GENERATING THEMATIC MAPS FROM HYPERSPECTRAL IMAGERY USING A BAG-OF-MATERIALS MODEL

DISSERTATION

Presented in Partial Fulfillment of the Requirements for the Degree Doctor of Philosophy in the Graduate School of the Ohio State University

By

Kyoung-Jin Park, B.S., M.S.
Graduate Program in Geodetic Science and Surveying

The Ohio State University
2013

Dissertation Committee:

Alper Yilmaz, Advisor
Carolyn Merry
Alan Saafeld
ABSTRACT

Obtaining information about Earth’s surfaces and generating a land cover map is essential to remote sensing research. For that purpose, identifying and classifying the characteristics of pixels are fundamental problems. To obtain a significant amount of information about the Earth’s surface, hyperspectral images taken from high altitude are used. However, exploiting a hyperspectral image is challenging because the spectral dimension is high and the spatial resolution is low. In order to overcome both the high dimensionality and low spatial resolution problems, I introduce a novel method for robust identification and classification of pixels’ properties based on the Latent Dirichlet Allocation model. The proposed method first analyzes mixture spectral signatures to extract material combinations. These combinations are processed to discover the latent theme for each pixel. This process is governed by a hierarchical Bayesian learning model, whose distribution and parameters are estimated using Gibbs sampling. As a result of parameter estimation, each pixel is described using topics distribution and provides pixel descriptors used for identification and classification tasks. Compared to the original spectral information, these descriptors have significantly reduced dimensionality, yet provided efficient clustering to segment the image. Experimental results show that the proposed method effectively handles identification and classification problems for hyperspectral images.
This is dedicated to my family.
I would like to thank my advisor, Dr. Yilmaz. I have learned a lot from him while I was studying at the Ohio State University. He always encouraged and inspired me with his outstanding guidance. With his unlimited insights, my research experiences have been extended.

I would also like to thank Dr. Carolyn Merry for serving on my dissertation committee. Her remote sensing class gave many insights for further research.

My sincere thanks to Dr. Alan Saalfeld for serving on my dissertation committee, as a reviewer and examiner. His classes offer ideas about computational approaches for geospatial research.

I would also like to extend my sincere appreciation to Dr. Elena Irwin. She gave me an opportunity to participate in the project of generating a land use change simulation model. With her continuous support, I could finish my dissertation research.

I would also like to acknowledge Dr. Anthony Schenk. He introduced me to Digital Photogrammetry and offered me enlarged views of research.

I would like to thank to Darla Peelle who help to proofread. Her warm and nice comments encouraged me also.

I also thank my former and present Photogrammetric Computer Vision Laboratory members: Gabor Barsai, Panu Srestasathierm, Mohammed Al- Shahri, Po-Lun Lai, Sudhagar Nagarajan, Jinwei Jiang, Young-Jin Lee, Daniya Zamalieva, Heewon
Lee and Rhae Sung Kim for discussing their research ideas, as well as for giving comments to improve my ideas. Also, I thank my Korean friends and colleagues at OSU, Jihye Park, Hyojin Yang, Yoonjae Chung, and Yeosang Yoon.

And last, I would like to thank my parents, and parents-in-law who always believe and pray for me in Korea. I would like to thank my son, Young-Joon. He is always a good son to me. My little daughter Ji-Yune makes me happy all the time. They are precious gifts to my life. Last, but not least, I thank my husband, Seongjin, for all the love, support, patience, care and encouragement. Without his sacrifice and support, I could not finish this dissertation work. I love you, all my family.
VITA

1996 ................................. B.S in Computer Science, Sogang University, Seoul, Korea

1998 ................................. M.S in Computer Science, Sogang University, Seoul, Korea

2008 ................................. M.S in Geodetic Science, the Ohio State University

1998-1999 ........................... Software Developer, Hyundai Information Technology Ltd.

1999-2001 ........................... Researcher, National Information Society Agency

2005-2010 ........................... Graduate Research Associate, the Ohio State University

2010-2010 ........................... Intern, ERDAS

2011-present ........................ Graduate Research Associate, the Ohio State University
PUBLICATIONS


FIELDS OF STUDY

Major Field: computer vision

Specialization: machine learning

Specialization: photogrammetry

Specialization: remote sensing
# TABLE OF CONTENTS

Abstract ................................................................. ii  
Dedication ............................................................... ii  
Acknowledgments ..................................................... iv  
Vita ................................................................. vi  
List of Figures ........................................................... ix  

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1.1</td>
<td>4</td>
</tr>
<tr>
<td>1.2</td>
<td>5</td>
</tr>
<tr>
<td>1.2.1</td>
<td>6</td>
</tr>
<tr>
<td>1.2.2</td>
<td>7</td>
</tr>
<tr>
<td>1.2.3</td>
<td>8</td>
</tr>
<tr>
<td>1.3</td>
<td>9</td>
</tr>
<tr>
<td>1.3.1</td>
<td>12</td>
</tr>
<tr>
<td>1.3.2</td>
<td>13</td>
</tr>
<tr>
<td>1.4</td>
<td>14</td>
</tr>
<tr>
<td>2</td>
<td>16</td>
</tr>
<tr>
<td>2.1</td>
<td>17</td>
</tr>
<tr>
<td>2.2</td>
<td>19</td>
</tr>
<tr>
<td>2.2.1</td>
<td>20</td>
</tr>
<tr>
<td>2.2.2</td>
<td>23</td>
</tr>
<tr>
<td>2.3</td>
<td>24</td>
</tr>
<tr>
<td>2.3.1</td>
<td>25</td>
</tr>
<tr>
<td>2.3.2</td>
<td>26</td>
</tr>
<tr>
<td>2.3.3</td>
<td>30</td>
</tr>
<tr>
<td>2.4</td>
<td>35</td>
</tr>
<tr>
<td>2.4.1</td>
<td>37</td>
</tr>
</tbody>
</table>
Bag-of-Materials Model .................................................. 36

3.1 Data Preparation .................................................... 37
  3.1.1 Notations ....................................................... 38
  3.1.2 Spectral Unmixing ............................................. 40

3.2 Statistical Model for Bag-of-Materials .......................... 41
  3.2.1 Generative Model for Hyperspectral Images Analysis ...... 42
  3.2.2 Probability Density Functions for Hyperspectral Image ... 47

3.3 Estimation of Model Parameters ................................. 48
  3.3.1 Gibbs sampling for Probability Estimation ................. 49

4 Determination of Variables ........................................ 53

4.1 Determination of Hyperparameters ............................... 54
  4.1.1 Binomial Distribution and Multinomial Distribution ...... 54
  4.1.2 Beta Distribution and Dirichlet Distribution .............. 55
  4.1.3 Selection of Hyperparameter ................................. 59

4.2 Determination of the Number of Topics ....................... 60
  4.2.1 Perplexity ..................................................... 60
  4.2.2 Perplexity of the Bag-of-Materials Model ................. 62
  4.2.3 Estimated Model Comparison ................................. 63

5 Experiment and Results ............................................ 64

5.1 Hyperspectral Data ............................................... 64
5.2 Spectral Unmixing ............................................... 66
  5.2.1 Spectral Library ............................................. 67
  5.2.2 Non-negative Least Square Estimation ...................... 67
  5.2.3 Abundance Map ............................................... 70

5.3 Model Variables and Determinations of Values ................ 71
  5.3.1 Dirichlet Parameters ...................................... 71
  5.3.2 Topic Dimension ............................................. 78

5.4 Inference of Latent Topics for Each Pixel .................... 79

6 Land Cover Map Generation using Unsupervised Classification .. 97

6.1 Three types of Pixel Descriptors ............................... 98
6.2 Unsupervised Clustering ......................................... 98
  6.2.1 K-means clustering ......................................... 100
  6.2.2 Meanshift Clustering ....................................... 104

6.3 Quantitative Measurement ....................................... 107
  6.3.1 Cluster Cohesion Coefficient .............................. 107
  6.3.2 Kappa Coefficient ......................................... 111
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>FIGURE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Hyperspectral Imaging. Hyperspectral sensor has hundreds of spectral channels that collect radiance reflectance in general. Therefore, the acquired images are called <em>hypercubes</em>. Spectral reflectance of each pixel plots a signature along with the wavelength, and it is represented with <em>l</em>-dimensional vectors, which have the same number of channels. [10]</td>
<td>3</td>
</tr>
<tr>
<td>1.2 Flow of proposed algorithm. Let there be three regions in which I perform the analysis as shown in leftmost hypercube. These regions provide a range of hyperspectral signatures. In this example, Region 1 is selected from an area covered with short vegetation; Region 2 contains a building’s rooftop; and Region 3 has tree coverage. The approach identifies a mixture of materials from observed spectral signatures, as shown in the middle plot. The abundances are then used to discover latent topics distribution. Finally, the topic distributions and the abundances provide inferred topics of the selected regions. The topic distributions are plotted as functions of endmembers.</td>
<td>11</td>
</tr>
<tr>
<td>2.1 Latent Semantic Analysis</td>
<td>24</td>
</tr>
<tr>
<td>2.2 Example of graphical model</td>
<td>29</td>
</tr>
<tr>
<td>2.3 Probabilistic Latent Semantic Analysis Model</td>
<td>31</td>
</tr>
<tr>
<td>2.4 Graphical representation of the LDA model. Observed variable <em>w</em> is denoted by a shaded circle, and latent variables <em>θ</em> and <em>φ</em> are denoted by white circles. Each rectangle, which is called a plate, indicates a repetitive process. <em>α</em> and <em>β</em> are corpus level hyperparameters for the Dirichlet distributions. In this model <em>α</em> provides the condition for generating <em>M</em> multinomial distributions, <em>θ</em>, and <em>β</em> provides the condition for generating words distribution.</td>
<td>34</td>
</tr>
<tr>
<td>Section</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>3.1</td>
<td>The generative model and discriminative model. The generative model examines the possible causes for producing output data. On the other hand, the discriminative model indicates a possible class label of a given output data.</td>
</tr>
<tr>
<td>3.2</td>
<td>Graphical representation of the bag-of-materials model. From the observed spectral reflectance of each pixel, composing material, or visual word, is denoted as variable ( w ) and depicted by a shaded circle in order to indicate an observed variable. Latent variables ( \theta ) and ( \phi ) are denoted by white circles, which indicate non-observed variables. Each rectangle, which is called a plate, indicates a repetitive process. The graphical model represents that a pixel contains ( N ) types of materials and an image is composed of ( M ) pixels. ( K ) indicates the number of the topics used in the model. ( \alpha ) and ( \beta ) are image level hyperparameters for the Dirichlet distributions. In this model, ( \alpha ) provides the condition for generating ( M ) multinomial distributions, ( \theta_i ), and ( \beta ) provides the condition for generating ( K ) multinomial distributions, ( \phi_j ).</td>
</tr>
<tr>
<td>4.1</td>
<td>The probability density of the Beta distribution has various shapes, depending on the ( \alpha = \alpha_1 ) and ( \beta = \alpha_2 ).</td>
</tr>
<tr>
<td>5.1</td>
<td>Examples of spectral reflectances of selected regions. Three red rectangle regions have ( 10 \times 10 ) pixels, and each region represents ground area, rooftop area, and vegetation area, respectively.</td>
</tr>
<tr>
<td>5.2</td>
<td>Spectral signatures of 49 endmembers in a spectral library provided publicly by the U.S. Army Geospatial Center.</td>
</tr>
<tr>
<td>5.3</td>
<td>Abundance maps according to material</td>
</tr>
<tr>
<td>5.4</td>
<td>Abundance maps according to material (Cont.)</td>
</tr>
<tr>
<td>5.5</td>
<td>Abundance maps according to material (Cont.)</td>
</tr>
<tr>
<td>5.6</td>
<td>Multinomial distribution according to the value of the hyperparameter, ( \alpha ). (a) ( \alpha = 0.01 ), (b) ( \alpha = 0.1 ), (c) ( \alpha = 0.5 ), (d) ( \alpha = 1 ), (e) ( \alpha = 5 ), (f) ( \alpha = 10 ).</td>
</tr>
<tr>
<td>5.7</td>
<td>Multinomial distribution according to the value of the hyperparameter, ( \beta ). (a) ( \beta = 0.01 ), (b) ( \beta = 0.1 ), (c) ( \beta = 0.5 ), (d) ( \beta = 1 ), (e) ( \beta = 5 ), (f) ( \beta = 10 ).</td>
</tr>
<tr>
<td>5.8</td>
<td>Perplexity (solid line) and the perplexity ratio (dotted line) plotted as functions of topic count (( K )). Latent parameters, ( \phi ) and ( \theta ), are generated based on the smallest topic count (( K = 6 ), in this example) which is higher than the 95% line (denoted by the thin, dotted line).</td>
</tr>
</tbody>
</table>
5.9 Topic distribution of materials in the spectral library. Each column in the latent parameter matrix, $\Phi$, provides the allocation of topics for each respective material. Compared to Figure ??, this plot provides information about which materials can be grouped into similar categories. For instance, copper and rubber are different materials and they have different spectral signatures. These materials, however, can be grouped into the same category because they are both used to build roofs.

5.10 Each layer is reconstructed from column vectors in a matrix of topic-document distribution, $\Theta$. The model parameter $\Theta$ is estimated from $\theta \sim \text{Dir}(0.5)$. (a) 1st topic layer, (b) 2nd topic layer, (c) 3rd topic layer, (d) 4th topic layer, (e) 5th topic layer, (f) 6th topic layer.

5.11 Comparison of the first topic distribution according to various $\alpha$ when $\beta$ is 0.1.

5.12 Comparison of the second topic distribution according to various $\alpha$ when $\beta$ is 0.1.

5.13 Comparison of the third topic distribution according to various $\alpha$ when $\beta$ is 0.1.

5.14 Comparison of the fourth topic distribution according to various $\alpha$ when $\beta$ is 0.1.

5.15 Comparison of the fifth topic distribution according to various $\alpha$ when $\beta$ is 0.1.

5.16 Comparison of the sixth topic distribution according to various $\alpha$ when $\beta$ is 0.1.

5.17 Comparison of the first topic distribution according to various $\beta$ when $\alpha$ is 0.5.

5.18 Comparison of the second topic distribution according to various $\beta$ when $\alpha$ is 0.5.

5.19 Comparison of the third topic distribution according to various $\beta$ when $\alpha$ is 0.5.

5.20 Comparison of the fourth topic distribution according to various $\beta$ when $\alpha$ is 0.5.

5.21 Comparison of the fifth topic distribution according to various $\beta$ when $\alpha$ is 0.5.
5.22 Comparison of the sixth topic distribution according to various $\beta$ when $\alpha$ is 0.5. ................................................................. 96

6.1 Three types of pixel descriptors ........................................... 99

6.2 Clustering the study site hyperspectral image using the $K$-means approach and using three different descriptors for $K = 6$ clusters. Clusters for (a) topic distribution-based descriptor; (b) abundance-based descriptor; and, (c) spectral reflectance-based descriptor. The clusters are ordered manually based on appearance similarity. .................... 102

6.3 Clustering Results ............................................................... 103

6.4 Meanshift clustering results of topic distribution ....................... 106

6.5 Comparison of Cluster Cohesion Coefficients for three different descriptors. The black, gray and white bars, respectively, indicate the CCC values computed from the topic distribution-based descriptor (Figure ??a), the abundance-based descriptor (Figure ??b) and spectral reflectance-based descriptor (Figure ??c). ......................... 109

6.6 Comparison of Cluster Cohesion Coefficients for K-means and Meanshift using topic descriptor. ........................................ 110

6.7 Comparison of clustering results with color masked ground truth. (a) masked regions that introduced as a ground truth abundance map in [103]. (b) clustering results of the topic distribution-based descriptor. (c) clustering results of the abundance-based descriptor. (d) clustering results of the spectral reflectance-based descriptor. ....... 113

6.8 Comparison of clustering results with color masked ground truth. (a) masked regions that introduced as a ground truth abundance map in [103]. (b) clustering results of the topic distribution-based descriptor. (c) clustering results of the abundance-based descriptor. (d) clustering results of the spectral reflectance-based descriptor. .................. 114
CHAPTER 1
INTRODUCTION

During the past several decades, imaging science has received a great deal of attention in many fields, because of its capability of capturing and gathering scene information [61] [40]. The evolution of imaging technology not only has developed various image acquisition methods with diverse sensor types, but also has minimized the limitations of temporal and spatial accessibility for acquiring an image. Consequently image types and formats vary widely from close-range images to remotely sensed images, from panchromatic images to hyperspectral images, or from a still image to a dynamic video [9] [97] [12].

Among the many imaging science applications, remote sensing is one of the major fields benefiting from the use of advanced imaging technologies, since the main objective of remote sensing is to obtain information about the Earth’s surface as well as its atmosphere without having physical contact with objects. In order to achieve that goal, various types of remote sensors are used [63].

As an active remote sensor, light detection and ranging (LiDAR) and radar gather information about the Earth and its atmosphere using a scanning method [17]. Structural information about a target object [35] or the speed of a moving object is obtained when time series images are available. For example, digital elevation models (DEM) are generated using interferometric synthetic aperture radar (SAR) [26] or LiDAR [119], and large-scale meteorological data is monitored using radar.
On the other hand, passive sensors collect spectral reflectance information about object surfaces based on electromagnetic radiation [17]. The ranges of reflected and emitted radiation wavelengths that are collected varies from $\gamma$ ray, x-ray, ultraviolet, visible, infrared, microwave, to radio wave. Measuring and quantifying the reflectance characteristics of object surfaces or materials result in various types of images according to the wavelength range of a spectral channel and the number of channels used [110]. For example, a panchromatic image is produced using one spectral channel in visible wavelength, and the result of imaging is a grey-scaled image, where the value of each pixel shows a quantified reflectance of a corresponding object surface. A red-green-blue-based (RGB) color image is an example of using more than one spectral channel. A color image consists of three channels that are within visible wavelength ranges, 620-750nm, 495-590nm, and 450-495nm, respectively. The composite of the spectral reflectance of the three channels is shown as the color of the object surface. Similarly, a multispectral image increases the number of channels and expands the spectral wavelength into the infrared range, thus it provides more spectral information about an object surface than an RGB color image. Furthermore, a hyperspectral image uses many dense spectral channels by placing channels with a narrow wavelength gap. Therefore, a hyperspectral image is obtained from hundreds of spectral channels and provides a great deal of spectral information about the object surface. Thus, images that are acquired from passive sensors are valuable to understand Earth surface types and for remote sensing research, since the observed spectral reflectances, depending on the wavelength ranges of the channels used, can provide information about object surface properties. [63].
Figure 1.1: Hyperspectral Imaging. Hyperspectral sensor has hundreds of spectral channels that collect radiance reflectance in general. Therefore, the acquired images are called hypercubes. Spectral reflectance of each pixel plots a signature along with the wavelength, and it is represented with $l$-dimensional vectors, which have the same number of channels. [10]
1.1 Hyperspectral Images in Remote Sensing

Discovering land cover and extracting valuable information from remotely sensed images is an important process to understand Earth as a system and to manage its resources in the fields of agriculture, hydrology and ecology. Among the various remotely sensed images, hyperspectral images have recently been mainly used than other types images when generating land cover maps for analyzing the Earth’s surface properties [29].

