Building Reliable Frameworks for 3D Object Classification Based on Bayesian and Deep Learning Approaches

Ahmed Yasser Eita

A Thesis
in
The Department
of
Concordia Institute for Information Systems Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of
Master of Applied Science (Quality Systems Engineering) at
Concordia University
Montréal, Québec, Canada

September 2023

Concordia University
School of Graduate Studies

This is to certify that the thesis prepared:

By: **Ahmed Yasser Eita**
Entitled: **Building Reliable Frameworks for 3D Object Classification Based on Bayesian and Deep Learning Approaches**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science (Quality Systems Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

Dr. Zachary Patterson ________________________Chair and Examiner

Dr. Arash Mohammadi ________________________Examiner

Dr. Nizar Bouguila ________________________Supervisor

Approved by ____
Dr. J. Yan, Graduate Program Director,
Concordia Institute for Information Systems Engineering

_______2023 ____
Dr. Mourad Debbabi, Dean
Faculty of Engineering and Computer Science

Abstract

Building Reliable Frameworks for 3D Object Classification Based on Bayesian and Deep Learning Approaches

Ahmed Yasser Eita

In the past decade, 3D objects have gained remarkable importance in everyday applications, and the ability to recognize them has therefore became a vital task in numerous fields. Ever since the emergence of 3D object recognition, there have been certain drawbacks that each newly invented model is striving to overcome. Among those shortcomings are; the ability to capture all critical features of the object, lack of spatial attributes consideration, insufficient visual relationships between semantic features, the necessity for expensive resources, and slow manipulation consequently. Computer Vision researchers have accomplished an excellent performance with multiple models, however, there is still an area for improvement. In this thesis, we are proposing two different novel 3D multi-view object classification methodologies inspired by Natural Language Processing (NLP) well-known approaches. The reason for this motivation is due to the NLP models' impressive capability in capturing the underlying characteristics in texts and the semantic feature relationships from sequential data types. The first model is a statistical approach, named F-GDA, which deploys Generalized Dirichlet (GD) distribution in all its priors to compose a fully flexible framework and the later one, named VAeViT, incorporates the reputed deep learning architectures; Variational Autoencoder (VAE) and Vision Transformer (ViT) to form a comprehensive structure. Each model has been innovatively invented to resolve some major limitations confronted by the model's methodology. Both models were evaluated on benchmark datasets and have proven reliably effective in classifying 3D multi-view objects and outperformed the state-of-the-art methodologies in the field.

Acknowledgements

First of all , I would like to express my deepest gratitude to my supervisor , Prof. Nizar Bouguila for supporting me on my Master’s journey and always providing me with the most impactful research tips and whichever resources I need. It was an honor be a student of his and I will always be thankful to him for providing me with this great opportunity.

Secondly, I would like to thank Hafsa Ennajari for working closely with me, always making the time whenever needed even though she was usually occupied by her work and other students’, and advising me with very useful tips on every step that I take. It could not have been done without you.

I want to thank my parents and wife for standing by my side during the whole time supporting me mentally, financially when needed, always praying for me and providing me with the love of the whole world. I love you very much.

Lastly, I want to thank my friends in Canada and overseas for motivating me to work hard and get it done, always checking on me, supporting me when needed , and being super happy for my success. I appreciate you a lot and for those who are overseas and I could not see them for a long time, you will always be the closest to my heart.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Advancements in 3D Object Recognition

In the ever-evolving landscape of computer vision, 3D object recognition stands as a pivotal frontier. The capacity to accurately identify and classify 3D objects is increasingly becoming integral to numerous fields, including robotics [1], autonomous navigation [2], medical imaging [3], bioinformatics [4], and scene monitoring [5]. This interdisciplinary effort combines progress in deep learning, neural networks, and sensor technology to understand complex spatial structures in the real world. It positions 3D object recognition as a vital link between digital and physical realms, poised to revolutionize industries, enhance human-machine interaction, and advance artificial intelligence.

