector in the image space 3.3
\(\psi\) Weight function (the derivative of some object function) or the pose vector 3.3
\(\mathbf{\theta}\) Weighted mean or flight trajectory parameters 3.3
\(s\) Median absolute deviation 3.3
\(\epsilon_{BV}\) Bivariate difference 3.4.3
\(\epsilon_{VN}\) Vector normed difference 3.4.3.1
\[ \eta \] Linear predictor 3.4.4
\[ \eta_{BV} \] Bivariate mean 3.4.4
\[ \Sigma_{BV} \] Bivariate covariance 3.4.4
\[ B \in \mathbb{R}^{2 \times 2} \] Effects matrix of the covariates DistPPX and DistPPY 3.4.4

\[ m \] Lagrange multipliers 3.5.2.1
\[ \alpha_j \] Rotation scale factor 3.5.2.1
\[ D \] Arbitrary square matrix 3.5.3.1
\[ P \in \mathbb{R}^{3 \times N} \] Point cloud matrix 4.2
\[ B \in \mathbb{R}^{L \times M} \] Pixel attribute matrix 4.2

\[ \lambda^T \] Location of the projective center of the camera in object space 4.2
\[ b \] Solar attribute 4.3.1
\[ v_{\Psi_l} \] The event that the pose vector is selected 4.4.1
\[ b_{i,j} \] The observed attribute of the pixel in the image frame 4.4.1

\[ P(\hat{\Psi}_l) \] Pose probability 4.4.2
\[ P(\delta_{i,l}|\hat{\Psi}_l, p, P) \] Reduction probability 4.4.2
\[ P(b_{i,j}|s_{i,j}) \] Attribute probability 4.4.2
\[ P(\hat{s}_{i,j}|\hat{\Psi}_l, p) \] Transformation probability 4.5