As shown in Figure 1.1, a hyperspectral sensor collects the spectral reflectance of various object surfaces with many spectral channels, whose wavelength ranges from $350\, \text{nm}$ to $2500\, \text{nm}$ and whose interval is as narrow as 5-10 $\text{nm}$. Therefore, a hyperspectral image generally has more than a hundred spectral imaging layers and is also called a hypercube. Due to many and dense spectral channels, hyperspectral imaging provides a wealth of spectral information represented as a high dimensional vector. As a result of enhanced spectroscopy information, hyperspectral images outperform to identify materials or to detect target objects that look similar in visible wavelength ranges, such as RGB channeled images [63].

Because a hyperspectral image provides enriched spectral reflectance information about an object surface, it brings the benefit of identifying Earth’s surface materials such as water, soil, vegetation, metal or an urban structural object. Because of its ability to detect a target object in terms of spectroscopy, hyperspectral images are applied in various analysis studies. Mineralogy uses hyperspectral images to analyze the mineral composition of the rocks and soil of Earth’s surface [105] [23]. In agricultural applications, hyperspectral images are used for vegetation classification [90], estimating crop residues [6], and monitoring the health of crops and detecting the spread of some diseases [121]. Also, environmental research uses hyperspectral images to track oil spills in an affected area [109], or to monitor water quality [136].
Hyperspectral images have not only been applied in remote sensing research, but also have extended into other research fields, such as medicine and food science. Examples include discriminating between healthy food and defective food [46], [2], and distinguishing normal from abnormal skin types [33].

Those interesting regions are extracted through a spectral classification process of identified materials, then a thematic map is generated that implies meaningful information. Each pixel can be identified its corresponding object or types of material, and a class label can be assigned after identification. Classification groups those class labels and that result will generate segmented regions. So, the classification result of a remote sensing image has efficiently replaced a traditional thematic map, or a land cover map generation method that is based on a person surveying a field [130].

1.2 Problem Statement

In order to generate segmented regions, many image classification algorithms have been proposed by computer vision research for decades. In general, classification algorithms have been developed with supervised or unsupervised learning methods. Supervised classification methods assign a label of class for a pixel, after identifying the pixel’s properties. So, guidance such as a truth data set or well trained data set, paired with observed features and its corresponding label, should be provided to identify a pixel’s class. On the other hand, clustering algorithms, called unsupervised classification, make a group or cluster of pixels having similar features instead of assigning them a specific label. So, unsupervised algorithms do not need to know what class the cluster of pixels is in.

In this section, I briefly introduce several algorithms published as supervised and unsupervised classifications. I then state their advantages and defects, and address
the algorithms’ potential problems and limitations when they are applied to hyperspectral images.

1.2.1 Availability of Ground Truth

The main objective of supervised classification is to categorize observed data and to provide a proper label to represent that observation. In the case of image classification, identifying a pixel’s properties is an important phase to assign a relevant class label. In order to assign a proper class label to a pixel, many supervised classifications are based on the discriminative model which utilizes a ground truth data set or training data set. In other words, supervised classification for a discriminative model requires a set of pairs for features and their corresponding class label. By comparing features between an observed data set and a given training data set, the label of the closest feature in the training data set is allocated. Thus, the accuracy of the classification result depends on how a well-trained data set is given. If the size of a well-trained data set or ground data set is not big enough for comparing features and finding similarity, then the classification accuracy would decrease.

Until recently, the remote sensing field has actively employed machine learning algorithms for pattern classification and adapted discriminative models as a supervised classification, such as a Logistics regression model [75] [21] [3], Support Vector Machine(SVM) [7] [79] [81], [92], or Linear Discriminant Analysis (LDA) [5] [98] [100]. However, little prior knowledge of that part of the Earth’s surface is known, if the area photographed is unexplored. That means few ground truth data sets are provided and there is no well-trained data set in the remote sensing study. Consequently, a sparsely-trained data set may induce a low accuracy of classification. In short, applying a supervised classification method for a hyperspectral image in a remote
sensing field requires previous knowledge about the region; otherwise the accuracy of its classification would be poor.

On the other hand, an unsupervised classification, called image clustering algorithms, does not need to assign a label for each pixel as a result of pixel identification and does not need a trained data set. Instead of assigning a specific label to a pixel, clustering methods group pixels that have similar properties [22] [139]. So, clustering methods require us to set some conditions, such as the number of clusters or threshold value for a similarity measurement, as a prior condition. Determining the number of clusters without previous knowledge about the region is not a trivial problem, because selecting the proper criteria affects the quality of clustering results. Nonetheless, most of the methods choose those conditions with heuristics.

1.2.2 Identification of Pixel Properties

Another problem is identifying the pixel properties of a remotely sensed image. Remotely sensed images are generally taken at very high altitudes, ground surface covered within a pixel may range from a few square meters to a few square kilometers. Therefore, multiple types of Earth’s surface objects are included within each pixel [114]. For that reason, a pixel tends to contain more than one pure object of the ground surface imaged. So, when viewing a hyperspectral image, the observed spectral reflectance of a pixel in that image is the result of the mixed spectral signatures of contributing objects or materials.

Another challenge is identifying a pixel for mixed spectral signatures of materials in remotely sensed image classification. That is because the combination of these materials in the spectral signature is subject to variance based on their combined weight.

In order to solved the problem of classifying mixed spectra, spectral unmixing
algorithms have been suggested [66] [10]. Spectral unmixing algorithms focus on identifying contributing materials referred to as endmembers in pixels, as well as their proportions referred to as abundances. Unmixing methods typically assume that the spectral reflectance of a pixel is composed of a linear combination of endmembers [67]. One way of estimating the abundances is the least squares adjustment using a given spectral library of end members [112]. Alternatively, if the spectra of endmembers are unknown due to the lack of a spectral library, pixel purity index (PPI) [14] and N-FINDR [134] methods can be used to identify both the endmembers and their abundances. The estimated abundances and respective endmembers are later processed for classification to generate a thematically segmented map.

1.2.3 High Spectral Resolution

There is, however, a significant limitation in applying traditional classification algorithms to a hyperspectral image because it has a high spectral resolution. Although high spectral resolution provides advantages in providing spectral information, there is also a disadvantage. Consider that a hyperspectral sensor uses many bands to collect spectral reflectances for each pixel and it creates a significant number of imaging layers that form the spectral-cube as shown in Figure 1.2. While the increased spectral dimensionality of a pixel is advantageous in containing distinctive spectral information, it also brings a complexity to manage high dimensionality [41] [132]. Therefore, clustering for a hyperspectral image is challenging in addition to the general considerations of a clustering algorithm.

For example, a pixel in an RGB channel based image is represented with three-dimensional vectors, while that of a hyperspectral image is represented with an \( l \)-dimensional vector, where \( l \) is the same as the number of spectral channels. Due to high spectral resolution, image clustering of a hyperspectral image has suffered from
the Hough phenomenon, also called the *curse of dimensionality* [57]. More formally, if the dataset is not big enough and distinctive enough, high dimensional data causes data sparseness for the clustering process.

In order to resolve this data sparseness problem, researchers have proposed various dimensionality reduction methods that ignore redundant or less informative dimensions while retaining the most informative dimensions. For example, Principle Component Analysis (PCA) extracts predominant components in vectors by linearly projecting the data to a new space based on the eigenspace decomposition [45]. In contrast, nonlinear dimensionality reduction algorithms, such as Local Linear Embedding (LLE) [8] or ISOMAP [122], analyze the nonlinear structure of the data and perform local projections to reduce the vector dimensionality. Studies of dimensionality reduction for hyperspectral image classification were initially introduced in [51], and [137]. Harsanyi and Chang used selected eigenvectors generated from the data to reduce dimensionality [51]; while Ying *et al.* reduced the dimensionality by selecting informative bands [137]. These methods, however, are applied directly to the observed spectral reflectances, and there is little consideration given to the possibility that the spectral values can be generated from multiple materials. In other words, they assume that the pixel contains a single material. This assumption, however, is not realistic in remotely sensed images.

### 1.3 Proposed Method

The main objective of image classification of a hyperspectral image is to create a land cover map to extract and present surface information. While many methods have been proposed for classification problems of a hyperspectral image, most algorithms tackle either the high dimensional vector problem or pixel identification problem but not both.
The goal of this study is to suggest an effective classification method for a hyperspectral image. In order to overcome the aforementioned limitations, lack of a previous data set for study sites, high dimensional spectral problem for finding similarity, and the low spatial resolution problem for identification of mixed spectral reflectance, I propose a generative learning model that does not require prior knowledge about a data set and its class label. The proposed model is called *bag-of-materials model*, where each pixel serves as a bag containing meaningful multiple materials and thus the topics of a bag are inferred by considering those contributing materials. Consequently, each pixel, or a bag, has a novel representative form called the *topics* distribution of a pixel. The representation of a pixel has lower dimensionality than the spectral reflectance channels and the topics distributions are inferred from the abundances of the endmembers.

The diagram in Figure 1.2 shows the information flow in the proposed approach. In order to effectively manage the aforementioned problems, the main goal of this model is to discover latent semantic topics for a pixel that represent the characteristics of that pixel by considering the included objects or materials. The proposed approach first identifies combined materials and their proportional weights called endmembers and abundances, respectively. The abundances of combined endmembers in a pixel are later used to infer the latent semantic topics. The proposed topic inference model employs a generative graphical model utilizing the Latent Dirichlet Allocation (LDA) and is depicted with a Bayesian network to express relationships between the latent topics and the observed hyperspectral reflectances. These latent topics distribution of endmembers within a pixel is used as a representative descriptor. The descriptors of pixels reduce the dimensionality of the spectral observations and are used to generate a land cover map.
Figure 1.2: Flow of proposed algorithm. Let there be three regions in which I perform the analysis as shown in leftmost hypercube. These regions provide a range of hyperspectral signatures. In this example, Region 1 is selected from an area covered with short vegetation; Region 2 contains a building’s rooftop; and Region 3 has tree coverage. The approach identifies a mixture of materials from observed spectral signatures, as shown in the middle plot. The abundances are then used to discover latent topics distribution. Finally, the topic distributions and the abundances provide inferred topics of the selected regions. The topic distributions are plotted as functions of endmembers.
1.3.1 Contribution

Unlike many traditional remote sensing image classification methods based on discriminative learning models, which require a well-trained data set as prior knowledge, this dissertation proposes a novel method for unsupervised classification of a pixel through a generative model. Therefore, the proposed method is advantageous for remote sensing applications, such as generating an Earth surface thematic map if there are few available ground truth data sets of the regions and environment.

Another contribution of the proposed method is to provide semantic meaning when it reduces the dimensionality for a pixel description vector. While most traditional methods of dimensionality reduction are based on Eigenvector transform, they do not preserve the semantics of observed objects. Eigenvector transformation only preserves the important features of spectral channels in a projected space, thus there is little meaning about the object itself. On the other hand, the proposed method offers semantic properties of the objects that appear in a pixel. In other words, the latent topics distribution of combined objects are inferred, thus the topics distribution of a pixel abstracts and summarizes the semantic meanings of contributing objects. So, the class allocation of a pixel is not considered from the direct comparison between ground truth and observed spectral reflectance information, but it is considered from the major topics of a pixel.

The number of topics for describing a pixel is much smaller than the number of spectral channels or the number of materials, thus the proposed method shows great dimensionality reduction effects for a pixel representation.

In short, this proposed method will make contributions to remote sensing image classification as follows:

- Proposed method is a generative unsupervised classifier so it does not depend on the previous knowledge.
• Inferred topics distribution also provides an abstracted semantic meaning for the combined objects, thus it generates an Earth surface thematic map effectively and efficiently.

• Discovered topics distribution describes a pixel with the dimensionally reduced form.

1.3.2 Assumptions and Limitations

The proposed bag-of-materials model presents a pixel having mixture objects by inferred latent topics that explains the characteristics of that pixel. In order to infer the latent topics of a pixel, this model requires the types of objects or materials that belong in a pixel and how much they contribute to composing a pixel component. Those endmembers and abundances are provided by spectral unmixing.

In order to identify combining materials from the mixture spectral signature, this model makes the following assumptions. First, a ground area corresponding to a pixel contains multiple objects. Second, the mixture spectral reflectance of a pixel is linearly combined from the contributing materials' spectral reflectances. The least squares estimation method is applied for spectral unmixing; otherwise another solution for the non-linearity equation should be used for spectral unmixing. Third, those mixed materials have known spectral signatures called a spectral library. If there is no known spectral library, other types of spectral unmixing, such as N-FINDR or PPI should be used.

Also, this model uses the fact that spectral reflectance signatures from a hyperspectral sensor can discriminate an object surface material using detailed spectral information. Thus this model does not work in an image using low spectral channels, since it does not discriminate materials as an endmember and its abundances.
1.4 Organization

The rest of this dissertation is organized as follows:

Chapter 2 introduces latent topic inference models, which are widely applied in natural language processing to understand the content of a document. Those models are explained in two approaches: first, the matrix decomposition approach based on eigenvector is introduced for extracting intrinsic features; second, a statistical Bayesian inference method is expressed for latent theme inference.

Chapter 3 introduces a proposed method, the bag-of-materials model. As a statistical generative model, relevant terminologies and the probability density function of the proposed model are explained. The model parameters estimation is addressed. This chapter shows how the latent topics of a pixel are inferred using Bayesian inference.

Chapter 4 discusses the selection of the values for setting the proposed model. First, the hyperparameter of the Dirichlet distribution is explained. Proper selection of the hyperparameter for the Dirichlet distribution is important, since that value controls the shape of the Dirichlet distribution and then assigns a multinomial distribution. Second, selecting the number of topics of the model is discussed. In order to evaluate the suitability of the model, the perplexity is considered as a means of measuring uncertainty.

Chapter 5 describes the detailed experiment for the suggested bag-of-materials model. As a result, the proposed model is visualized through the topics distribution per pixel.

Chapter 6 shows the clustering results. By comparing the clustering results with traditional clustering algorithms, it shows the clustering result using the proposed method outperforms the clustering result using spectral reflectance. For a quantitative measurement, the clustering coefficient is also introduced.
Chapter 7 summarizes the advantages, disadvantages and limitations of the proposed method. Recommendations are made to overcome the limitations of the proposed method.
Understanding given data and discovering its main characteristics is important in order to classify the data set. Through the investigation of observed data, common patterns or features can be discovered and then a representative descriptor can be assigned as a label. However, discovering the main characteristics can be challenging when an observation is composed of many elements, and its characteristics may vary according to the way those elements are combined.

For example, no documents are exactly alike, but many documents may deliver a similar message or theme. Requests are often made to retrieve similar or relevant documents, so understanding the content of those documents and classifying them is necessary in the field of natural language processing. To understand a document’s content and thereby to find a relevant document requires the ability to abstract and summarize the document’s content using key words instead of memorizing all the words that we read [49]. The abstraction process is performed by analyzing co-occurred words and inferring the latent meaning of them. Therefore, finding latent topics from the observed document provides abstracted semantics about the document and helps one find similar documents with briefly described words. In other words, a topic of a document that has many words is described in high-dimensional data space with a low dimensional subspace [28] [55] [13].

This chapter introduces latent topic inference models, which are well known in
the field of natural language processing. For a computational analysis, the bag-of-words model will be addressed as a simplified document form to represent in a vector space; two approaches are introduced. The first approach is based on a matrix factorization method based on Singular Value Decomposition. The other approach is based on statistical inference using a Bayesian framework. The background idea about discovering topics in a document also would provide insight into automated image understanding in computer vision fields.

2.1 Bag-of-Words Representation

A document carries a message or an idea through sets of words, or sentences, where a word is a basic unit for delivering meaning. Although a document’s length and word order differ and it may not contain the same words as other documents some other documents may deliver similar messages. That is to say the document’s message is understood by the frequency of the words used versus their order.

A bag-of-words model is one of the simplified models that represent a document by considering only the words that appear and their frequency. Assuming that the order of words can be ignored for delivering semantics, each document can be denoted by the histogram of its words and represented in a vector space. To illustrate, if a dictionary having $V$ vocabularies is given, then a document is described with a $V$-dimensional vector. When the $i^{th}$ word of a dictionary, $w_i$, appears $n$ times in a document, $\mathbf{d}$, then the $i^{th}$ element of a document vector has a frequency, $f(w_i)$, as the value.

$$\mathbf{d} = (f(w_1), f(w_2), ..., f(w_V)),$$

where $f(w_i) = \begin{cases} 0 & \text{if } w_i \text{ is not observed} \\ n & \text{if } w_i \text{ is observed } n \text{ times} \end{cases}$ (2.1.1)

The notation 2.1.1 is called a document-term frequency matrix and simply describes a
document. Also, from the notation, co-occurred words are assumed to be semantically related when $f(w_i)$ and $f(w_j)$ are positive, $f(w_i) > 0$ and $f(w_j) > 0$.

Based on the vector form representation shown in (2.1.1), a set of documents, called a corpus generates a matrix by stacking each document vector in a row. If there are $M$ documents in a corpus, all the documents in that corpus are represented by a $M \times V$ document-term frequency matrix, where $V$ is the number of vocabularies in a dictionary. So, the earlier bag-of-words model represents a document in a vector space with the histogram of words, and is regarded as a fundamental representation for bag-of-words models. Although a document-term frequency matrix removes the complexity of word order by ignoring grammar, it still requires a huge dimensional vector space that is the same size as a dictionary in order to represent documents. However, many vocabulary elements in a document-term matrix are marked as zero in that representation scheme, since not all the vocabularies appear all the time.

In order to overcome the high dimensionality problem in an early bag-of-words model, dimensionality reduction algorithms for bag-of-words models is suggested. Based on the matrix decomposition approach, Latent Semantic Indexing (LSI) is introduced [28]. Later, statistical inference models also propose handling multiple topics in a document and this is improved by the Probabilistic Latent Semantic Analysis (pLSA) [55]. Also, the Latent Dirichlet Allocation (LDA) model suggests managing the unseen data effectively by introducing the Dirichlet distribution as a prior condition of document analysis. [13]. The following two sections will show the fundamental background about mathematical and statistical approaches for those models.
2.2 Matrix Decomposition Approach

In section 2.1, the document-term frequency matrix is introduced as a simple notation for documents and represented in a tabular form describing co-occurrences. This section introduces a matrix decomposition method for discovering latent factors, which are regarded as principal components of the given document-term frequency matrix.