1.1.1 Challenges in 3D Object Recognition Models

As 3D object recognition models continue to emerge, there are certain drawbacks that have always been obstructive from accomplishing a comprehensive model that would practically be sufficient when applied to real-world applications. These shortcomings include:

- The emergence of big data and large data collection require a highly efficient methodology that is capable of capturing the semantic structures in an unsupervised way, which would be otherwise impractical to annotate manually.

- Many widely utilized architectures in spite of handling the data manipulation effectively, suffer from a limited memory that would potentially miss critical features in the data with increasing the data-sequence length such as Recurrent Neural Network (RNN) [6], and Long Short-Term Memory (LSTM) [7].

- Independent analysis of points/views of the 3D object without spatial consideration yielding to only a local-precepting model that is incapable of recognizing the object from unseen angles rather than a comprehensive conceptive.
- Lack of grasping the relationships between visual features in objects which cause the model to fail in constructing a robust classification performance when exposed to changes in viewpoints, context, and occlusion.
- The necessity for expensive resources; huge amounts of data, high-quality inputs, and powerful processors that will certainly cause slow or/and costly outcomes which is definitely impractical in applications that require swift and high-quality results.

There are several models that excelled in overcoming some of those limitations via various methodologies, however, they must have lacked performance in one or a few of them as mentioned in sections 3.2 and 2.2, which is why we have committed to crafting models that can address all these limitations.

1.1.2 Diverse 3D Representation Types

The domain of 3D object representation is multifaceted, encompassing diverse methodologies tailored to capture spatial intricacies from varying perspectives. These methodologies have given rise to three prominent representation types: multi-view, voxel-based, and point cloud. Multi-view approaches provide a comprehensive 3D representation by utilizing multiple 2D views from various angles. This interpretable technique leverages multiple vantage points to construct a holistic understanding of an object's spatial attributes. Models like MVT [8], View-GCN [9], CAR-Net [10], RotationNet [11], and MVCNN [12] excel in this category. Voxel-based representations, conversely, decompose objects into volumetric grids, affording fine-grained 3D information. While computationally intensive, they provide a detailed structural view. Notable models in this domain include VRN Ensemble [13], LP-3DCNN [14], and 3DShapeNets [15]. Point cloud representations rely on spatially scattered points to encapsulate object surfaces, making them particularly apt for tasks like LIDAR-based perception. Pioneering models such as RS-CNN [16], LDGCNN [17], and PointNet++ [18] have made significant strides in this arena. Each approach has distinct strengths and limitations, with multi-view methods often excelling due to their interpretability and ability to capture object nuances from different angles while utilizing the least possible resource and generally outperforming all other paradigms. In light of all these factors, we chose to customize our novel models to be applied to the 3D multi-view representation.

1.1.3 Methodologies in 3D Object Classification

In the realm of 3D object classification, a diverse array of methodologies provides the scaffolding for these classification approaches. Each methodology brings its own unique strengths and perspectives, besides its common weaknesses. Convolutional Neural Network (CNN)-based models are the most obvious framework since it has gained remarkable success in image understanding in the past decade [19] and have been employed widely dominating the field of Computer Vision. Many 3D multi-view object recognition methodologies have therefore been motivated by that and attempted to deploy CNN in 3D object classification models [12][20][21] which achieved excellent performance with each approach seeking to address distinct challenges however, they suffer from some limitations as explained in section 3.2.

On a different realm, NLP methods have excelled in the tasks of acquiring the underlying characteristics of texts and grasping the relationships in sequential data types which is why they were an inspiration for 3D object classification methodologies and especially the multi-view representation type. Topic modeling methods are particularly effective in capturing the semantic structure of large data collections in an unsupervised manner due to their probabilistic nature and ability to discern the underlying by identifying latent topics based on the co-occurrence patterns of words within the data. This capability makes topic modeling highly adaptable to various types of data, ranging from text documents to images and more. Furthermore, topic models excel at handling the high dimensionality of large datasets. They employ techniques like dimensionality reduction to represent complex data in a more manageable form. This not only aids in visualization and interpretation but also facilitates downstream tasks such as clustering, classification, and recommendation systems. Therefore, utilizing the combination of Topic Modeling and Bag of Visual Words (BoVWs) [22], which is a feature extraction and image classification technique that is an adaptation of the Bag of Words model from Natural Language Processing as well, has been proven to be one of the most effective approaches for efficiently recognizing and classifying large visual data collections in an unsupervised way, which would be otherwise impractical to annotate manually [23].