In general, matrix decomposition, as an inverse process of matrix multiplication, is used for reducing computational complexity by deriving triangular or diagonal matrices. Matrix decomposition methods are grouped into the following: one for solving the linear equation system and the other for finding eigenvectors. LU decomposition, QR decomposition, or Cholesky decomposition methods are used to reduce the computational complexity for solving the least squares estimation, matrix inversion, or determinant. A proper decomposition method is selected according to the form of a given matrix. For example, the LU decomposition is applied to a square matrix decomposition with L as a lower triangular matrix and U as a upper triangular matrix. Also, the QR decomposition works for the non-squared matrix, m × n matrix with Q, an orthogonal matrix, and R, an upper triangular matrix [30] [82].

On the other hand, matrix factorization methods using eigen-decomposition are used to reveal the intrinsic and inherent characteristics of the observed matrix by transforming into a lower-ranked canonical matrix form. In other words, eigenvector decomposition shows the most principal components of a matrix by transforming the given matrix into the eigenspace. Singular Value Decomposition (SVD) and eigenvector decompositions belong to eigen decomposition. Usually, SVD can be applied into a non-squared matrix, whereas eigenvector decomposition should be applied into squared matrix form.

For a document-term frequency matrix analysis, SVD is used for two reasons. First, the main purpose of the matrix decomposition is to find the most important
features among the observed matrix. Second, the dimension of a corpus and the length of a dictionary is not the same, or it is a non-squared matrix. The detailed mathematical background will be addressed in the next section.

2.2.1 Principal Component Analysis and Singular Value Decomposition

Generally, a document-term frequency vector for describing a document gives a clue about how much co-occurring words are semantically related. However, finding co-occurring patterns and discovering the most important correlations between them is not a trivial matter because that vector is described in a high-dimensional vector space. In order to reveal the most important relationships from the observed matrix in a high dimensional vector space, Principal Component Analysis (PCA) is generally suggested as an analyzing method.

Based on the Singular Value Decomposition (SVD), PCA turns out the most highest eigenvalues and corresponding eigenvectors as the principal component of the covariance matrix of the given data. Therefore, once eigenvectors are found in the covariance matrix, the next step is to order them by eigenvalue, highest to lowest. This provides the components in their order of significance.

This section shows the definition of SVD, and its subsequent terms, such as eigenvalues and eigenvectors, are addressed.

**Eigenvalues and Eigenvectors**

The main idea behind eigenvectors and eigenvalues is based on linear transformations. The common prefix, *eigen*, means *intrinsic* or *characteristic*. So, the eigenvector implies the characteristic vector and an eigenvalue means a characteristic value of the corresponding eigenvector.

Let \( A \) be an \( n \times n \) squared transformation matrix, and let \( \mathbf{x} \) be a point vector
in \( n \)-dimensional space. If there exists a scalar \( \lambda \) satisfying \( Ax = \lambda x \), then \( x \) is a
eigenvector of \( A \) and \( \lambda \) is the corresponding eigenvalues of \( x \). In other words, the
eigenvector keeps its vector property regardless of a transformation. That is the
reason the eigenvector is called the characteristic vector of the given transformation
matrix \( A \).

**Singular Value Decomposition**

Eigen decomposition is a factorization of a matrix using eigenvalues and eigenvectors.
Generally eigen decomposition works on a squared matrix. For a non-squared matrix
\( A \), the singular value decomposition is applied and it uses a covariance matrix of \( A \),
or \( AA^T \).

Let \( A \) be an \( m \times n \) matrix with \( m \geq n \). The matrix \( A \) is factorized into a product
of \( U \), \( \Sigma \), \( V^T \), and the factorized form \( U\Sigma V^T \) is called the *Singular Value Decomposition*
of \( A \).

\[
A = U\Sigma V^T \tag{2.2.1}
\]

where \( U \) is the eigenvectors of \( AA^T \) and an \( m \times m \) orthogonal matrix, \( V \) is the eigen-

vectors of \( A^TA \) and an \( n \times n \) orthogonal matrix. \( \Sigma \) is an \( m \times n \) matrix, where off-
diagonal elements are all zeros and diagonal elements are denoted as \( \sigma_1, \sigma_2, ..., \sigma_n \)
with \( \sigma_1 \geq \sigma_2 \geq .... \geq \sigma_n \). All \( \sigma_i \) are eigenvalues of \( AA^T \) and are called *singular values*
of $A$ and $\Sigma$ is noted as follows:

$$\Sigma = \begin{pmatrix}
\sigma_1 & 0 & \cdots & 0 \\
0 & \sigma_2 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & \sigma_n \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0
\end{pmatrix} \tag{2.2.2}$$

In this decomposition, $\Sigma$ indicates which elements are important features by the ordered eigenvalues. Most important $l$ eigenvalues are selected from the $\Sigma$, then the reduced $\Sigma$ is rewritten as $\Sigma_1$, with the size of $l \times l$. Then $\Sigma$ is written as:

$$\Sigma = \begin{bmatrix}
\Sigma_1 & 0 \\
0 & 0
\end{bmatrix} \tag{2.2.3}$$

Thus, the matrix $A$ is decomposed with reduced elements, $\Sigma_1$

$$A = U \Sigma V^T$$

$$= \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix}
\Sigma_1 & 0 \\
0 & 0
\end{bmatrix} \begin{bmatrix}
V_{\Sigma_1}^T \\
V_{\Sigma_2}^T
\end{bmatrix}$$

$$= U_1 \Sigma_1 V_{\Sigma_1}^T$$

$$= AV_{\Sigma_1}^T$$ \tag{2.2.4}

$$= A$$ \tag{2.2.5}

The (2.2.4) shows a matrix $A$ is decomposed of $U_1$, $\Sigma_1$, and $V_{\Sigma_1}^T$ as well. The covariance matrix $AA^T$ indicates how each element is correlated, thus the eigenvector
of $AA^T$, $U$ discovers the principal components of the correlation elements. So, PCA selects the most important features by selecting the largest $l$ eigenvalues from $\Sigma_1$.

### 2.2.2 Latent Semantic Analysis

In Section 2.1, the document-term frequency matrix is introduced as a simple representation of a document indicating word frequencies, as well as their co-occurrences. Although a whole document is described as a vector space by ignoring the order of words, however, the vector’s dimension is still too large. Since the column space of the document-term matrix spans the same size as the vocabulary set, the analysis of a document using directly observed words requires very expensive computational complexity.

In order to reduce the vector dimension of a document-term frequency matrix, a matrix decomposition method is suggested. Specifically, PCA is applied to that matrix to reduce the vector dimension by selecting the most important correlated features of a given data set. So, a SVD decomposition is performed for a document-term matrix.

Deerwester et al. presented the Latent Semantic Analysis (LSA) for discovering latent semantic features of a document using SVD [28]. Figure 2.1 illustrates the main concept of the LSA. Let $X$ be a term-by-document matrix. Then the covariance matrix of $X$, $XX^T$ shows the degrees of correlation among the words set. Therefore, $l$—largest eigenvalues of $XX^T$, which are represented as $D$, indicate the most important features among the correlation of words. Therefore, the matrix $D$ is called a latent vector space of $XX^T$, and is also called a topic space. The eigenvector of $XX^T$, denoted as $U$, indicates how the words are represented in the transformed topic space. Similarly, the eigenvector of $X^TX$ or $V^T$ implies how each document is represented in topic space.
2.3 Statistical Approach

In this chapter, I introduce two approaches to the topic inference method by analysis data. Section 2.1 introduced the data set and its representation for a computational analysis and section 2.2 presents a data analysis approach based on the matrix transformation for discovering latent features. This section introduces a statistical approach for discovering latent topics.

As a data analysis tool, statistical approaches are the most popular methods used today. The main goal of statistical data analysis is to make an inference from the given data set, while data analysis using a matrix decomposition method derives latent intrinsic features by transforming into subspace.

Inference in statistics is generally categorized as one of two types. One is the estimation of the probability distribution of the model, which is derived from a given data set; thus, inference is focused on discovering statistical model parameters. The other type of inference is predicting the probability for new observation data.
Therefore, this section will cover discovering the latent topic model based on a statistical approach. First, the background knowledge for statistical inference is provided. In order to set up a model, variables are introduced and then model parameters are introduced. Finally, the model parameter estimation is introduced.

### 2.3.1 Statistical Inference

Let \( \mathcal{X} \) be an observation set and \( \theta \) be a model parameter set. The first goal of statistical inference is to discover a probability distribution that explains well the given data set, \( \mathcal{X} \), and to estimate the model parameter for the distribution, \( \theta \). The first thing needed for parameter set estimation is to model the probability distribution for the observed data, \( p(\mathcal{X}) \). However, it is sometimes not feasible due to the sparseness of samples. So the most common way to obtain it is simply to use an assumption that a probability density of \( p(\mathcal{X}) \) follows a well-known distribution, such as a Gaussian distribution or Poisson distribution, etc. For example, if a data set \( \mathcal{X} \) follows a normal distribution, then it is represented as \( \mathcal{X} \sim \mathcal{N}(\mu, \sigma^2) \). Model parameters are notated with mean, \( \mu \), and standard deviation, \( \sigma \); so a parameter set, \( \theta \) represents with \( \theta = \{\mu, \sigma\} \).

Estimating model parameters is to discover instance values of \( \mu \) and \( \sigma \) in this example. For this purpose, the maximum likelihood estimation is performed. In statistics, likelihood, \( p(\mathcal{X} | \theta) \), is represented as a function of the parameters of a statistical model. The likelihood of a set of parameter values, given some observed outcomes, is equal to the probability of those observed outcomes, given those parameter values:

\[
L(\theta | \mathcal{X}) = p(\mathcal{X} | \theta) = \bigcap_{x \in \mathcal{X}} \{X = x | \theta\} = \prod_{x \in \mathcal{X}} p(x | \theta)
\]  

(2.3.1)
Thus, a parameter set that has maximum likelihood is calculated as written below:

\[
\hat{\theta}_{ML} = \arg \max_{\theta} L(\theta | \mathcal{X}) = \arg \max_{\theta} \sum_{x \in \mathcal{X}} \log p(x | \theta)
\] (2.3.2)

Next, as with the other types of statistical inference, predicting the probability of new observed data, \(\tilde{x}\) regarding to the given data set \(\mathcal{X}\) is shown as written below:

\[
p(\tilde{x} | \mathcal{X}) = \int_{\theta \in \Theta} p(\tilde{x} | \theta)p(\theta | \mathcal{X})d\theta
\]

\[
\approx \int_{\theta \in \Theta} p(\tilde{x} | \hat{\theta}_{ML})p(\theta | \mathcal{X})d\theta
\] (2.3.4)

\[
= p(\tilde{x} | \hat{\theta}_{ML})
\] (2.3.5)

This section shows the basics of statistical inference for estimating model parameters and predicting a new probability. The following section will address how to build a statistical model for observation data.

### 2.3.2 Statistical Modeling

Statistical approaches explain a given data set using a probability distribution. Indeed, various probability distributions already have been introduced in the statistics fields, such as normal distribution, multinomial distribution, or Bernoulli distribution, etc. However, some types of a data set do not fit into the known distributions to explain the observation, so they need to build their own statistical model, which is expressed with a mathematical expression using parameters.

A document model, shown in section 2.1, is one of the examples. In order to provide a probability distribution of a document, statistical language models, such as the bigram model or n-gram model, have been proposed in the natural language processing field. For example, if a document is composed of \(m\) words, then the
probability of that document would be \( p(w_1, w_2, ..., w_m) \). The probability is realized with a joint probability of \( m \) random variables. In this case, the possible instance of each word, \( w_i \) is the same of the number of vocabularies, \(|V|\). Thus all the possible events would be \(|v|^m\). So the estimated probability, \( p(w_1, w_2, ..., w_m) \approx \frac{1}{|v|^m} \), is very small or close to zero.

In order to handle the complexity of a joint probability in multivariate variables, many statistical language models use the Bayesian network to reduce the burden of joint probability. The Bayesian network depicts the conditional dependencies of variables and then rewrites a joint probability into a conditional probability. So the Bayesian network effectively rewrites a high dimensional multivariate joint probability into the lower-dimensional probability distributions.

The rest of this section explains the Bayes’ rule and introduces several statistical models which are based on the Bayesian graphical model.

**Bayes’ Theorem**

In order to explain the Bayes’ theorem, joint probability and conditional probability are explained first. Let us assume there are two events \( A \) and \( B \). The probability of event \( A \) is \( p(A) \) and the probability of event \( B \) is \( p(B) \). The joint probability of two events, \( A \) and \( B \), is notated as \( p(A, B) \), and it is the same as the product of each event probability, \( p(A, B) = p(A)p(B) \) if event \( A \) and \( B \) are independent of each other.

Conditional probability is referred to as the probability of event \( A \) with the given event \( B \), then written as \( p(A|B) \). Thus, \( p(A|B) \) is also the proportional ratio of the joint probability to the probability of event \( B \), \( p(A|B) = \frac{p(A, B)}{p(B)} \). So the joint probability, \( p(A, B) \), is organized as \( p(A, B) = p(A|B)p(B) \).

The Bayes’ theorem shows the relationship between two conditional probabilities
between \( p(A|B) \) and \( p(B|A) \):

\[
p(A|B) = \frac{p(A, B)}{p(B)} = \frac{p(B|A)p(A)}{p(B)}
\]

(2.3.6)  
(2.3.7)

The Bayes’ theorem is used to measure a degree of belief. It then links the degree of belief in a proposition before and after accounting for evidence. For example, a probability of evidence, \( p(B) \), and a degree of belief \( A \), \( p(A) \) are given. Also, if the likelihood of the event \( B \) occurs and the given \( A \) condition is known, then the degree of belief considering the event \( B \) is obtained through the Bayes’ theorem.

- prior, \( p(A) \) is a degree of belief in \( A \).
- evidence, \( p(B) \) is the probability of event \( B \).
- likelihood, \( p(B|A) \) shows how likely an event \( B \) occurs under the given condition \( A \).
- posterior, \( p(A|B) \) is the degree of belief having accounted for \( B \).

**Graphical Model and Bayesian Network**

The Bayes’ theorem also brings the benefit of interpretation of a statistical model. The joint probability of having a multivariate data set is represented with the conditional probability based on the dependencies of variables. It reduces the multivariate dimensionality for random variables as shown in 2.3.2.

In order to show the conditional dependencies among variables, a graphical model is generally used, where a graphical model combines two concepts: a probability model in statistics and a graph theory in mathematics. Each node of a graph represents variables of a statistical model and the directed edge of the graph indicates a
conditional dependency between variables. Also, if two nodes do not have an edge, then those two variables are independent. Specifically, a graphical model describing conditional dependencies among variables is called a Bayesian Network.

Figure 2.2 shows an example of a graphical model. The model has five variables, $\alpha$, $\beta$, $\theta$, $z$, and $w$, where each variable is represented with a node or circle. In this figure, one shaded node and four white nodes are shown. The shaded node represents an observed variable for the statistical model; the four white nodes indicate latent variables. This model also shows several edges. An edge between $\alpha$ and $\theta$ shows the variable $\theta$ has a dependency on $\alpha$. So, there is a conditional probability between $\alpha$ and $\theta$ such as $p(\theta|\alpha)$. Similarly, other directed edges are interpreted the same way.

The model also uses two rectangle notations. One includes $z$ and $w$ and the other contains $\theta$, $z$, and $w$. Each rectangular notation, which is called a plate notation, implies that the relationships of variables are repeated as $N$ times or $M$ times.

From the graphical model depicted, the probability of this model can be regarded
as:

\[ p(\alpha, \beta, \theta, z, w) \]  \tag{2.3.8}

According to the dependencies among the variables, (2.3.8) is decomposed using the following conditional probabilities:

\[ p(\alpha, \beta, \theta, z, w) = p(\theta|\alpha)p(z|\theta)p(w|z)p(w|\beta) \]  \tag{2.3.9}

### 2.3.3 Statistical Document Model

This section introduces two of the most popular statistical document models using the Bayesian Network. The first model is known as the Probabilistic Latent Semantic Analysis (PLSA) [55] and the other is known as the Latent Dirichlet Allocation (LDA) model [13].

**Probabilistic Latent Semantic Analysis**

In the previous section 2.2.2, Latent Semantic Analysis (LSA) was introduced as a method for discovering the latent topic space. LSA analyzes a co-occurrence matrix using a singular value decomposition. The form of data is described in a two-mode matrix, a document-term frequency matrix. The row of the matrix is a set of documents and the column of the matrix is a set of vocabularies. For example, there is a corpus, \( \mathcal{C} \), having \( M \) documents, \( \mathcal{C} = \{ d_1, ..., d_M \} \) and they are composed with \( N \) words selected from a set of vocabularies, \( \mathcal{V} = \{ w_1, ..., w_V \} \). Then the form of the co-occurrence matrix is a \( M \times V \) matrix and the frequency of \( w_j \) in a document \( d_i \) is notated as \( n(d_i, w_j) \).

PLSA analyzes a document-term frequencies matrix using a statistical approach. To apply the statistical approach, first, a probability distribution for describing a
document-term frequency data should be defined. As shown in Figure 2.1, the given document model describes a document in rows and words in columns, respectively. Thus this model has two observed variables: \( d \) and \( w \) indicate a document and a word, respectively. The topic to be discovered is a latent variable in this model, and is noted as \( z \). Figure 2.3 depicts the PLSA model using a graphical model representation [55].

The probability of words and document co-occurrence data, \( p(d, w) \) is represented as:

\[
p(w, d) = p(d) \sum_{z \in Z} p(w|z)p(z|d)
\]  

(2.3.10)

Once the probability distribution of a given data set is approximated, then a latent variable, \( z \), is estimated using a maximum likelihood estimation. The estimation process is performed through the Expectation-Maximization (EM) algorithm [31]. The EM algorithm runs with two steps: one is expectation, called an E-step, which computes posterior probabilities of the latent variables; and the other is a maximization
step, or M-step. It updates parameters. An estimation step is shown as:

\[ p(z|d, w) = \frac{p(d)p(w|z)p(z|d)}{p(d)\sum_{z \in Z} p(w|z)p(z|d)} \]  

(2.3.11)

Also, a maximization step is shown as:

\[ p(w|z) \approx \sum_{d \in D} n(d, w)p(z|d, w) \]  

(2.3.12)

\[ p(d|w) \approx \sum_{w \in W} n(d, w)p(z|d, w) \]  

(2.3.13)

\[ p(z) \approx \sum_{d \in D} \sum_{w \in W} n(d, w)p(z|d, w) \]  

(2.3.14)

The PLSA model has advantages over the LSA based on the SVD decomposition, because the PLSA supports multiple concepts of topics. With the assumption that each word \(w_i\) is conditional on its topic \(z\), the posterior probability shown in (2.3.11) states the degree of the meaning of a word depends on its context.

**Latent Dirichlet Analysis**

LDA is a statistical bag-of-words model which is the most frequently applied document model. The difference between the LDA and PLSA models is LDA uses a Dirichlet distribution as a conjugate prior.

The flow of an LDA process for generating documents is given in Algorithm 1. In order to explain the algorithm, let us first introduce the notations for the word, document and corpus. A word is denoted by \(w\) and belongs to a set of known words provided in a dictionary, \(w \in D\). When an unordered list of \(N\) words is used together, the words compose a document \(d = (w_1, w_2, ..., w_N)\) in which some words can be repeated. Finally, a corpus is a set of \(M\) documents, \(C = \{d_1, d_2, ..., d_M\}\) in which the document contents may be mutually exclusive.
Algorithm 1 Generative Process of LDA Model. [13]

1: Let $Psn$, $Dir$ and $Mul$, respectively, represent Poisson, Dirichlet and Multinomial distributions
2: for each of the $M$ documents, $d_i$ do
3: Choose number of words, $N_i$, in $d_i$ from a $Psn(\xi)$
4: Choose topic distribution $\theta_i$ from $Dir(\alpha)$
5: for $j$ from 1 to $N_i$ do
6: Choose a topic $z_{ij}$ from $Mul(\theta_i)$ where $1 \leq z_{ij} \leq K$
7: Choose a word distribution $\phi_{z_{ij}}$ from $Dir(\beta)$
8: Choose a word $w_{ij}$ from $p(w_{ij}|\phi(z_{ij}))$
9: end for
10: end for

Given these definitions, we need two Dirichlet distributions in order to generate a multinomial distribution for a corpus. The first is for modeling the document-topic distribution and is denoted by $Dir(\alpha)$, and the second is for modeling the topic-word distribution, $Dir(\beta)$. Each document, $d_i$, is generated by first choosing the number of words, $N_i$, from a Poisson distribution. For this document, I chose a topic distribution, $\theta_i$, from the prior Dirichlet distribution, $Dir(\alpha)$ with $\alpha$ as its known parameter. Once $\theta_i$ is known, a set of $N_i$ topics can be selected for each word from a multinomial distribution, $Mul(\theta_i)$. The selected topic $z_{ij}$ provides a means to select a word from the distribution $\phi_{z_{ij}}$ based on the dictionary and the prior $Dir(\beta)$. Finally, the word distribution is used to pick a word $w_{ij}$ by exploiting the conditional probability, $p(w_{ij}|\phi(z_{ij}))$. At the end of this process we will have two lists: one for chosen words $d = (w_1, \ldots, w_N)$ and another for corresponding topics $z = (z_1, \ldots, z_N)$.

The probability of a word, (2.3.8), is factorized into conditional probabilities based
Figure 2.4: Graphical representation of the LDA model. Observed variable \( w \) is denoted by a shaded circle, and latent variables \( \theta \) and \( \phi \) are denoted by white circles. Each rectangle, which is called a plate, indicates a repetitive process. \( \alpha \) and \( \beta \) are corpus level hyperparameters for the Dirichlet distributions. In this model \( \alpha \) provides the condition for generating \( M \) multinomial distributions, \( \theta \), and \( \beta \) provides the condition for generating words distribution.

on the hierarchical Bayesian network of Figure 2.4.

\[
p(w_{ij}, z_{ij}, \theta_i, \phi_{zi_j} | \alpha, \beta) = p(w_{ij} | z_{ij}, \phi_{zi_j}) \\
p(z_{ij} | \theta_i)p(\theta_i | \alpha) \\
p(\phi_{zi_j} | \beta),
\]

(2.3.15)

Therefore, based on equation (2.3.15), the probability of a document having multiple words and topics is represented as:

\[
p(\theta_i, \Phi, \mathbf{z}, d | \alpha, \beta) = p(\theta_i | \alpha) \prod_{j=1}^{N} p(z_{j} | \theta)p(w_{j} | z_{j}, \phi_{z_{j}})p(\phi_{z_{j}} | \beta),
\]

(2.3.16)

where \( \Phi = [\phi_1, \phi_2, \ldots, \phi_K] \) and \( 1 \leq K \).
2.4 Bag-of-Visual Words Model in Image Analysis

The success of the bag-of-words model for document analysis has led to its adaptation for the image analysis problem in the computer vision field based on the conjecture that an image is worth a thousand words. In this context a word in a document is considered a visual-word in an image. The LDA adaptation for the image domain seeks to discover the latent content in images, such as combining pixels into segments. For instance, Fergus et al. [38] classified objects in an image after inferring the latent topics of an image. Similarly, Sivic et al. [116] suggested their use for object categorization. In related works, Fei-Fei and Perona [37] proposed to use the LDA model for scene categorization. Bosh used the PLSA model for the same problem [15]. For remote sensing, Lienou et al. [76] applied the LDA model for satellite image classification.

All these methods use the RGB or panchromatic band for analyzing images and introduce predefined visual-words, since the images do not contain semantic units comparable to a word in a document. Different researchers have proposed different visual words with respective descriptors, such as the descriptors generated from either the SIFT key-point detector [77] or the vectorized image patches. Thus those visual words tend to be dependent images, or they can not be universally applied. Due to the locality of the content, these visual descriptors, however, cannot be attributed to semantic meanings; hence, inferring latent topics is limited.
CHAPTER 3

BAG-OF-MATERIALS MODEL

Due to its outstanding ability to detect materials, hyperspectral imaging is actively used to generate thematic maps in remote sensing. Two processes are required to generate a thematic map, identifying a pixel and assigning a proper class label. However, an image that is acquired from a high altitude generally covers a large ground surface area, so one pixel in this type of image covers several objects of ground surface. Thus, few pixels consist of the pure material of an object. In fact, most of the pixels in an urban area image tend to contain multiple objects, so identifying major themes for such a pixel requires further consideration.

This chapter proposes a novel method for discovering pixel themes that play the role of the representative descriptor of a pixel in a hyperspectral image. The proposed approach starts with the assumption that each pixel of a remote sensing image covers more than one object on the Earth’s surface; thus assigning a class to that pixel is very fuzzy. However, when we consider the materials that compose a pixel, then find the comprehensive characteristics of the contained materials, it provides semantic meaning for that pixel. The model is called a “Bag-of-Materials Model,” since each pixel contains multiple properties of the objects, and the goal of this model is to identify the characteristics of a bag or, a pixel, using the materials contained. For that reason, one of the key tasks is discovering the latent themes within the mixture
materials. Also, the degree of the themes, an indicator of how close a pixel is to a
theme, is described as a probability distribution.

The rest of this section addresses the flow of the proposed algorithm. The first
section introduces used notations and a preprocessing method for applying our bag-
of-materials model. The second section depicts the suggested model using a statistical
approach. In order to visualize the model, I used the Bayesian network to show con-
ditional dependencies between model description variables. The final section defines
model parameters and shows the way to estimate them.

3.1 Data Preparation

The core concept of the proposed approach is the identification of the pixels’ char-
acteristics by considering the co-occurring materials. Thus, the suggested method
considers that a pixel is a container of multiple materials, and each pixel is described
by the materials observed within it, called *endmembers*, and their proportions called
*abundances*. However, the observed spectrum of a pixel results from the combination
of the mixture spectra of many materials; thus, spectral unmixing is usually applied
as a pre-process to obtain endmembers and their abundance of pixels. In this un-
mixing process, I use two assumptions. First, mixture spectra are linearly combined.
Second, for the spectral unmixing, the spectral signatures of materials are given by
a known spectral library. Using this analogy, a document contains many words, and
its main topics are inferred from the words appearing in the document. The topics
of a pixel are discovered from the result of spectral unmixing.

This section introduces several notations for describing the bag-of-materials model
and a required pre-processing. The notations are a spectral library, a pixel, and an
image. In this notation, the spectral signature of a material in a given spectral library
is regarded as a visual word for the proposed model. Also, a pixel is regarded as a bag
of visual words – like a document, and an image is regarded as a corpus, a set of documents. In order to decompose an observed spectrum of a pixel, a spectral unmixing method is introduced where a set of spectral signatures of materials are known and the combination of materials are linear. Note that I will follow the aforementioned symbology, but the variables in this case become vectors, which will be denoted by boldface letters.

3.1.1 Notations

Spectral Library

Spectral library, \( \mathcal{D} \), provides the dictionary containing the set of spectral signatures of known materials which are referred to as the endmembers:

\[
\mathcal{D} = \{ \mathbf{w}_1, ..., \mathbf{w}_V \},
\]  

(3.1.1)

where \( \mathbf{w}_i \) is a spectral signature of \( i^{th} \) indexed material in the library and each spectral signature is represented with a \( L \)-dimensional vector, if a hyperspectral image has \( L \)-spectral channels. Also, \( V \) indicates the number of materials in the library or the size of a dictionary \( |\mathcal{D}| = V \).

Spectral Reflectance of a Pixel

The spectral reflection information \( \mathbf{s}_i \) at pixel \( i \) is the reflectance values of the \( i^{th} \) pixel of a hyperspectral image. The spectral values of \( \mathbf{s}_i \) are depicted with \( L \)-dimensional vectors in the same number as the spectral bands. Also, \( \mathbf{s}_i \) at pixel \( i \), is assumed that the spectrum is linearly combined with spectra of multiple endmembers \( \mathbf{w}_j \in \mathcal{D} \) with associated abundances \( a_j \). It is represented as follows:

\[
\mathbf{s}_i = \sum_{j=1}^{V} a_{ij} \mathbf{w}_j,
\]  

(3.1.2)
where \(0 \leq a_{ij} \leq 1\) and \(\sum a_{ij} = 1\), since the abundances cannot be a negative value and they are represented with the proportion of the combination weight. The abundance for each material generates the abundance vector of \(i^{th}\) pixel:

\[
a_i = (a_{i1}, a_{ij}, ..., a_{iV})
\]

(3.1.3)

that indicates how the \(j^{th}\) material, or visual word, contributes to the combination. The abundance vector is denoted as the combination percentages of visual words.

**Image**

An image, \(I\), is a collection of \(M\) pixels having observed spectral values \(s_i\):

\[
I = \{s_1, ..., s_M\}.
\]

(3.1.4)

where \(M\) is the same as the number of rows multiplied by the number of columns; thus an image corresponds to a corpus, which is a collection of documents.

**Document**

In this setting, a document \(d_i\) corresponds to the observed spectral signature of a pixel \(s_i\), and the document is considered to contain \(N\) endmembers, or visual-words. The endmembers are identified by spectral unmixing; thus their abundances are provided as a ratio or percentage, as a result of spectral unmixing. In order to consider the natural numbers, I simply multiply this 100 times; thus the number of words used, \(N\), would be 100 and the fractions of the ratio turned into the natural number of weights:

\[
d = (w_1, w_2, ..., w_N),
\]

(3.1.5)

where \(w_i \in D\) and the same visual word can be repeated multiple times according to its abundances. For example, if an abundance of endmember \(w_j\) is 0.2 and that of \(w_k\)
is 0.8, then a document $d$ is consists of 20 times repeated $w_j$ and 80 times repeated $w_k$. So, the length of this documents is 100, or the number of words for a document is 100 even though only two types of endmembers appear.

**Corpus**

An image having $M$ pixels generates $M$ documents and the set of documents is called a corpus:

$$C = \{d_1, ..., d_M\}.$$  \hfill (3.1.6)

### 3.1.2 Spectral Unmixing

In order to generate a document given in (3.1.5) from the observed spectral signature, $s_i$ of pixel $i$, I first decompose $s_i$ by spectral unmixing. In order to realize (3.1.2), I adopt an unmixing algorithm that will estimate the abundances $a_i$. Specifically, I used the assumption that the endmembers are linearly combined, thus the least square estimation is employed to solve an abundance vector [53].

Let both the observed spectral signatures and the endmembers be collected from $L$ spectral bands:

$$s = (s_1, s_2, ..., s_L)^\top, \quad w_i = (w_{i1}, w_{i2}, ..., w_{iL})^\top.$$  

Arranging the endmembers into a matrix $W = [w_1, w_2, ..., w_V]$ converts (3.1.2) into:

$$s = Wa.$$  \hfill (3.1.7)

In this equation, the only unknown variable is the abundance vector of endmembers, $a$, since a matrix form of a spectral library, $W$, is given and $s$ is the observed spectral
reflectance values. So the abundance vector, $a$, can be estimated by least squares minimization [53]:

$$\min_a \| Wa - s \|_2,$$

(3.1.8)

where $\| \cdot \|_2$ represents the $L_2$ norm. Rearranging (3.1.7) results in the abundance vector $a$ given by:

$$a = (W^T W)^{-1} W^T s.$$  

(3.1.9)

Direct application of the least squares, as given in (3.1.9), may, however, producing negative abundances, which are physically impossible. Therefore, I introduce $a_i \geq 0$ as an additional constraint to enforce positive abundances.

The abundance values computed using the non-negative least squares provide a descriptor for the pixel in terms of the endmembers in the library, and implicitly changes the dimensionality from the number of bands to the number of endmembers. This new descriptor serves as the histogram of the endmembers (words) for the document, representing the pixel as given in (3.1.5).

### 3.2 Statistical Model for Bag-of-Materials

The goal of the bag-of-materials model is to find hidden pixel themes, and the latent themes are statistically inferred from the observations. This section addresses the proposed model using a statistical approach. Thus, the observed data set should be described with a probability distribution. In other words, the given data is described with model parameters of the distribution. However, some of the data set cannot be explained with already published distributions, so a proper probability distribution to fit the given data set should be proposed. Therefore, finding a statistical model is to express the given data set with a probability density function using several parameters.
Finding a model in statistics is achieved in two ways. The first is to derive a model through the generative model and the second is to derive a model through the discriminative model. In this dissertation, I explained the proposed bag-of-materials model for a hyperspectral image using a generative model. The remainder of this section describes a generative model and depicts the proposed model using a Bayesian Network where the dependencies among the model variables are shown. After defining a probability density function for the proposed bag-of-materials model, parameters of the model are introduced and estimated. For the estimation, I adapted Gibbs sampling for this distribution.

3.2.1 Generative Model for Hyperspectral Images Analysis

Among many statistical models, a generative model explains the data generation process in the view of how observable data are randomly produced and what causes that to happen. Thus, a generative model focuses on revealing the casual relationship between observed variables and latent variables. Generative models usually consider the observable variables and latent variables together; thus probability distribution is explained with a joint probability of variables. On the other hand, a discriminative model considers a conditional probability between observed variables and unobserved variables; thus, discriminative models are applied to find supervised classification and regression models.

Figure 3.1 shows the relationship between the observed variable and latent variable, where the shaded circle indicates an observable variable and the white circle indicates a latent variable. A latent variable in a generative model works as a cause factor of output data, and a latent variable in a discriminative model works as a target variable, such as a class label. Even though the indicated meanings of a latent variable are different, the latent variable of each model has one thing in common in
that they are not directly observed, but inferred from the given data set through a statistical model.

In this dissertation, I selected a generative model to discover the pixels’ representative themes, whereas much of the previous work has been done using a discriminative model, such as a support vector machine (SVM) to assign a class label for a pixel. The goal of this research is to find latent factors of pixel generalization in a condition where there is no given training set, composed of an observation and its class label. Thus, in this circumstance, a generative learning model is more suitable.

In order to design a generative statistical model for the proposed bag-of-materials model, consider several examples. Here is an example of a pixel having multiple materials and multiple topics: a pixel represents a residential area where grass, asphalt road, and part of a roof can be included. In other words, that pixel may be regarded as not only a vegetation area, but also a man-made structural area. Also, the asphalt
material appears not only in the road, but also in parking lots. Through these examples, I must make the following considerations:

- An image is composed of $M$ pixels;
- Each pixel is composed of multiple endmembers, or $N$ materials;
- Each pixel has multiple themes or topics;
- Each endmember is regarded as a visual word and may have several meanings, or polysemes;
- Each endmember belongs to multiple topics; and
- Each topic has a group of visual words having similar meaning.

From the above considerations, suppose there are $K$ topics, $1 \leq k \leq K$, and $k$ indicates the index of a topic. Thus $k$ means the $k^{\text{th}}$ topic of the topic lists. A material or endmember that appears in a pixel is considered as a word and noted as $w$. Since each material, or word, has semantics, the representative meaning of a material belongs to some of the topics. Thus, a latent topic of each endmember in a certain context is noted as $z$, where $z \in \{1, ..., K\}$. So, a topic of a word, $z$, may vary according to the document’s context. Topics of a document are noted as $\theta$ and described with a multinomial distribution, $p_{\theta_i} = (p_1, ..., p_K)$ and $\sum_{i=1}^{K} p_i = 1$, since a pixel can be allowed to have multiple topics. Generally, the multinomial distribution is assumed to follow the Dirichlet distribution. So, the multinomial distribution of a document is represented as $\theta \sim \text{Dir}(\alpha)$. Similarly, each topic is composed of many words, which have a related meaning. Each topic is noted as a multinomial distribution of words, $\phi$. This distribution also assumed that it follows the Dirichlet distribution, $\phi \sim \text{Dir}(\beta)$. Those multinomial distributions, $\theta$ and $\phi$, are conditioned
on the Dirichlet distributions parameterized by $\alpha$ and $\beta$, respectively. In summary, given the above consideration, the variables for a model are defined as:

- $w$, a material or endmember as a word, an observed variable;
- $z$, a latent topic instance of an endmember according to a context, where $z \in \{1, \ldots, K\}$, a latent variable;
- $\theta$, multinomial topics distribution of a document, a latent variable;
- $\phi$, multinomial words distribution of a topic, a latent variable;
- $\alpha = [\alpha_1, \ldots, \alpha_k, \ldots, \alpha_K]$, a parameter of the Dirichlet distribution, $\text{Dir}(\alpha)$; and
- $\beta = [\beta_1, \ldots, \beta_v, \ldots, \beta_V]$ a parameter of the Dirichlet distribution, $\text{Dir}(\beta)$.

Using those variables, the joint probability of a material in the proposed model is computed as:

$$p(w, z, \theta, \phi, \alpha, \beta)$$  \hfill (3.2.1)$$

However, the joint probability is rewritten using the Bayes’ Rule in order to reduce the computational complexity. So, the dependencies of variables are examined and then depicted with a Bayesian Network.