Another very important NLP framework that has revolutionized the field recently and was the cornerstone of modern NLP models like BERT, GPT-3, and T5 is Transformers [24]. Transformers consist of an encoder block that processes the input to create a set of rich and contextually informed representations, and a decoder that uses these representations along with its own autoregressive context to generate an output sequence. They employ self-attention mechanisms to analyze contextual relationships within a text, enabling them to capture intricate linguistic nuances. Besides, the Positional Encoder layer plays a vital role in the preprocessing stage by adding positional information to the input which allows the model to recognize the spatial location of each data token and generate a well-ordered output. A Computer Vision model that is inspired by Transformers was subsequently developed which is the Vision Transformers (ViT) [25] model and it is currently the state-of-the-art framework in the field. ViT is known for its exceptional ability to capture the most critical features in the image without losing any important information because of short-term memory as the case in earlier models like RNN [6] and LSTM [7] by deploying the self-attention mechanism as in the conventional Transformer model. Also, the ability of the positional encoder layer to add positional information of image patches to draw a complete perception about the image, all have contributed to why ViT became the dominant framework.

In accordance with the above mentioned two NLP-inspired frameworks, we have developed two novel models for 3D multi-view object classification; F-GDA and VAeViT which are based on Bayesian and Deep Learning frameworks respectively. Each model was designed with careful consideration to overcome the major persisting obstructions faced by the methodology. Both models employ probability distribution in some phase as they provide a flexible framework for modeling complex relationships between features, able to model uncertainty which allows for more reliable and robust representations, can accommodate data of different scales, facilitates Bayesian inference as may be seen in section 2.4.2, and allow for the development of Bayesian neural networks in deep learning, which can provide better-calibrated uncertainty estimates as illustrated in section 3.3.1.

1.2 Contributions

Based on the constraints illustrated in section 1.1.1 and motivated by the NLP methodologies, we have contributed to this thesis with two novel models that are exclusively customized for the classification of 3D multi-view object type. The detailed impacts of each are demonstrated as follows:

- **Bayesian Fully Generalized Dirichlet Allocation Model:**

  We propose an efficient unsupervised probabilistic topic model, named F-GDA, which assumes a complete generative process by leveraging the Generalized Dirichlet (GD) distribution over all the priors, enabling a fully flexible model that generates more discriminative representations of objects while retaining an easy-to-understand and a simple-to-infer model by utilizing the Gibbs Sampling technique so that it can be applied to any large-scale application such as 3D object recognition. This model is focused more on the core generative process to resolve the complications encountered by other relevant topic models such as the topic correlation modeling issue and overfitting. Although this model is purely NLP-based in nature, we have preceded it with the BoVWs technique in order to tailor it for the classification of 3D multi-view objects. For a fair comparison, we initially executed an ablation study on a well-known gray-scale natural scene images dataset, N15 [26][27], to examine the competitiveness of our model's performance against other baseline models that have been evaluated on the same dataset. We then conducted extensive experiments on a benchmark 3D Multi-views dataset of real-world objects called ETH80 [28] to assess the performance of F-GDA in terms of accuracy, tolerance to topic correlation, descriptiveness, and scalability. The results of both datasets demonstrate the superiority of our proposed F-GDA model compared to the state-of-the-art Bayesian approaches.

- **Enhanced Vision Transformer Model with a Preceded Variational Autoencoder:**

  We present an enhanced Vision Transformer (ViT) model preceded by a Variational Autoencoder (VAE) model, named VAeViT, that is customized for the classification of 3D multi-view objects. This architectural refinement draws upon the established reliability of VAEs within the field of feature representation and further leverages the leading position of ViT in capturing semantic features from sequential data types. The VAeViT sequential model is designed to learn different levels or representations separately; VAE represents each 2D view in a low dimension latent vector whereas ViT utilizes those vectors to learn the deep feature representations of all the views and combine them with the added positional embedding information to draw a global perception of the 3D object. The idea of augmenting those two architecture allows the complete model to conquer the conventional ViT model's major limitation of necessitating a huge dataset and expensive resources in order to perform well. We also notice that the nature architecture of both models are ideal for mitigating the major deficiencies encountered by the 3D object recognition methods as illustrated in sections 3.1.1 and 3.1.2. Extensive experiments were conducted on two benchmark 3D multi-view datasets to prove the outperformance of VAeViT over the state-of-the-art models and the effect of critical attributes on the model's performance.