Since the proposed model uses a generative model, the explanation of the probability distribution begins with how each pixel is produced by the composing endmembers. First, a multinomial topic distribution of the $i^{th}$ pixel is selected from $\text{Dir}(\alpha)$. After determining $\theta_i$, the topic instance for an endmember that composes a pixel follows the multinomial distribution, $z_{ij} \sim \text{Mult}(\theta_i)$. So, the instance value of $z_{ij}$ is chosen from $\{1, \ldots, K\}$. Also, a multinomial endmembers distribution of the $k^{th}$ topic is represented with $\phi_k$ and noted as $\phi_k \sim \text{Dir}(\beta)$. This distribution indicates
that each endmember or material belongs to some of topics. In order to generate a
document, or a pixel of mixture materials, $N$ of materials are chosen according to
the topic distribution of a pixel, $\theta_i$. So, the probability of a word is considered as
$p(w_{ij}|z_{ij}, \phi_{z_{ij}})$

Figure 3.2 shows the relationships between variables. In a Bayesian framework,
the conditional dependencies among the variables are illustrated as an edge of a
network, and variables are illustrated as nodes. Specifically, an observed variable is
depicted as a shaded node, and a latent variable, which must be inferred, is depicted
as a non-shaded node to make a distinction. For the bag-of-materials model, each endmember is an observed variable and latent topics are latent variables.

### 3.2.2 Probability Density Functions for Hyperspectral Image

This section introduces a probability density function of the proposed model to express that model using variables shown in Figure 3.2. First, the probability density function is approximated using a generative model. Then, the approximated probability distribution is expressed using model parameters.

The joint probability of the proposed model can be decomposed with several conditional probabilities based on conditional dependencies which are shown in Figure 3.2. Thus the probability of a material defined in (3.2.1) is rewritten as:

\[
p(w_{ij}, z_{ij}, \theta_i, \phi_{z_{ij}}, \alpha, \beta) = p(w_{ij}, z_{ij}, \theta_i, \phi_{z_{ij}} | \alpha, \beta) \]
\[
= p(w_{ij} | z_{ij}, \phi_{z_{ij}}) p(z_{ij} | \theta_i) p(\theta_i | \alpha) (\phi_{z_{ij}} | \beta),
\]

where \(w_{ij}\) indicates the \(j^{th}\) material or endmember appeared in the \(i^{th}\) pixel. \(z_{ij}\) is a latent topic of the material \(w_{ij}\) in the context \(d_i\). \(\theta_i\) indicates a multinomial topics distribution of a pixel; thus the instance of \(z_{ij}\) is determined by a distribution \(\theta_i\). \(\phi_{z_{ij}}\) is the multinomial words distribution about the \(k^{th}\) topic, where \(z_{ij} = k\).

#### Probability Density Function of a Pixel

Based on the probability of a material shown in (3.2.3), the probability of a pixel is represented as:

\[
p(\theta_i, \Phi, z_i, d_i, \alpha, \beta) = p(\theta_i, \Phi, z_i, d_i | \alpha, \beta) \]
\[
= p(\theta_i | \alpha) \prod_{j=1}^{N} p(z_{ij} | \theta_i) p(w_{ij} | z_{ij}, \phi_{z_{ij}}) p(\phi_{z_{ij}} | \beta),
\]
where $d_i$ is a pixel having $N$ materials, $d_i = (w_{i1}, ..., w_{iN})$. $z_i$ is the sequence of $N$ topic instances corresponding to the sequence of $N$ materials, $d_i$, and implies $z_i = (z_{i1}, ..., z_{iN})$, where $1 \leq z_{ij} = k \leq K$. $\Phi$ is a set of multinomial distributions for $K$ topics, $\Phi = [\phi_1, \phi_2, ..., \phi_K]$.

Applying the Bayes’ rule to (3.2.4), the probability of a pixel is decomposed as two probability functions:

$$p(\theta, \Phi, z, d | \alpha, \beta) = \frac{p(\theta, \Phi, z, d | \alpha, \beta)}{p(\alpha, \beta)} = p(\theta, \Phi, z | d, \alpha, \beta) \frac{p(d, \alpha, \beta)}{p(\alpha, \beta)} = p(\theta, \Phi, z | d, \alpha, \beta) p(d | \alpha, \beta).$$

The probability density function of a pixel is derived by integrating over $\theta_i$:

$$p(d | \alpha, \beta) = \int p(\theta_i | \alpha) \prod_{j=1}^N p(z_{ij} | \theta_i) p(w_{ij} | z_{ij}, \phi_{z_{ij}}) p(\phi_{z_{ij}} | \beta) d\theta_i. \tag{3.2.8}$$

Similarly, the probability density function of an image, $C = \{d_1, ..., d_M\}$ is obtained by the product of all marginal probabilities of a pixel, (3.2.8).

$$p(C | \alpha, \beta) = \prod_{i=1}^M \int p(\theta_i | \alpha) \prod_{j=1}^N p(z_{ij} | \theta_i) p(w_{ij} | z_{ij}, \phi_{z_{ij}}) p(\phi_{z_{ij}} | \beta) d\theta_i. \tag{3.2.9}$$

### 3.3 Estimation of Model Parameters

Equation (3.2.8) represents a probability density function of a bag-of-material model, where the probability distribution function is described with variables, $\alpha$, $\beta$, $\theta$, $\phi$, $z$ and $w$. Among those variables, values of some can be given by observation or some of the values are fixed by a given constant. In order to reduce the number of unknown variables, generally the variable $\alpha$ and $\beta$ can be given. Because those variables are hyperparameters of a Dirichlet distribution, which are regarded as conjugate priors,
the values of \( \alpha \) and \( \beta \) can be set in some constant. In this model, \( w \) is an observable variable and \( z \) is a latent topic of a word, so the variable \( z \) is collapsed in both \( \theta \) and \( \phi \). Thus, I consider \( \theta \) and \( \phi \) as the model parameters for the bag-of-materials model.

The rest of this section addresses the model parameter estimation method.

### 3.3.1 Gibbs sampling for Probability Estimation

A statistical model should provide a distribution to predict an expectation using a parametric function. However, sometimes the distribution is not known or that distribution is intractable because the statistical model is explained with the joint probability of multivariate random variables. Therefore approximating the distribution is required in order to find model parameters that explain the distribution.

For that purpose, the Expectation-Maximization (EM) algorithm, Variational Inference, or Gibbs sampling are used. The EM algorithm obtains model parameters by repeating two steps: estimating the value of parameters and maximizing the likelihood. Gibbs sampling, which is based on a Markov chain Monte Carlo (MCMC) algorithm, approximates a distribution by a randomizing process where the observations sequence is simulated, and estimates model parameters. In this dissertation, I employed the Gibbs sampling method for approximating the probability distribution of the proposed bag-of-materials model and estimating parameters, since the Gibbs sampling is particularly well-adapted to sampling the posterior distribution of a Bayesian network.

Algorithm 2 introduces the concept of the Gibbs sampling. The probability distribution expressed with \( n \) random variables is estimated using \( T \) times the sampling repetition. Suppose that we want to obtain \( T \) samples of \( \mathbf{x} = \{x_1, \ldots, x_n\} \) from a joint distribution \( p(x_1, \ldots, x_n) \), then denote the \( i^{th} \) sample by \( \mathbf{x}^{(i)} = \{x_1^i, \ldots, x_n^i\} \).

That is, sample each variable from the distribution of that variable conditioned
Algorithm 2 Process of Gibbs Sampling

1: Randomly initialize each $x_i$
2: for $t=1$ to $T$ do
3: for $i = 1$ to $n$ do
4: $x_{i}^{t+1} \sim p(x_i | x_1^{t+1}, \ldots, x_{i-1}^{t+1}, x_{i+1}^{t}, \ldots, x_n^{t})$
5: end for
6: end for

on all other variables, making use of the most recent values and updating the variable with its new value as soon as it has been sampled.

The samples then approximate the joint distribution of all variables. Furthermore, the marginal distribution of any subset of variables can be approximated by simply examining the samples for that subset of variables. In addition, the expected value of any variable can be approximated by averaging over all the samples.

**Collapsed Gibbs Sampling**

In this study, I used the Gibbs sampling method to estimate the unknowns in (3.2.8). I particularly exploit the collapsed Gibbs sampling due to its simplicity and the fact that the latent variables in (3.2.8), $\phi_{z_i}$ and $\theta$, can be estimated based on $z_i$ [47] [52].

Before providing the details, let us first introduce the matrices used in the process. The corpus $C$ is a known $M \times N$ matrix generated from observed spectra as discussed above, where $M$ is the number of all the documents and $N$ is the maximum number of words used in each document, $N = \max_i \{N_1, \ldots, N_m\}$. The matrix $Z$ is an $M \times Z$ matrix keeping the topic index for a word $w$ within all the document $d_i$'s. The $Z$ matrix is generated by the Gibbs sampling process and is updated in each sampling iteration. For details for the generation of $Z$, we refer the reader to [52]. For iteration $t$, $Z^t$ is given, and two accumulator arrays $T$ and $P$ are computed to reflect the changes
Algorithm 3 Collapsed Gibbs Sampling Algorithm for LDA [52].

1: Initialize all count variables,
2: for all documents \( d \in [1, M] \) do
3:     for all words \( w \) in a document, \( d \), \( d = (w_1, \ldots, w_j, \ldots w_{N_m}) \), and \( w_j \in [v_1, \ldots, v_V] \) do
4:         sample topic index \( z_{d,w} = k \sim \text{Mult}(1/K) \)
5:         increase document-topic count : \( n^{(k)}_d = n^{(k)}_d + 1 \)
6:         increase document-topic sum : \( n_d = n_d + 1 \)
7:         increase topic-words count : \( n^{(v)}_k = n^{(v)}_k + 1 \)
8:         increase topic-words sum : \( n_k = n_k + 1 \)
9:     end for
10: end for
11: for all sampling \( t \in [1, T] \) do \( \triangleright \) repeat sampling procedure \( T \) times
12:     for all documents \( d \in [1, M] \) do
13:         for all words \( w \) in a document, \( d \), \( d = (w_1, \ldots, w_j, \ldots w_{N_m}) \), and \( w_j \in [v_1, \ldots, v_V] \) do
14:             decrease document-topic count : \( n^{(k)}_m = n^{(k)}_m - 1 \)
15:             decrease document-topic sum : \( n_m = n_m - 1 \)
16:             decrease topic-words count : \( n^{(v)}_k = n^{(v)}_k - 1 \)
17:             decrease topic-words sum : \( n_k = n_k - 1 \)
18:             sample topic index \( \tilde{k} \sim p(z_i | z_{-i}, d) \) \( \triangleright \) for new assignment of \( z_{d,w} \) to the
19:             term \( v \) for word \( w \)
20:             increase document-topic count : \( n^{(k)}_m = n^{(k)}_m + 1 \)
21:             increase document-topic sum : \( n_m = n_m + 1 \)
22:             increase topic-words count : \( n^{(v)}_k = n^{(v)}_k + 1 \)
23:             increase topic-words sum : \( n_k = n_k + 1 \)
24:         end for
25:     end for
26: if sampling is done then
27:     Set parameter \( \Theta \) from \( T \)
28:     Set parameter \( \Phi \) from \( P \)
29: end if
to the sampled topic indices in $Z$. $T$ is an $M \times K$ matrix keeping the frequencies of topics for documents, and $P$ is an $K \times V$ matrix keeping the frequencies of words for topics. The $T$ and $P$ matrices are initially set to 0 and are updated after the $t^{th}$ sampling process by the following rules:

\begin{align*}
\text{decrement} & \quad T^t(d, Z^{t-1}(d, w)), \\
\text{increment} & \quad T^t(d, Z'(d, w)), \\
\text{decrement} & \quad P^t(Z^{t-1}(d, w), v), \\
\text{increment} & \quad P^t(Z'(d, w), v),
\end{align*}

(3.3.1)  
(3.3.2)  
(3.3.3)  
(3.3.4)

where $Z(d, w)$ denotes the $d^{th}$ row and the $w^{th}$ column of $Z$. Once the sampling iterations are finalized, the latent parameters are organized into matrices $\Theta = [\theta_1, \ldots, \theta_M]^\top$ and $\Phi = [\phi_1, \ldots, \phi_K]^\top$, which are computed as:

\begin{align*}
\Theta(d, k) &= \frac{T(d, k) + \alpha_k}{\sum_{i=1}^{K} T(d, i) + \alpha_k} \\
\Phi(k, v) &= \frac{P(k, v) + \beta_v}{\sum_{i=1}^{V} P(k, i) + \beta_v}.
\end{align*}

(3.3.5)  
(3.3.6)

In these equations, $\alpha_k$ and $\beta_v$ are the Dirichlet parameters, where $\alpha_k \in \{\alpha_1, \ldots, \alpha_K\}$, and $\beta_v \in \{\beta_1, \ldots, \beta_V\}$ respectively.
In Chapter 3, a statistical generative model for a hyperspectral image is introduced. The proposed statistical model is described with not only observed variable, but also latent variables. Since a generative model is expressed with a joint probability of variables for the model, the greater the number of variables the greater the tendency to increase the computational complexity. Therefore, assigning a value for a random variable to make a constant value can reduce the computational complexity.

This chapter introduces the way to determine a constant value for random variables, specially, two parameters of the Dirichlet distribution, $\alpha$ and $\beta$. The first section introduces the binomial distribution and the Beta distribution. A multinomial distribution and the Dirichlet distribution are then introduced in its multivariate generalized form. With an explanation for the relationship between the Dirichlet distribution and the multinomial distribution, it is shown how the Dirichlet parameter controls the shape of the multinomial distribution.

The next section discusses how to determine the number of topics, $K$, for the inference model. Although $K$ is not a random variable of the model, it should be fixed to perform the topic inference procedure. For the proper selection of the number of $K$, the perplexity is computed to evaluate the proposed probability model and model parameters.
4.1 Determination of Hyperparameters

In the previous chapter, the proposed model is described with one observed variable and five latent variables; \( w, \alpha, \beta, z, \theta, \) and \( \phi \). The variables, \( \theta \) and \( \phi \), indicate two multinomial distributions: \( \theta \) is the topics distribution for a document and \( \phi \) is the words distribution for a topic. \( \alpha \) and \( \beta \) are parameter vectors of two Dirichlet distributions, \( \text{Dir}(\alpha) \) and \( \text{Dir}(\beta) \), where each Dirichlet distribution is a conjugate prior of the multinomial distribution, \( \theta \) and \( \phi \), respectively.

This section introduces the basic properties of the multinomial distribution and Dirichlet distribution, and then a heuristic method for assigning parameter values of \( \alpha \) and \( \beta \) is described.

4.1.1 Binomial Distribution and Multinomial Distribution

The binomial distribution is a discrete probability distribution for \( N \) successive events, whose values are categorized with two values, such as “yes” or “no.” Suppose that a random variable \( X \in \{v_1, v_2\} \) follows the binomial distribution. There are \( N \) times trials and probability is \( p(X = v_1) = p_1 \). If \( m \) times events show that \( X = v_1 \), then the probability function is represented as:

\[
f(m; N, p_1) = \frac{N!}{m!(N-m)!} p_1^m (1 - p_1)^{N-m}
\]  

(4.1.1)

where \( 1 - p_1 = p_2 = p(X = v_2) \).

The multinomial distribution is a generalized distribution of the binomial distribution by increasing the dimension of variable \( X \) from 2 to \( K \). Thus, \( X \in \{v_1, v_2, ..., v_K\} \) and the random variable \( X \) has \( K \) categories. Suppose the probability of \( p(X) \) is \( p(X = v_1) = p_1, p(X = v_i) = p_i, \) and \( p(X = v_K) = p_K, \) thus \( \sum_{i=1}^{K} p_i = 1 \). Also, note that if \( X = v_1 \) occurs \( m_1 \) times, and \( X = v_i \) occurs \( m_i \) times, similarly, \( X = v_K \)
occurs \( m_K \) times, then \( \sum m_i = N \). The probability function of this multinomial distribution is:

\[
f(m_1, \ldots, m_K; N, p_1, \ldots, p_K) = \frac{N!}{m_1!m_2! \cdots m_K!} p_1^{m_1} \cdots p_K^{m_K}
\]  

(4.1.2)

### 4.1.2 Beta Distribution and Dirichlet Distribution

The Beta distribution describes a behavior of the binary random variable, \( X \in \{v_1, v_2\} \), whose probability \( p_1 \) ranges in \([0, 1]\). In general, the Beta distribution is represented with two parameters, \( \alpha_1 \) and \( \alpha_2 \), and the values of \( \alpha_1 \) and \( \alpha_2 \) control the shape of the Beta distribution. The two parameters of Beta distribution are called hyperparameters of a binomial distribution. The probability density function of the Beta distribution is represented as:

\[
f(p; \alpha_1, \alpha_2) = \frac{1}{B(\alpha_1, \alpha_2)} (p)^{\alpha_1-1}(1-p)^{\alpha_2-1}
\]

(4.1.3)

where \( B(\alpha_1, \alpha_2) \) is Beta function, and the Beta function is described with the Gamma function, \( \Gamma \).

\[
B(\alpha_1, \alpha_2) = \frac{\Gamma(\alpha_1)\Gamma(\alpha_2)}{\Gamma(\alpha_1 + \alpha_2)}
\]

(4.1.4)

The Beta distribution is considered as a *conjugate prior* for a binomial distribution [36]. Conjugate means that the posterior probability is represented with the same parametric function as prior probability. According to the Bayes rule, posterior
probability, \( p(\theta|\mathcal{D}) \) is denoted with likelihood, \( p(\mathcal{D}|\theta) \), and prior, \( p(\theta) \).

\[
p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta)p(\theta)}{p(\mathcal{D})} \quad \text{(4.1.5)}
\]

\[
\approx p(\mathcal{D}|\theta)p(\theta)
\]

where \( \mathcal{D} \) is a given data set, \( \mathcal{D} = (x_1, ..., x_N) \) and \( x_i \in \{v_1, v_2\} \). Thus, the prior probability can be rewritten with a model parameter \( \theta \) based on (4.1.3):

\[
p(\theta) = \theta^{\alpha_1 - 1}(1 - \theta)^{\alpha_2 - 1} \quad \text{(4.1.6)}
\]

The likelihood function for \( N \) samples is also represented as:

\[
p(\mathcal{D}|\theta) = \prod_{n=1}^{N} p(x_i|\theta) = \prod_{n=1}^{N} \theta^{x_n}(1 - \theta)^{1-x_n} = \theta^{N_1}(1 - \theta)^{N_2} \quad \text{(4.1.7)}
\]

where \( N_1 = \sum x_n \) and \( N_2 = \sum (1 - x_n) \). Therefore, the posterior probability can be written as:

\[
p(\theta|\mathcal{D}) \approx p(\mathcal{D}|\theta)p(\theta) \quad \text{(4.1.8)}
\]

\[
\approx [\theta^{N_1}(1 - \theta)^{N_2}][\theta^{\alpha_1 - 1}(1 - \theta)^{\alpha_2 - 1}] \quad \text{(4.1.9)}
\]

\[
= \theta^{N_1 + \alpha_1 - 1}(1 - \theta)^{N_2 + \alpha_2 - 1} \quad \text{(4.1.10)}
\]

The posterior probability shown in (4.1.10) represents with the same parametric function of prior shown in (4.1.6). It shows that Beta distribution is a conjugate prior for binomial distribution.

Figure 4.1 \(^1\) shows several shapes of the probability density function of the Beta distribution.

\(^1\)http://en.wikipedia.org/wiki/Betadistribution
distribution, according to the various values of parameters. In order to understand this figure, take the coin tossing examples. Every coin has two sides, a head and a tail. The first example explains the pink shape distribution. The coin is tossed. There are four tosses - when a head shows two times, $\alpha_1 = 2$ and a tail shows two times, $\alpha_2 = 2$. The probable probability of $p(x = \text{head}) = 0.5$. Thus, the probability density function is highest at $p_{\text{head}} = 0.5$ and $p_{\text{tail}} = 0.5$. The second example is the black shape distribution. Among the seven coin tossing, the head appears two times and the other side appears five times. Then it can be assumed that the probability, $p_{\text{head}}$, may be smaller than $p_{\text{tail}}$. Thus, the probability density function for this Beta distribution is highest at $p_{\text{head}} = 0.2$ and $p_{\text{tail}} = 0.8$. In conclusion, according to the values of $\alpha_1$ and $\alpha_2$, the probabilities of the binomial random variables, $p_{\text{head}}$ and $p_{\text{tail}}$, have uniform or weighted probabilities.

As the multinomial distribution is a generalized form of binomial distribution by
increasing the number of categories of random variables. Thus, the Dirichlet distribution is a generalized form of the Beta distribution for multivariate variables. The Dirichlet distribution is noted as $\text{Dir}(\vec{\alpha})$. In this notation, $\vec{\alpha}$ is a $K$-dimensional parameter vector, $\vec{\alpha} = [\alpha_1, ..., \alpha_K]$ if there are $K$ categories for random variables. Compare the Beta distribution that has two parameters, $\alpha_1$ and $\alpha_2$, with two categories of random variable.

The probability density function of Dirichlet distribution is represented as:

$$f(p_1,...,p_K; \alpha_1,...,\alpha_K) = \frac{1}{\mathcal{B}(\vec{\alpha})} \prod_{i=1}^{K} p_i^{\alpha_i-1}$$

(4.1.11)

where $p_K = 1 - \sum_{i=1}^{K-1} p_i$, and $\mathcal{B}(\vec{\alpha}) = \frac{\prod_{i=1}^{K} \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{K} \alpha_i)}$.

In a special case, a symmetric Dirichlet distribution, the parameter vector $\vec{\alpha}$ has the same elements, $\vec{\alpha} = [\alpha_0, ..., \alpha_0]$, and then $\text{Dir}(\vec{\alpha})$ is represented with a simple form, $\text{Dir}(\alpha)$. In this dissertation, I used a symmetric Dirichlet distribution, so the rest of the notation for the Dirichlet distribution is $\text{Dir}(\alpha)$. The reason why I am using a symmetric Dirichlet distribution for a document is that a topic assigning probability for a document is uniform. In other words, each document may have several topics, but at a corpus level, the selection of a topic for generating a document is the same.

Similar to the Beta distribution, the value of $\alpha$, a parameter of the Dirichlet distribution, controls the shape of the Dirichlet distribution. Also, the Dirichlet distribution is a conjugate prior of a multinomial distribution.

In the proposed model, I use two multinomial distributions: $\theta$ for multiple topics distribution of a document, and $\phi$ for multiple words distribution of each topic. First, the relationship between $\theta$ and $\alpha$ is explained. There are $M$ multinomial distributions for an image having $M$ pixels. Those $M$ multinomial distributions explain how a
Table 4.1: Relationship between variables and its probability distribution

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Two Variables</th>
<th>Multi-variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distribution</td>
<td>Binomial distribution</td>
<td>Multinomial distribution</td>
</tr>
<tr>
<td>Categorical distribution</td>
<td>Beta distribution</td>
<td>Dirichlet distribution</td>
</tr>
<tr>
<td>Notation</td>
<td>Beta($\alpha, \beta$)</td>
<td>Dir($\alpha$)</td>
</tr>
<tr>
<td>Probability density function</td>
<td>$\frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha,\beta)}$</td>
<td>$\frac{1}{B(\tilde{\alpha})} \prod_{i=1}^{K} x_i^{\alpha_i-1}$</td>
</tr>
</tbody>
</table>

where $B(\alpha) = \frac{\prod_{i=1}^{K} \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{K} \alpha_i)}$

corpus shows topics distribution. So, it can be said that the topics distribution of a corpus follows a Dirichlet distribution, since the Dirichlet distribution is a conjugate prior $\theta \sim \text{Dir}(\alpha)$. Similarly, the relationship between $\phi$ and $\beta$ is explained. $\phi$ shows the $V$ categorial distribution, where $V$ is the number of endmembers in a library. So, the words distribution of a corpus also follows the Dirichlet distribution having $V$ categories, $\phi \sim \text{Dir}(\beta)$.

### 4.1.3 Selection of Hyperparamter

In order to demonstrate the proper selection of the hyperparameters, I showed two Beta distribution examples in Figure 4.1. Because the Beta distribution is a binomial case of the Dirichlet distribution, I use the following notation: $B(\alpha_1, \alpha_2) = \text{Dir}(\tilde{\alpha})$, where $\tilde{\alpha} = [\alpha_1, \alpha_2]$. The first example is a pink shape distribution having $\alpha_1 = 2$ and $\alpha_2 = 2$. The second example is a red shape distribution having $\alpha_1 = 0.5$ and $\alpha_2 = 0.5$. Both cases have the same parameter values, thus they are notated as a symmetric Dirichlet distribution, $\text{Dir}_{\text{pink}}(2)$ and $\text{Dir}_{\text{red}}(0.5)$.

The first example shows that if $\alpha$ is larger, then the probability of $p_1 = 0.9$ and
\( p_2 = 0.1 \) occurring is very low. Many cases follow \( p_1 = 0.5 \) and \( p_2 = 0.5 \). On the other hand, the second example is reverted. The chance of \( p_1 = 0.9 \) and \( p_2 = 0.1 \) is much higher than that of \( p_1 = 0.5 \) and \( p_2 = 0.5 \). So, if a binomial distribution follows the red-shaped distribution, \( \theta \sim \text{Dir}_{\text{red}}(0.5) \), then the selection of one topic between two topics is highly expected. Likewise, if a binomial distribution follows the pink-shaped distribution, \( \theta \sim \text{Dir}_{\text{pink}}(2) \), then the selection of two topics is common.

Suppose the above example is a \( K \)-multivariate case. It is said that this is a selection of a topic in \( K \)-categories. If \( \alpha \) is larger than one, \( \alpha > 1 \), then the chance of many of the multiple topics selection is much more common, and their probabilities for \( p_1, \ldots, p_K \) are close to uniform. On the other hand, if \( \alpha \) is smaller than one, \( \alpha < 1 \), then the chance of one or a few topics selection is more probable.

One of the assumptions of the proposed model is that a pixel has a few multiple topics, and a topic has several endmembers. Thus the hyperparamters of the proposed model are chosen in \( 0 < \alpha < 1 \), and \( 0 < \beta < 1 \).

### 4.2 Determination of the Number of Topics

In the previous section, I assumed that the number of topics was known for the estimation of the latent model parameters, \( \theta \) and \( \phi \). The number of topics, however, may not be available, if the given image set does not provide the prior information about the site. In this section, I discuss a perplexity-based approach to properly estimate the number of topics.

#### 4.2.1 Perplexity

Information theory introduced the Shannon entropy or entropy as a measurement of the uncertainty of a system having randomness. For a given data set \( X = \{x_1, \ldots, x_n\} \)
and its probability density function $p(X)$, entropy, $H$, is defined as:

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log_b(p(x_i)),$$  \hfill (4.2.1)

where $b$ is the base of the logarithm. If the uncertainty of random variables increases, then the entropy is high. Otherwise, the entropy is low.

Cross-entropy also defines the measurement of two different probability distributions’ similarity. Indeed, it measures how close a new probability distribution is to the given probability distribution.

$$H(p, q) = -\sum_{i=1}^{n} p(x_i) \log(q(x_i))$$  \hfill (4.2.2)

Based on the concept of cross-entropy, perplexity is defined and is used in the machine learning field to evaluate the soundness of the estimated probability distribution model and its parameters. In other words, a probability distribution is approximated and its model parameters can be estimated using a training data set, then another probability distribution and its model parameter is estimated using a test data set. The cross-entropy between a probability distribution of a training set and another probability distribution of a test set is measured for evaluation how two estimated distribution is close. Low entropy means that the soundness of the estimated distribution of a training set.

Perplexity is defined as:

$$\text{Perplexity} = \exp^{H(p_{\text{test}}, p_{\text{estimate}})} = \exp^{\sum_{i=1}^{n} p_{\text{test}}(x_i) \log(p_{\text{estimate}}(x_i))}$$  \hfill (4.2.3)

The definition of perplexity is interpreted as follows: if there is an unknown probability, $p$, then a model can be estimated from the training samples. Suppose
the probability distribution of an estimated model is $p_{\text{estimate}}$. Then the estimated model $p_{\text{estimate}}$ is evaluated with a set of test samples, $p_{\text{test}}$, which is not used in a set of training samples. So, if an estimated model $p_{\text{estimate}}$ of the unknown distribution $p$ has high probability about new data, $p_{\text{test}}(X)$, then the cross-entropy, $H(p_{\text{test}}, p_{\text{estimate}})$ will be low. Thus, if the model shows low cross-entropy, then the model can be said to be a well-established model.

4.2.2 Perplexity of the Bag-of-Materials Model

The computation of perplexity is applied to the proposed model. Since the probability distribution of the bag-of-materials model is not known, I approximated the probability distribution of the bag-of-materials model and estimated its model parameters. If the number of topics is given as $k$, then two model parameters of corpus are denoted as $\Phi^k$ and $\Theta^k$. The perplexity for the test pixels, $P^k(C_{\text{test}})$, is computed as:

$$P^k(C_{\text{test}}) = \exp \left( \frac{-\sum_{i=1}^{M_{\text{test}}} \log p(d_i | \Phi^k, \theta_i)}{M_{\text{test}}N} \right)$$

(4.2.4)

where $\theta_i$ is estimated as the latent variable from the test corpus, $\theta_i \in \Theta^k_{\text{test}}$. $N$ is the number of materials composing a pixel, which is set to 100 for hyperspectral image analysis. $M_{\text{test}}$ is the number of test pixels in test corpus $C_{\text{test}}$. The probability of a pixel $d_i$ in a test corpus, $P(d_i | \Phi^k, \theta_i)$ can be computed from:

$$p(d_i | \Phi^k, \theta_i) = \prod_{j=1}^{N} p(w_j | z_j, \phi_{z_j}^k)p(z_j | \theta_i).$$

(4.2.5)

Thus, (4.2.4) represents the perplexity of the estimated model, which has $k$ topics. The concept of perplexity is also interpreted as a computation of likelihood, since
perplexity calculates a probability of test samples with a given model distribution. It is represented as:

$$p(\mathcal{X}_{test}|\Theta_{test}, \Phi_{train}) = \exp \left( -\frac{\sum_{i=1}^{n} \log p(d_i|\theta_{test}, \Phi_{train})}{\sum_{i=1}^{n} N_i} \right)$$

(4.2.6)

where $\Theta_{train}$ and $\Phi_{train}$ are model parameters of training samples, and $\mathcal{X}_{test}$ is a test data set, $\mathcal{X}_{test} = \{x_1,...,x_n\}$. $N_i$ is the number of materials of a pixel $d_i$ and $n$ is the number of pixels in test set.

### 4.2.3 Estimated Model Comparison

By changing the number of topics, $k$, many estimated models can be obtained. These estimated models are evaluated with perplexity and compared. If a model shows minimum perplexity, the model and its parameters can be regarded as a well-established model for the proposed bag-of-materials model. So the number of topics can be determined with $k$ having the smallest perplexity value.

---

**Algorithm 4** Determination of the number of topics

1: Randomly select test pixels form the whole corpus, $C_{test} \subset C$, and $C_{test} = \{x_1, \ldots, x_n\}$.

2: Set training sample as $C_{train} = C - C_{test}$.

3: for $k=1$ to $K$ do

4: \[ P^k(C_{test}) = \exp \left( -\frac{\sum_{i=1}^{n} \log p(d_i|\Phi^k, \theta_i)}{100n} \right) \]

5: end for

6: Find $k$ which has minimum $P^k(C_{test})$
This chapter shows the result of the latent topics inference of a pixel. The first section introduces the hyperspectral image used in this experiment and outlines the preprocessing steps for a bag-of-materials model. Following this introduction, I discuss the thematic map generation, which provides high level classification of pixels in terms of the distribution of topics per pixel. At the end, I compare the thematic map-based clustering results with clustering of spectral signatures, as well as clustering of abundances.

5.1 Hyperspectral Data

This experiment analyzes a hyperspectral image of an urban site, which was acquired from the HYDICE sensor. The given image is composed of 307 rows and 307 columns and has 210 spectral bands whose spectrum range from 450 nm to 2500 nm. It is recorded with the 16-bit band interleaved by line (BIL) format. So, the imaging results generate a hypercube.

The 3D form of the hypercube needs to be transformed as a 2D matrix form to generate a pixel-based information vector. The rows of the hypercube image are then concatenated to generate a 2D spectral reflectance matrix, where the dimensions refer...
Figure 5.1: Examples of spectral reflectances of selected regions. Three red rectangle regions have 10 × 10 pixels, and each region represents ground area, rooftop area, and vegetation area, respectively.

to the pixel index, $i$, and the spectral channel index $s$:

$$f : I(x, y, s) \rightarrow R(i, s).$$

Prior to processing, I removed noisy bands 76, 87, 101-110, 138-152, 156, and 204-210 after examining the band images. This reduction resulted in 175 spectral bands
for further processing. So, if the number of imaging layers is 175, then the dimension of the spectral reflectance vector of a pixel is 175. Since the number of rows is 307 and the number of columns is 307, the total number of pixels, $M$, is 94,249. This relation generates an $M \times 175$, whose rows indicates an index of a pixel and whose columns are the *spectral reflectance* of a pixel. Figure 5.1 shows some of the spectral reflectance values of the study site.

### 5.2 Spectral Unmixing

The given hypercube image is converted into a pixel-spectral information based 2D matrix using a matrix transformation. Analyzing the data set is composed of pixel descriptor vectors whose elements come from 175 channels of spectral reflectance. However, the spectral reflectance of a pixel in a remotely sensed hyperspectral image results in the combination of multiple spectra of materials according to their combined weight. As shown in Figure 5.1, the spectral reflectance of pixels shows variation, even though those pixels are assumed to be in the same surface type. Therefore, spectral decomposition is needed to process the bag-of-materials model.

I employed a spectral unmixing algorithm for that purpose. In this experiment, I assumed that the spectra of multiple materials are linearly mixed, thus the unmixing is solved using a least squares estimation method. Also, the name of the contributing materials are known by the spectral library.
5.2.1 Spectral Library

Well-known spectral libraries are published in the U.S. Geological Survey (USGS) and U.S. Army Geospatial Center. The library used in this experiment was obtained from the U.S. Army Geospatial Center.¹

Table 5.1 lists the materials in the given library and Figure 5.2 shows its corresponding spectral signatures.

5.2.2 Non-negative Least Square Estimation

A mixture spectral reflectance, \( s_i \), can be represented as a linear combination of the endmembers of the spectral library, and it is shown with Equation (3.1.2), \( s_i = \sum_{j=1}^{V} a_{ij} w_j \), in Section 3.1.1. Since the spectrum of a pixel, \( s_i = (s_1, s_2, ..., s_L)^T \), is an observed variable and the spectra of endmembers, \( w_i = (w_{1,i}, w_{2,i}, ..., w_{L,i})^T \), are given, the abundance vector of the \( i^{th} \) pixel, \( a_i = [a_{i1}, a_{i2}, ..., a_{iV}] \), is the only unknown variable.

The spectral vector \( w_i \) is arranged as a column of the endmember matrix \( W \).

\[
\begin{align*}
\mathbf{s}_i &= \sum_{j=1}^{V} \mathbf{w}_j \mathbf{a}_i = \mathbf{w} \mathbf{a}_i \\
\begin{bmatrix}
\mathbf{s}_1 \\
\mathbf{s}_2 \\
\vdots \\
\mathbf{s}_N
\end{bmatrix} &=
\begin{bmatrix}
w_{1,1} & w_{1,2} & \ldots & w_{1,V} \\
w_{2,1} & w_{2,2} & \ldots & w_{2,V} \\
\vdots & \vdots & \ddots & \vdots \\
w_{L,1} & w_{L,2} & \ldots & w_{L,V}
\end{bmatrix}
\begin{bmatrix}
a_1 \\
a_2 \\
\vdots \\
a_V
\end{bmatrix}
\end{align*}
\]

(5.2.1)

(5.2.2)

In order to obtain the solution for the abundances of the \( i^{th} \) pixel, I applied the non-negative least squares estimation method, because the combination is assumed to be linear and the abundances of the endmembers can not be a negative value.