1.3 Thesis Overview

This thesis is structured as follows:

- In chapter 1, we introduce the 3D object recognition realm and discuss recent applications of the field and their significance. In subsections, we show the limitations encountered by most methodologies, various 3D object representation types, most common 3D object classification frameworks, and present our contributions to this thesis.

- In chapter 2, we describe the background of topic modeling, pioneering models in the field, the BoVWs preprocessing technique, inference methodologies, and propose our 3D-customized Bayesian approach. We also demonstrate a detailed inference of the derivation of our generative model and how this unique addition contributes to resolving the drawbacks of the topic modeling methodology. We lastly, conduct an ablation study to compare our designed framework with baseline models and investigate its performance on an RGB 3D multi-view dataset.

- In chapter 3, we focus on Deep Learning methodologies and develop our second novel architecture by adapting two pioneering structures of those, after we have explored the most dominant methodologies in the field and their main defects. A comprehensive explanation of the innovative model's architecture, its components and the process flow are then demonstrated. Extensive assessments on two 3D multi-view object datasets are subsequently presented to show the effectiveness of our model and that it has outperformed the state-of-the-art methods.

- In chapter 4, we briefly summarize our contributions, provide concluding remarks, and demonstrate some potential future work directions.

Chapter 2

Bayesian Fully Generalized Dirichlet Allocation Model

2.1 Background

Topic modeling is a vital technique in NLP that serves as a powerful tool for uncovering latent thematic structures within large collections of textual data. By employing sophisticated algorithms, topic modeling extracts underlying patterns and identifies coherent topics that are prevalent across diverse documents. This capability aids in the organization and summarization of extensive datasets [29] and makes topic modeling highly adaptable to various data types, such as images, videos and more. This unsupervised learning approach has found extensive applications in diverse fields [30]; including data clustering [31], anomaly detection for videos [32], and image spam filtering [33].

One of the most common topic modeling approaches is Latent Dirichlet Allocation (LDA) [34], which derives its priors from the Dirichlet distribution and has shown promising results for different downstream tasks. However, LDA suffers from topic correlation issues due to the inflexibility of its priors, induced by the Dirichlet distribution. To overcome these limitations, several approaches that integrate more flexible priors have been introduced. Examples of those models include Pachinko Allocation Model (PAM) [35], Correlated topic model (CTM) [36], Generalized Dirichlet LDA (GD-LDA) [37], Latent Generalized Dirichlet Allocation (LGDA) [38], and Collapsed Variational Bayes Latent Generalized Dirichlet Allocation (CVB-LGDA) [39] in which only a few of them have been successful in identifying and capturing semantic relationships among topics. Nevertheless, many others either suffer from incomplete generative processes that adversely affect the efficiency of parameter inference or result in low performance due to the complexity of the model and the high number of learning parameters.

Topic models are probabilistic models that seek to uncover the latent topics that generate the observed documents. However, it is often intractable to directly calculate the exact distribution of these latent variables given the data due to the complexity of the models. Therefore, the utilization of an inference method when deriving topic models is crucial because they enable us to estimate the underlying structure of the data. Variational Inference [31], MCMC (Markov Chain Monte Carlo) [40], and HMM (hidden Markov model) [32] are the prominent inference techniques, each offering distinct advantages. Variational Inference approximates complex probability distributions with simpler, parameterized ones. While computationally efficient, it may sometimes lead to biased estimates. HMMs can struggle with capturing long-range dependencies, potentially leading to oversimplification or overfitting. On the other hand, Gibbs Sampling is an MCMC method that iteratively samples from conditional distributions. It provides unbiased estimates, but can be computationally expensive. In practice, Gibbs Sampling tends to be more robust and reliable, especially when dealing with complex, high-dimensional data.