¹http://www.agc.army.mil/Hypercube/
<table>
<thead>
<tr>
<th>ID</th>
<th>Material</th>
<th>ID</th>
<th>Material</th>
<th>ID</th>
<th>Material</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Asphalt</td>
<td>18</td>
<td>Gravel 2</td>
<td>34</td>
<td>Rubber, foam</td>
</tr>
<tr>
<td>2</td>
<td>Asphalt, roofing</td>
<td>19</td>
<td>Gravel 3</td>
<td>35</td>
<td>Sand</td>
</tr>
<tr>
<td>3</td>
<td>Asphalt, dark</td>
<td>20</td>
<td>Oak veneer</td>
<td>36</td>
<td>Sand gravel 1</td>
</tr>
<tr>
<td>4</td>
<td>Asphalt, light</td>
<td>21</td>
<td>Pavement</td>
<td>37</td>
<td>Sand gravel 2</td>
</tr>
<tr>
<td>5</td>
<td>Brass</td>
<td>22</td>
<td>Paver brick, gray</td>
<td>38</td>
<td>Sand gravel 3</td>
</tr>
<tr>
<td>6</td>
<td>Chipboard</td>
<td>23</td>
<td>Paver brick, red</td>
<td>39</td>
<td>Soil 1</td>
</tr>
<tr>
<td>7</td>
<td>Concrete</td>
<td>24</td>
<td>Plastic, rust</td>
<td>40</td>
<td>Stucco</td>
</tr>
<tr>
<td>8</td>
<td>Copper</td>
<td>25</td>
<td>Road 1</td>
<td>41</td>
<td>Trees 1</td>
</tr>
<tr>
<td>9</td>
<td>Cotton canvas</td>
<td>26</td>
<td>Road 2</td>
<td>42</td>
<td>Trees 2</td>
</tr>
<tr>
<td>10</td>
<td>Felt tarpaper</td>
<td>27</td>
<td>Road 3</td>
<td>43</td>
<td>Trees 3</td>
</tr>
<tr>
<td>11</td>
<td>Fiberglas, brown</td>
<td>28</td>
<td>Road 4</td>
<td>44</td>
<td>Vegetation 1</td>
</tr>
<tr>
<td>12</td>
<td>Grass 1</td>
<td>29</td>
<td>Road 5</td>
<td>45</td>
<td>Vegetation 2</td>
</tr>
<tr>
<td>13</td>
<td>Grass 2</td>
<td>30</td>
<td>Road, shade</td>
<td>46</td>
<td>Vegetation 3</td>
</tr>
<tr>
<td>14</td>
<td>Grass 3</td>
<td>31</td>
<td>Roof 1</td>
<td>47</td>
<td>Water 1</td>
</tr>
<tr>
<td>15</td>
<td>Grass, mowed</td>
<td>32</td>
<td>Roof 2</td>
<td>48</td>
<td>Water 2</td>
</tr>
<tr>
<td>16</td>
<td>Grass, sparse</td>
<td>33</td>
<td>Rubber, black</td>
<td>49</td>
<td>Water 3</td>
</tr>
<tr>
<td>17</td>
<td>Gravel 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure 5.2: Spectral signatures of 49 endmembers in a spectral library provided publicly by the U.S. Army Geospatial Center.
Also, the abundances indicate the weight ratio of mixture endmembers, and the sum of the abundance should be close to 1, $\sum_{j=1}^{V} a_{j,i} \approx 1$.

After estimating the abundances, each pixel is described with the abundances of the endmembers. In this experiment, the number of given endmembers is 49, thus the pixel-abundance matrix is represented with a $M \times 49$ matrix. This matrix form is a base format of a bag-of-materials model because the proposed model must count which materials are present and how frequently they appear. In order to make abundance be an integer number, I multiplied the abundance by 100, or the ratio of weight. The acquired frequencies are considered to be the number of frequencies of the endmembers for a pixel.

### 5.2.3 Abundance Map

Through the spectral unmixing process, each pixel is described with the ratio of 49 materials. By transforming this 2D matrix into 3D multiple layered images, 49 abundance maps are obtained:

$$f_{\text{Abundance}} : \mathbf{m}_{\text{Abundance}}(i, a) \rightarrow \mathbf{I}_{\text{Abundance}}(x, y, a).$$

Figure 5.5 shows that the degree of the material for pixels, so that the white region means those pixels have high values of the materials. On the other hand, the black regions show that the degree of that material is very low.

In the following figures, some materials such as Asphalt have relatively low ratio through out the whole image, and the imaging layer shows as dark. It cannot be discerned in this figure. However, the third material, dark Asphalt, is shown in pixels that belong to the road and parking lot.

Also, the 15th material, named as mowed grass, and the 16th material, which
is sparse grass, show as separated regions, however, the union of those two regions would cover much ground area as a short grass region. The 44th layer shows a lower abundance than those two materials, 15th and 16th, however, it shows that many pixels are overlapped in that region. So the pixels appeared commonly in those three materials could be considered as a type of vegetation area.

5.3 Model Variables and Determinations of Values

The proposed model, a bag-of-materials model, is explained with a probability density function of a pixel as Equation (3.2.8), where several parameters are used to describe the model.

\[
p(d | \alpha, \beta) = \int p(\theta_i | \alpha) \prod_{j=1}^{N} p(z_{ij} | \theta_i) p(w_{ij} | z_{ij}, \phi_{z_{ij}}) p(\phi_{z_{ij}} | \beta) d\theta_i
\]

Among the variables, I mentioned that the model parameters are only \( \theta \) and \( \phi \), so those variables are inferred using the LDA process. Other variables should be given prior to inferring those model parameters.

This section provides the threshold for the determination of values for the model variables. First, the hyperparameters \( \alpha \) and \( \beta \) are determined by the shape of multinomial distribution explaining the pixel generation assumptions. Second, the number of topics, \( K \), is determined after examination of the model evaluation.

5.3.1 Dirichlet Parameters

The proposed model assumes that there are multiple topics in a pixel and composing materials for that pixel are chosen according to those topics. The shape of a multinomial distribution, whether it shows uniform distribution or uneven distribution,
Figure 5.3: Abundance maps according to material
Figure 5.4: Abundance maps according to material (Cont.)
Figure 5.5: Abundance maps according to material (Cont.)
is affected by the Dirichlet distribution. Thus, the parameter of Dirichlet distribution, called a hyperparameter of the model, is chosen after considering the given assumption. In this model, there are two multinomial distributions, \( \theta \) and \( \phi \). Hyperparameter \( \alpha \) controls the multinomial distribution \( \theta \), and hyperparameter \( \beta \) controls the multinomial distribution \( \phi \). Thus, the values of hyperparameters are chosen after considering the given assumption. Particularly, if a hyperparameter is larger than one, then the probability of a multinomial distribution becomes uniform, so the value of hyperparameters is less than one.

In this experiment, I set up \( \alpha = 0.5 \) and \( \beta = 0.1 \) based on Figure 5.6 and Figure 5.7. Figure 5.6 shows an example of multinomial topics distribution when the number of topics is six, \( K = 6 \). When \( \alpha \) is 0.5 or 1, a multinomial distribution has a few major multiple topics, and that is suitable in the given assumption. On the other hand, when \( \alpha \) is 0.1 or 0.01, a multinomial distribution has one major topic. Also, when \( \alpha \) is larger than 1, a multinomial distribution has many topics that shows a uniform distribution. It is not suitable for the given assumption of the model. Based on the assumption that each pixel has multiple topics, however, one or two topics would play the role of major topics. Figure 5.6(a) shows that a multinomial distribution is not evenly distributed when \( \alpha \) is 0.01. It has only one high probability for one category, thus a pixel may have only one topic and it is not suitable for the previous assumption that a pixel have multiple topics. Figure 5.6(c) and (d) show that a pixel may have multiple topics. The multinomial distributions were generated when \( \alpha = 0.5 \) and \( \alpha = 0.1 \), respectively. If \( \alpha \geq 1 \), then a multinomial distribution of topics would be close to a uniform distribution. It is also not suitable to the assumption.

Similarly, Figure 5.7 shows the multinomial distribution of words according to the hyperparameter \( \beta \) of Dirichlet distribution. When \( \beta \) is 0.01 or \( \beta \) is 0.1, each topic may contain several words that have similar meaning.
Figure 5.6: Multinomial distribution according to the value of the hyperparameter, $\alpha$. (a) $\alpha = 0.01$, (b) $\alpha = 0.1$, (c) $\alpha = 0.5$, (d) $\alpha = 1$, (e) $\alpha = 5$, (f) $\alpha = 10$. 
Figure 5.7: Multinomial distribution according to the value of the hyperparameter, $\beta$. (a) $\beta = 0.01$, (b) $\beta = 0.1$, (c) $\beta = 0.5$, (d) $\beta = 1$, (e) $\beta = 5$, (f) $\beta = 10$. 
5.3.2 Topic Dimension

Figure 5.8: Perplexity (solid line) and the perplexity ratio (dotted line) plotted as functions of topic count \((K)\). Latent parameters, \(\phi\) and \(\theta\), are generated based on the smallest topic count \((K = 6, \text{in this example})\) which is higher than the 95% line (denoted by the thin, dotted line).

In this implementation, I repeatedly generated document models for a number of topics ranging from \(1 \leq K \leq 49\) and computed the model parameters \(\theta_i\) for each document and \(\Phi^K\) for the training corpus. The choice of 49 as the upper bound for \(K\) is attributed to the 49 endmembers in our library. In order to select the “smallest” number of topics expressing the characteristics of a pixel, I examined the ratio of change in perplexity starting from a single topic and going to multiple topics, as shown in Figure 5.8. The ratio of change is computed as the ratio between two consecutive perplexities: \(\mathcal{P}^{K+1}/\mathcal{P}^K\). If this ratio is small, then the estimated model for \(K\) topics is conjectured not to model the corpus well enough. On the other hand, if the ratio is close to 1, \(K\) is assumed to model the corpus as well as \(K + 1\) topics and provides the solution to our problem. With this insight, I selected the number
of topics as the smallest $K$ that is above the 95th percentile of the ratio, as shown in Figure 5.8. For the experiment discussed in the next section, the plot shows that $K = 6$ topics for the hyperspectral image classification problem.

### 5.4 Inference of Latent Topics for Each Pixel

Considering the low spatial resolution of an aerial or satellite hyperspectral image, each pixel is described with multiple materials with abundances estimated using the spectral unmixing technique discussed in Section 5.2. In order to perform unmixing, I used the spectral library with 49 endmembers ranging from 450 $nm$ to 2500 $nm$, which is provided publicly by the U.S. Army Geospatial Center. In Figure 5.2, I tabulated the spectral signatures and the names of each endmember in the library. This library will later be used as the dictionary of words, $\mathcal{D}$, for the proposed model.

To discover the latent topics of the proposed model, I generated a bag-of-materials, $d_i$, for each pixel $i$ by considering the abundances, $a_i$, representing the linear combination of the endmembers. Once the bags-of-materials are generated, the number of topics $K = 6$ is chosen based on the perplexity shown in Figure 5.8.

In order to estimate model parameters, the probability density function of the model should first be known. However, the given data set, $M_{Abundance}$, does not provide the probability density function, so, the probability distribution of given data set should be approximated. Then, the model parameters are estimated. In this study, the probability density of the given data set is estimated with the Monte Carlo approach, because direct computation of joint probability of the model is intractable. The variable $z$ is dependent on both a topic of a pixel and a topic of a material. In other words, the variable $z$ is related to both $\theta$ and $\phi$.

The Monte Carlo approach suggests data sampling and then estimated the distribution with the samplings. Specifically, I adapted the collapsed Gibbs sampling to
handle variable z in both parameters. First, the endmembers-topic distribution Φ as a $K \times V$ ($6 \times 49$) matrix is set to perform for the Gibbs samplings. In this experiment, 1000 times samplings are performed. Six distributions of topics are organized as rows in Φ and are plotted in Figure 1.2. In addition, I tabulate the endmembers that have the maximum percentiles within each topic in Table 5.2. In other words, the table lists the major materials contained in each topic. The first topic includes ground surface materials such as grass and soil. The second topic is composed of asphalt road and parking lot area. The third topic also contains road segments, but the major materials are different from asphalt. The fourth topic represents low-rise trees and vegetations. The fifth topic contains roof materials, including metals and rubber. Finally, the sixth topic contains tall trees. Another interpretation of the Φ matrix is to plot how the topics are distributed with respect to the endmembers. The 49 distributions coming from the columns of Φ matrix are plotted as bar plots in Figure 5.9. In Figure 5.9, the plots with uniform distribution indicate that none of the topics contain that material.

Similar to estimating Φ, I construct the pixel-topic distribution Θ as an $M \times K$ ($(307 \times 307) \times 6$) matrix by 1000 sampling. In this matrix, each column can be considered an image representing topic. The resulting six layers are shown in Figure 5.10. In this figure, (a) shows the likelihood of each pixel in the image belonging to the first topic, (b) for the second topic, and so on.

Each layer is reconstructed from column vectors in a matrix of pixel-topic distribution, Θ. The gradient values of each layer indicate the probability of how much the pixel belongs to the topic. A pixel having a white area, in the $i^{th}$ layer, refers to that region having a high probability of belonging to the $i^{th}$ topic, while the black region indicates those pixels that have a low probability of belonging to the topic; with (a)
1st topic layer, (b) 2nd topic layer, (c) 3rd topic layer, (d) 4th topic layer, (e) 5th topic layer, and (f) 6th topic layer.

Figure 5.11 to 5.16 show how topic distributions are generated differently according to the value of the hyperparameter of the Dirichlet distribution. The images are obtained from the pixel-topic distribution matrices, Θs, which are estimated under various conditions: With the fixed $\beta = 0.1$, I changed the value of $\alpha$ as 0.01, 0.1, 0.5, 1, 5, and 10, respectively. When $\alpha$ is less than 1, each topic shows a higher probability than that of $\alpha = 5$ and $\alpha = 10$. The higher probability of the topic in a pixel is represented with white, and the lower probability of the topic in a pixel is shown as dark. The reason for lower probability of topics when $\alpha = 5$ and $\alpha = 10$ is that multinomial topics are uniformly distributed, as shown in Figure 5.6.

On the other hand, Figures 5.17 to Figures 5.22 show the various topics distributions according to the various values of $\beta$, where $\beta = 0.01$, $\beta = 0.1$, $\beta = 0.5$, $\beta = 1$, $\beta = 5$, and $\beta = 10$. For all those cases, $\alpha$ is fixed with 0.5. The variations of $\beta$ do not have much effect on the topics distribution than those of $\alpha$, since the value of $\beta$ is related with the multinomial words distributions.
Table 5.2: Topics and representative end-members. The latent parameters $\Phi$ provide the material distributions for each topic. The parameter $\Phi$ is estimated from $\phi \sim \text{Dir}(0.1)$

<table>
<thead>
<tr>
<th>Material</th>
<th>1st topic</th>
<th>%</th>
<th>2nd topic</th>
<th>%</th>
<th>3rd topic</th>
<th>%</th>
<th>4th topic</th>
<th>%</th>
<th>5th topic</th>
<th>%</th>
<th>6th topic</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grass, sparse</td>
<td>47</td>
<td></td>
<td>Asphalt, dark</td>
<td>52</td>
<td></td>
<td>Road 5</td>
<td>39</td>
<td></td>
<td>Tree 3</td>
<td>44</td>
<td></td>
<td>Rubber</td>
</tr>
<tr>
<td>Soil</td>
<td>18</td>
<td></td>
<td>Felt Tarpaper</td>
<td>13</td>
<td></td>
<td>Felt, tarpaper</td>
<td>9</td>
<td></td>
<td>Grass, mowed</td>
<td>36</td>
<td></td>
<td>Water 1</td>
</tr>
<tr>
<td>Grass 1</td>
<td>10</td>
<td></td>
<td>Road 5</td>
<td>11</td>
<td></td>
<td>Roof 1</td>
<td>9</td>
<td></td>
<td>Vegetation</td>
<td>10</td>
<td></td>
<td>Cropper</td>
</tr>
<tr>
<td>Felt, tarpaper</td>
<td>9</td>
<td></td>
<td>Road Shade</td>
<td>9</td>
<td></td>
<td>Road 3</td>
<td>8</td>
<td></td>
<td>Grass 3</td>
<td>4</td>
<td></td>
<td>Roof 2</td>
</tr>
<tr>
<td>Vegetation</td>
<td>5</td>
<td></td>
<td>Road 1</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Brass</td>
<td>4</td>
</tr>
</tbody>
</table>
Figure 5.9: Topic distribution of materials in the spectral library. Each column in the latent parameter matrix, $\Phi$, provides the allocation of topics for each respective material. Compared to Figure 5.2, this plot provides information about which materials can be grouped into similar categories. For instance, copper and rubber are different materials and they have different spectral signatures. These materials, however, can be grouped into the same category because they are both used to build roofs.
Figure 5.10: Each layer is reconstructed from column vectors in a matrix of topic-document distribution, $\Theta$. The model parameter $\Theta$ is estimated from $\theta \sim \text{Dir}(0.5)$. (a) 1$^{st}$ topic layer, (b) 2$^{nd}$ topic layer, (c) 3$^{rd}$ topic layer, (d) 4$^{th}$ topic layer, (e) 5$^{th}$ topic layer, (f) 6$^{th}$ topic layer.
Figure 5.11: Comparison of the first topic distribution according to various $\alpha$ when $\beta$ is 0.1.
Figure 5.12: Comparison of the second topic distribution according to various $\alpha$ when $\beta$ is 0.1.
Figure 5.13: Comparison of the third topic distribution according to various $\alpha$ when $\beta$ is 0.1.
Figure 5.14: Comparison of the fourth topic distribution according to various $\alpha$ when $\beta$ is 0.1.
Figure 5.15: Comparison of the fifth topic distribution according to various $\alpha$ when $\beta$ is 0.1.
Figure 5.16: Comparison of the sixth topic distribution according to various $\alpha$ when $\beta$ is 0.1.
Figure 5.17: Comparison of the first topic distribution according to various $\beta$ when $\alpha$ is 0.5.
Figure 5.18: Comparison of the second topic distribution according to various $\beta$ when $\alpha$ is 0.5.
Figure 5.19: Comparison of the third topic distribution according to various $\beta$ when $\alpha$ is 0.5.
Figure 5.20: Comparison of the fourth topic distribution according to various $\beta$ when $\alpha$ is 0.5.
Figure 5.21: Comparison of the fifth topic distribution according to various $\beta$ when $\alpha$ is 0.5.
Figure 5.22: Comparison of the sixth topic distribution according to various $\beta$ when $\alpha$ is 0.5.
CHAPTER 6
LAND COVER MAP GENERATION USING
UNSUPERVISED CLASSIFICATION

Generating a land cover map using a remotely sensed image is important when providing information about the Earth’s surface. For that reason, many of the image classification algorithms published in the computer vision field are applied to a remote sensing image. Generally, classification methods are divided into two groups, supervised classification and unsupervised classification. In order to perform a classification task using supervised algorithms, a set of training samples that are composed of an observed data and truth class label should be provided. However, these methods are not always available when the truth data set is limited. Specifically, an unexplored area, where little information is known about the surface type, may not provide a set of training samples. Therefore, if training sample sets are limited, then an unsupervised clustering algorithm is more useful.

This section shows unsupervised clustering results of a hyperspectral image. For the experiment, I prepared three types of pixel descriptive vectors. The first is based on the spectral information of a pixel; the second descriptor is an abundance vector of a pixel; and the third type is a topic distribution of a pixel. For each pixel descriptor, I applied two well known clustering methods: K-means clustering and Meanshift clustering. This experiment shows that the results of unsupervised clustering with
inferred topic vectors have cohesive clustering result. The coefficients of clusters evaluates the performance of the results.