Since we are committed to tailor our novel model for 3D multi-view object representation type which basically consists of images captured from multiple angels, there has to be a preprocessing methodology that can refine those 2D views into the same input form that topic models would accept which is in a Bag of Words (BoWs) arrangement as per the case for textual data types. The Bag of Visual Words (BoVWs) is an adaptation of BoWs that revolutionized the way we analyze and understand visual data. It operates on the premise that an image can be represented by a histogram of visual words. This technique involves breaking down an image into smaller, discernible components, extracting their features, and quantizing them into a predefined visual vocabulary. BoVW has proven to be immensely useful in tasks like object recognition, image categorization, and scene understanding. By converting complex visual information into a structured, quantitative format, BoVW serves as a cornerstone in the development of robust and efficient computer vision systems.

2.2 Related Work

LDA [34] paved the way for precise new document predictions, and research in topic modeling has since been focused on developing variants and extensions of LDA to overcome its limitations. Models such as PAM [35], GD-LDA [37], VarInGDM [33], and CVB-LGDA [39] have been proposed to improve upon LDA, with the aim of capturing topic correlation and avoiding overfitting. Researchers have been trying to explore other distributions to tolerate the topic correlation issue ever since, such as the logistic normal distribution which CTM [36] and IFTM [41] models are derived from but the distribution was not conjugate to the multinomial distribution. As a result, these two models and PAM [35] have proven to be complex to implement due to the incompatibility and proneness to overfitting, while models such as CVB-LDA [42], S-LDA [43], and all GD derived models have shown better performance in scenarios where topic size increases due to the additional integrated features, collapsed space of latent variables in the first, and the spatial information in the second to be precise. The generative process of CVB-LDA [42] and S-LDA [43] are mainly considering the Dirichlet distribution for the model priors. As a result, the models have a limited ability to take topic correlation into account. Another major gap in these approaches is the lack of robustness to ensure good performance for large-scale data, given the vocabulary size and the average word length per document.

The introduction of the Generalized Dirichlet distribution has presented another approach for topic analysis to address the issue of topic correlation. GD-derived models such as GD-LDA [37] and LGDA [38] are efficient in preventing overfitting when the number of topics is increasing, but suffer from an incomplete generative process. GD-LDA [37] has developed a Gibbs sampling inference scheme using GD as a prior to LDA, but has only done so for the topics parameter which is inefficient in the case of a large vocabulary size within the BoVW framework. On the other hand, CVB-LGDA [39] dominates a complete generative process that is derived from the GD distribution however it is computationally expensive due to the complexity of the model and the high number of parameters to be learned during the training.

Gibbs sampling as compared to other learning algorithms for LDA such as variational EM and Expectation-Propagation has been demonstrated to be more efficient in [44]. This efficiency can be attributed to LDA's inherent property of conjugacy between the Dirichlet prior and the multinomial one. As a result, Gibbs sampling algorithms have been developed for many models that extend LDA, including [37][23][40]. Given its suitability for sampling from complex distributions, we chose Gibbs sampling as the inference method for our approach.

In this chapter, we introduce F-GDA, a novel Bayesian Fully Generalized Dirichlet Allocation Model for 3D objects classification. Unlike existing models, F-GDA aims to capture topic correlation and avoids overfitting while assuming a complete probabilistic process that draws both the documents-topics and the topics-words distributions from a Generalized Dirichlet distribution to guarantee a completely priors-flexible model. As for the model parameters inference, we develop an efficient Markov Chain Monte Carlo (MCMC)-based sampling approach that approximates the model latent parameters. Our sampling method is proven to be less complex, more tolerant to local optima, and does not suffer from large biases so that the model can be applied to any large-scale 3D objects data type such as the 3D-Multi-views objects.

2.3 Generalized Dirichlet Distribution

The Generalized Dirichlet (GD) distribution was developed as a response to the limitations of the Dirichlet distribution since it has a more general covariance structure making it more practical and useful in capturing data covariances [40]. Dirichlet distribution affects the resilience of the model when the probability of a sample is changed since all the entries of a random vector of proportions in this distribution must share a common variance and must sum up to one. In addition, it has a limited degree of freedom when used as a prior for Multinomial distribution. Hence, the ability to sample each entry of the proportions vector from independent Beta distributions is the key property of the GD distribution as it provides more flexibility from this perspective. The GD distribution of the topics parameter can be defined as:

$$p(\theta|\alpha,\beta) = \prod_{j=1}^{K-1} \frac{\Gamma(\alpha_j+\beta_j)}{\Gamma(\alpha_j)\Gamma(\beta_j)} \theta_j^{\alpha_j-1}\,(1-\theta_1-\cdots\theta_j)^{\eta_j} \tag{2.1}$$

where $\theta_1+\theta_2+\cdots+\theta_{K-1}+\theta_K=1, \eta_j=\beta_j-\alpha_{j+1}-\beta_{j+1}$ for $1\leq j\leq K-2$ and $\eta_{K-1}=\beta_{K-1}-1$.

Similar to the Dirichlet distribution, the Generalized Dirichlet distribution is a conjugate prior distribution to the Multinomial distribution as illustrated in detail in [45]. Thus, we can integrate the parameter of the Multinomial distribution to obtain:

$$p(T|\alpha,\beta) = \int p(T|\theta)p(\theta), d\theta = \prod_{k=1}^{K-1} \frac{\Gamma(\alpha_k+\beta_k)}{\Gamma(\alpha_k)\Gamma(\beta_k)} \prod_{k=1}^{K-1} \frac{\Gamma(\alpha'_k)\Gamma(\beta'_k)}{\Gamma(\alpha'_k+\beta'_k)} \tag{2.2}$$

where T is a discrete random variable derived from Multinomial distribution with parameter $\theta_1 \ldots \theta_K$ and $\alpha'_k=\alpha_k+T_k$, $\beta'_k=\beta_k+T_{k+1}+\cdots+T_K$. Since this integral cannot be computed analytically in a closed form, we utilize Gibbs sampling in our approach to approximate the hidden parameters of the model as shown in section 2.4.2. Another key property of the GD distribution is that it facilitates data dimensionality reduction within its cascaded tree-based structure. For example, if β_k is too small compared to α_k, we could dispose of the lower-level topics in the structure making the model more robust and effective in reducing the dimensionality of topics.

2.4 Proposed Model

In this section, we introduce first the generative process of our proposed model in detail, and then we present our Gibbs sampling-based method for learning the F-GDA model's hidden parameters. The mathematical notations used in our model are all summarized in the table below.

Table 2. 1: Summary of Mathematical Notations.

Notation	Meaning
D	number of documents
K	number of topics
V	vocabulary size
j	index of a document in the collection
k	a specific topic
i	index of a word in the vocabulary
w	specific observed word
z	topics assignments
$z_{w,j}$	topic assignment of word w at the document index j
N_j	number of words in the document at index j
$N_{k,i}$	frequency of the word at index i that is assigned to topic k
$N_{j,k}$	the frequency of topic k in the document at index j
x^{-wj}	a quantity without accounting the word w in document j
θ	mixture probabilities of K topics
ϕ	mixture probabilities of V words
α, β	GD hyperparameters of the document-topics parameter
λ, η	GD hyperparameters of the topic-words parameter
$GenDir(x)$	Generalized Dirichlet distribution
$Mult(x)$	Multinominal distribution

2.4.1 Generative Process

The generative process of the F-GDA topic model is grounded on fully probabilistic foundations. Instead of drawing the prior of the document-topic proportions and topic-word vectors from a Dirichlet distribution, we draw these priors from a Generalized Dirichlet distribution to better handle the issue of correlation between topics and allow for more flexible information sharing between the model components.