6.1 Three types of Pixel Descriptors

In the framework introduced, the information contained for each pixel can be expressed by three different descriptors. The first of these is the raw spectral reflectance vector $s_i$ acquired from the hyperspectral imaging system. For the study site, the spectral descriptor is an $L = 175$ dimensional vector. The second type of descriptor is obtained from the spectral unmixing. The form of this descriptor is an abundance vector of mixed materials $a_i$. Since I used the spectral library provided by the U.S. Army Geospatial Center, the dimensionality of the abundance descriptor is 49, which is the same as the number of endmembers. Finally, the topic distribution of a pixel, $\theta_i$, estimated by the bag-of-materials model, can be used to describe the pixel. The dimensionality of the descriptor in this case reduces to six. Figure 6.1 shows three types of descriptors for a selected pixel.

6.2 Unsupervised Clustering

In order to generate segmented partitions whose pixels have similar properties, clustering methods are performed in an image. Unlike classification which assigns a label to data, clustering methods assign the types of cluster and each cluster is represented with a centroid. Among many clustering methods, this experiment performed two clusterings. One is the parametric clustering method, K-means, and the other one is non-parametric clustering, or Meanshift clustering. The rest of this section briefly addresses each clustering method and shows the results of clusterings for three types of pixel descriptors.
Figure 6.1: Three types of pixel descriptors
Algorithm 5 K-means clustering

1: Initialize $K$ cluster centers
2: while termination condition is not satisfied do
3: Assign instance to the closest cluster center
4: Update cluster centers based on the assignment
5: end while

6.2.1 K-means clustering

K-means clustering is one of the simplest unsupervised learning algorithms. In order to perform this algorithm, the number of clusters, $K$, should be determined first, so this is a parametric clustering. Also, to find $K$ segmented pixel groups having similar properties, this clustering method computes the Euclidian distance between two vectors. So the sum of the squared distances among the pixels in the same cluster are minimized.

Algorithm 5 shows the procedure of the K-means algorithm. First, each cluster center is initially set with randomness. Then every pixel is assigned to one of the clusters whose distance is smallest after computing the distance between the pixel and centers. Next, cluster centers are updated by finding the minimized sum of the squared distance among the pixels in the same cluster. The second and third steps are repeated until the termination condition is satisfied.

Given a set of observations $X = (x_1, ..., x_n)$, where each observation is a $d$–dimensional vector, then the K-means clustering aims to partition the $n$ observations into $K$ sets, such as $S = \{S_1, ..., S_K\}$, so as to minimize the within-cluster sum of squares.

$$\arg \min_{s} \sum_{i=1}^{K} \sum_{x_j \in S_i} ||x_j - \mu_i||$$

(6.2.1)

where $\mu_i$ is the mean of points in $S_i$. 

100
Results of K-means Clustering

Using any one of these three descriptors, the thematic map representing the hyperspectral image can be generated by clustering the pixels. For the sake of simplicity, I tested the $K$–means clustering approach for $K = 6$ to perform this task. The clustering results for the three different descriptors are shown in Figure 6.2 and Figure 6.4. The clusters for each descriptor are reordered based on appearance similarity, such that the first cluster in parts (a), (b) and (c) are conjectured to be matching. Considering that the topic distribution per material tabulated in Figure 5.9, the clusters of the first row for each descriptor in Figure 6.2 can be explained as the soil area. Similarly, the clusters in the second row come from the left delineated paved areas, such as parking lots and road. It is unclear what the clusters of the third row in Figure 6.2 represent. In the case of (a), the segmented regions belong to urban structural material, such as a road. However in the case of (b) in the third clusters, some of the pixels are related to vegetation materials. (c) shows part of some urban structural materials, and a few pixels show a residential area that is unlike (a). The fourth clusters from (a),(b) and (c), can be conjectured to correspond to grass and vegetation areas. The fifth clusters of (a) and (b) show the rooftop area, while the cluster of (c) is mixed, showing a road segment and rooftop. The sixth cluster, which includes all three, seems to indicate a tree and vegetation area.
Soil and ground

Asphalt area, parking lot

Road

Figure 6.2: Clustering the study site hyperspectral image using the $K$-means approach and using three different descriptors for $K = 6$ clusters. Clusters for (a) topic distribution-based descriptor; (b) abundance-based descriptor; and, (c) spectral reflectance-based descriptor. The clusters are ordered manually based on appearance similarity.
Figure 6.3: Clustering Results
6.2.2 Meanshift Clustering

Meanshift clustering is another popular clustering algorithm used in computer vision fields [25]. Unlike $K$-means clustering, the Mean Shift clustering algorithm does not require the number of clusters, because it is a nonparametric clustering. That is based on kernel density function [25]. Using the kernel density estimator, the algorithm seeks modes of density of data points. Finding modes is performed iteratively by updating means, and it stops when the threshold converges.

Given a set of observations $X = (x_1, ..., x_n)$ of $d$-dimensional vector, the kernel density estimator is defined as:

$$f(x) = \frac{1}{n\sigma^d} \sum_{i=1}^{n} K\left( \frac{x - x_i}{\sigma} \right)$$ (6.2.2)

where $K$ is a given kernel as $K(x) = c_{k,d}k(\|x\|^2)$, and $\sigma$ is a given bandwidth. The modes of the density function are located at the zeros of the gradient function $\nabla f(x) = 0$.

$$\nabla f(x) = \frac{2c_{k,d}}{n\sigma^{d+2}} \sum_{i=1}^{n} (x_i - x)g\left( \frac{\|x - x_i\|^2}{\sigma} \right)$$ (6.2.3)

$$= \frac{2c_{k,d}}{n\sigma^{d+2}} \left[ \sum_{i=1}^{n} g\left( \frac{\|x - x_i\|^2}{\sigma} \right) \right] \left[ \frac{\sum_{i=1}^{n} x_ig\left( \frac{\|x - x_i\|^2}{\sigma} \right)}{\sum_{i=1}^{n} g\left( \frac{\|x - x_i\|^2}{\sigma} \right)} - x \right]$$ (6.2.4)

where $g(s) = -k'(s)$. The first term is proportional to the density estimate at $x$ computed with kernel $G(x) = c_{g,d}g(\|x\|^2)$ and the second term is the mean shift [25].

$$m_{\sigma}(x) = \frac{\sum_{i=1}^{n} x_ig\left( \frac{\|x - x_i\|^2}{\sigma} \right)}{\sum_{i=1}^{n} g\left( \frac{\|x - x_i\|^2}{\sigma} \right)} - x$$ (6.2.5)
Algorithm 6 Meanshift clustering

1: for \( i = 1 \) to \( n \) do
2: \hspace{1em} while update is not converged do
3: \hspace{2em} Each data point, \( x_i \) calculate the gradient by \( \nabla p(x) \)
4: \hspace{2em} compute the mean shift vector, \( m_\sigma(x^t) \)
5: \hspace{2em} translate the window as \( x^{t+1} = x^t + m_\sigma(x^t) \)
6: \hspace{1em} end while
7: end for

The mean shift vector 6.2.5 moves to the direction of the maximum increase in the density.
Figure 6.4: Meanshift clustering results of topic distribution
6.3 Quantitative Measurement

This section introduces two quantitative measurement methods for comparison of the clustering results. One method is the silhouette coefficient and the other method is the kappa coefficient. The silhouette coefficient called measures the cohesion of the clusters, so the coefficient indicates how degree of the dispersion. Next, kappa coefficient measures the degree of accordance between two data set. If the ground truth data set is provided then accuracy and precision can be measured. However, the give data set does not provide the ground truth set data. It provides the some marks of the land types [106]. Therefore, the kappa coefficient can be used for agreement between probable ground truth data set and experimental results. The rest of this section shows the comparison results of those two measurement methods.

6.3.1 Cluster Cohesion Coefficient

Qualitative comparisons of the clustering results for the three types of pixel descriptors can be made based on segmented images. Figure 6.2 shows the K-mean clustering results and Figure 6.4 shows the Mean Shift clustering results.

For instance, in Figure 6.2, the paved (asphalt) regions in the image, which includes the parking lot and roads, are better delineated in (a). As for roofs of buildings, the clustering result of topic descriptor vectors appears to provide most pixels correctly by visual inspection. I should, however, note that the ground truth for the study site is not available; hence, a direct comparison against a known truth is not possible. In the following discussion, I suggest a quantitative comparison between the three clustering results.

In order to quantitatively evaluate the clustering results shown in Figure 6.2, I compute the cluster cohesion coefficient, $CCC$, for each cluster [108]. Let the pixels
inside cluster $i$ be defined by neighborhood $\mathcal{N}_i$, such that the image $I = \bigcup_{i=1}^{k} \mathcal{N}_i$. The number of pixels inside a cluster is denoted by $N_i$ where $M = \sum_{i=1}^{k} N_i$ is the total number of pixels. Using these notations, the cluster cohesion coefficient, which measures the maximum distance between the descriptors within a cluster and minimum distance between the clusters, can be computed by:

$$CCC_i = \frac{1}{N_i} \sum_{j \in \mathcal{N}_i} \frac{B_j - A_j}{\max(A_j, B_j)},$$  

(6.3.1)

In this equation, $A$ and $B$ can be written for each descriptor type in similar fashion. To save space, I show these parameters for the spectral signature-based descriptor:

$$A_j = \max_{k \in \mathcal{N}_i} ||s_k - s_j||_2,$$  

(6.3.2)

$$B_j = \min_{k \in I, k \notin \mathcal{N}_j} ||s_k - s_j||_2,$$  

(6.3.3)

where $A_j$ is the maximum distance between the descriptor $j$ and all other descriptors in the same cluster, and $B_j$ is the minimal distance between the descriptor $j$ and all descriptors that belong to the other clusters. However, the computations of all distances are expensive, so I modified the computation as follows. Instead of using the average distance between vectors in the same cluster, $\text{average}(\text{dist}(x_i, x_j))$, I computed the distance between the center of the same cluster, $c_i$ and a vector for $A_i$. Also, distance $B_i$ is computed with a vector and centers of the other clusters.

**Comparison of Cluster Cohesion Coefficient of Pixel Descriptors**

The cluster coherence coefficient typically ranges from $-1$ to $1$, where $-1$ indicates low coherence; hence, a poor model. This observation is due to distance $A_j$ being much larger than the distance $B_j$. In the case of negative $CCC$, the intra-cluster distance is much larger than the inter-cluster distances. On the other hand, a $CCC$ value close
Figure 6.5: Comparison of Cluster Cohesion Coefficients for three different descriptors. The black, gray and white bars, respectively, indicate the $CCC$ values computed from the topic distribution-based descriptor (Figure 6.2a), the abundance-based descriptor (Figure 6.2b) and spectral reflectance-based descriptor (Figure 6.2c).
to 1 indicates a good clustering. In Figure 6.6, I show the comparison between the CCC of the clustering results for three different descriptors for each cluster. It can be observed from the plot that the CCC for a cluster is more coherent and better when I used the topic distribution $\theta_i$ as the pixel descriptor. On the other hand, the CCC values computed for the spectral signature based descriptor showed poor clustering performance.

**Comparison of K-means clustering and Meanshift clustering**

Figure 6.6 shows the comparison of CCCs. In multi-dimensional vector space, the results of K-means clustering show higher cluster cohesions than the mean shift cluster cohesions.
Table 6.1: Example of two observations

<table>
<thead>
<tr>
<th>Yes in 2\textsuperscript{nd} observation</th>
<th>No in 1\textsuperscript{st} observation</th>
<th>total (2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>$a_1$</td>
<td>$b_1$</td>
</tr>
<tr>
<td>No</td>
<td>$a_2$</td>
<td>$b_2$</td>
</tr>
<tr>
<td>total (1)</td>
<td>$a_1 + a_2 = m_1$</td>
<td>$b_1 + b_2 = m_2$</td>
</tr>
</tbody>
</table>

\[6.3.2 \text{ Kappa Coefficient}\]

As an assessment tool of the quality of the clustering results, the kappa coefficient is introduced [24]. Instead of computing the accuracy and precision of the results, which are calculated with a ground truth data set, the Cohen’s kappa coefficient accesses the degree of agreements of the clustering results without ground truth. [50] [127]

Since the ground truth data set for all the pixels of the experiment image is not provided, I referred the color composite image having notations and shown in [106] and referred the ground truth abundance map for the experiment image shown in [103]. By applying threshold into the abundance maps, I generated the black masked image shown Figure 6.7(a). The Cohen’s kappa coefficient calculates the agreement between the generated mask image in Figure 6.7(a) and the clustering results, Figure 6.7(b), Figure 6.7(c), or Figure 6.7(d).

The Cohen’s kappa coefficient is calculated as following:

\[
\kappa = \frac{P_a - P_e}{1 - P_e} \tag{6.3.4}
\]

where, $P_a$ is the portion of observed agreement and $P_e$ is the expected agreement; $P_a = \frac{a_1 + b_2}{T}$ and $P_e = \frac{m_1}{T} * \frac{n_1}{T} + \frac{m_2}{T} * \frac{n_2}{T}$.

Table 6.2 shows the coefficient between masked ground truth map and clustering results of each descriptor. The kappa coefficients between the ground truth map and the clustering results of topic distribution-based descriptor show higher coefficient
Table 6.2: Cohen’s kappa coefficient

<table>
<thead>
<tr>
<th></th>
<th>(a) topic distribution</th>
<th>(b) abundance</th>
<th>(c) spectral reflectance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mowed Grass and Soil</td>
<td>0.7302</td>
<td>0.4015</td>
<td>0.1279</td>
</tr>
<tr>
<td>Asphalt</td>
<td>0.6197</td>
<td>0.5335</td>
<td>0.4912</td>
</tr>
<tr>
<td>Road</td>
<td>0.4970</td>
<td>0.3841</td>
<td>0.3805</td>
</tr>
<tr>
<td>Grass</td>
<td>0.3180</td>
<td>-0.0702</td>
<td>-0.1121</td>
</tr>
<tr>
<td>Roof</td>
<td>0.4048</td>
<td>0.3036</td>
<td>0.1331</td>
</tr>
<tr>
<td>Tree</td>
<td>0.4739</td>
<td>0.4955</td>
<td>0.0698</td>
</tr>
</tbody>
</table>

than the clustering results of others; abundance map or spectral reflectance descriptors. Thus, Table 6.2 implies that the clustering results of topic distribution-based descriptor provides well segmented region for generating a thematic map.
Figure 6.7: Comparison of clustering results with color masked ground truth. (a) masked regions that introduced as a ground truth abundance map in [103]. (b) clustering results of the topic distribution-based descriptor. (c) clustering results of the abundance-based descriptor. (d) clustering results of the spectral reflectance-based descriptor.
Figure 6.8: Comparison of clustering results with color masked ground truth. (a) masked regions that introduced as a ground truth abundance map in [103]. (b) clustering results of the topic distribution-based descriptor. (c) clustering results of the abundance-based descriptor. (d) clustering results of the spectral reflectance-based descriptor.
CHAPTER 7
CONCLUSION AND FUTURE WORK

7.1 Summary

In this dissertation, I introduced a new method for generating a thematic representation of a pixel for hyperspectral images. The method first decomposes the spectra of mixed spectral signatures using an unmixing algorithm, then estimates the latent topics of composing materials. The latent topics are inferred from the combinations of co-occurring endmembers provided with a spectral library. After latent topics are estimated, each pixel is described according to topic distributions.

The proposed representation has two advantages. First, it significantly reduces the dimensionality of pixel vectors for description. Second, the distribution of topics per pixel embodies the semantics implied by co-occurring materials. Therefore, the proposed method of using the topic distribution vector has great potential for effectively handling low spatial resolution and high spectral resolution problems in hyperspectral images.

In order to show the effectiveness of the proposed representation, I apply unsupervised learning algorithms to segment the image. For comparison, three types of pixel description vectors are used: topics distribution, abundances distribution, and spectral reflectance information. Each cluster in this result implies the semantic class of pixel to generate a thematic map without using supervised classification and it...
corresponds to each topic. For quantitative evaluation of the segmentation result, I computed the coherency of clusters measured by the cluster cohesion coefficient, and compared the clustering results using three types of pixel descriptors. The clusters derived from the proposed representation exhibited dense cohesion than the other two descriptors — abundance descriptors and spectral reflectance descriptors.

I conclude that discovering latent topics of pixels is an effective, low-dimensional way to represent material characteristics of spectral data and can be used to generate high quality thematic maps.

## 7.2 Limitations

The proposed method has some limitations due to assumptions and due to the properties of hyperspectral images. Two assumptions used in this study are: The first assumption is that a spectral library is known, so the spectral signature of materials that appeared in the images should be known. If unknown materials exist in a site being explored, then the abundances of materials may include errors. The second assumption is that a spectrum of a pixel is linearly combined from mixture materials. Thus, a linear spectral unmixing method is used to obtain the ratio of the combining materials. However, in nature, mixture spectra are generated with a non-linear combination. It is not known what effect this changed condition may have on a generation of abundance vectors.

Another limitation is that the proposed method only works in hyperspectral images. The reason is derived from the characteristics of hyperspectral sensors. Due to its better performance when detecting the chemical and spectral properties of materials using the hyperspectral image, spectral unmixing algorithm can be applied. In other words, each material has its own spectral signature, and it helps to identify from other materials. However, other types of images, such as multispectral images
and RGB-color channel based images, have much less spectral information; thus, decomposition of materials is very difficult in these image domains.

### 7.3 Future Work

As future work, I would like to suggest the following:

- model development using spatial locality; and
- model development for dynamic topics.

First, the proposed model adapts a statistical generative model, which explains how the observations are produced. The generative model focuses on how a mixture spectra of a pixel is generated from materials involved and latent themes are inferred. In this phase, I do not consider its neighboring pixels. However, unlike a document corpus, an image has much more correlation among neighboring pixels. The latent topics of neighboring pixels can be considered in their generation process. In statistics, the Markov Random Field (MRF) model provides an undirected graphical relationship between neighbors. I would like to suggest combining the MRF model and a bag-of-materials model at the stage of generating topics for a pixel.

Second, the proposed model can be applied to agricultural land and wetland imagery for detecting dynamic changes. The proposed model could bring benefit if applied to dynamic environmental sites, as agricultural land and wetlands need periodic monitoring for proper resource management. For example, monitoring healthy crops is important and detecting a polluted region and monitoring its recovery is also important. So, a pixel of dynamically changing environmental sites may show different abundances, as well as various topics distribution. To detect temporal changes in those sites, a dynamic topic model can be suggested. In this model, both the topics of the site’s previous status and the abundances of the previous time frame
can contribute to a dynamic model. Therefore, the Hidden Markov Model (HMM), which explains the sequential status of latent variables, can be combined with the proposed bag-of-materials model.