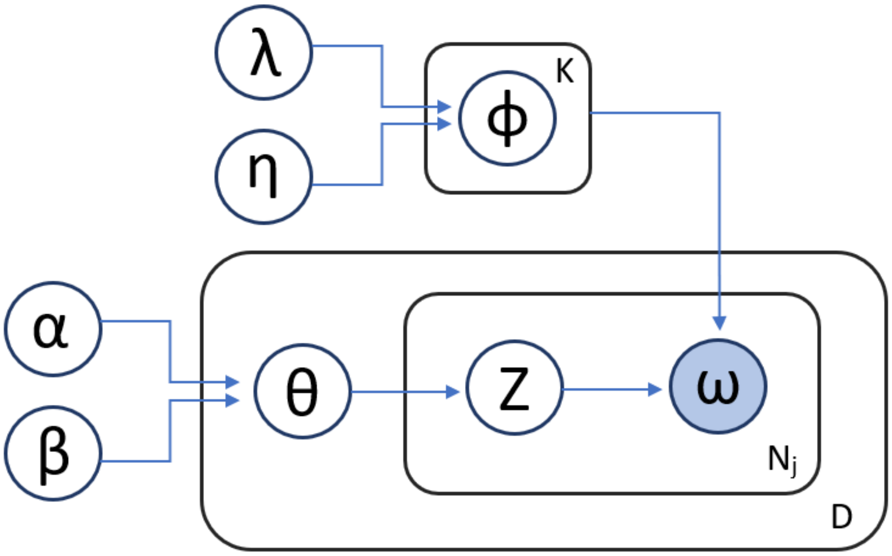


Figure 2.1: F-GDA Graphical Model Representation.

The graphical model representation of F-GDA is illustrated in Figure 2.1, and its complete generative model is outlined in Algorithm 1. From the generative process of F-GDA, we can see that it allows for sampling from a richer family of distributions, namely, the GD distribution, which can help generate more discriminative representations of objects, and capture the semantic structure of large data collections.

Algorithm 1 F-GDA Generative Model

```
for topic $k \leftarrow 1$ to $K$ do
    draw $\phi_k \sim GenDir(\lambda, \eta)$
end for
for document $j \leftarrow 1$ to $D$ do
    draw $\theta_j \sim GenDir(\alpha, \beta)$
    for word $w \leftarrow 1$ to $N_j$ do
        draw $z_{w,j} \sim Mult(\theta_j)$
        draw $w|z_{w,j} \sim Mult(\phi_{z_{w,j}})$
    end for
end for
```

2.4.2 Model Fitting

From the graphical model in Figure 2.1, we can see the F-GDA model has three unobserved parameters θ, ϕ, z and four priors hyperparameters $\lambda, \eta, \alpha, \beta$. Our objective is to approximate the values of the model's hidden parameters, mainly the topic's assignments vector given the observed ones. To do so, we develop an efficient Gibbs sampling method to approximate the hidden topic assignment $z_{w,j}$. In this work, we have assumed the hyperparameter vectors to be of fixed values in order to avoid overfitting from learning too many parameters and for the sake of simplifying the model and reducing the computation cost. Formally, the joint probability associated with the F-GDA probabilistic model is defined as:

$$p(w, z|\alpha, \beta, \lambda, \eta) = p(w|z, \lambda, \eta)p(z|\alpha, \beta) \tag{2.3}$$

The joint probability consists of two probabilities where the first one describes the distribution of words while the second one is for topics distribution. Starting first with the document-topics parameter θ, we split the probability into two separate probabilities for simplification in which, we derive the first one from a Multinomial distribution and the other one from a GD distribution:

$$p(z|\alpha, \beta) = \int_{\theta} p(z|\theta)p(\theta|\alpha, \beta)\, d\theta \tag{2.4}$$

$$p(z|\alpha, \beta) = \prod_{d=1}^{D} p(T|\theta)p(\theta|\alpha, \beta) \tag{2.5}$$

By substituting equation (2.2) and the GD distribution (2.1), we obtain:

$$p(z|\alpha, \beta) = \prod_{j=1}^{D} \prod_{k=1}^{K-1} \frac{\Gamma(\alpha_k + \beta_k)}{\Gamma(\alpha_k)\Gamma(\beta_k)} \prod_{k=1}^{K-1} \frac{\Gamma(\alpha_k^j)\Gamma(\beta_k^j)}{\Gamma(\alpha_k^j + \beta_k^j)} \tag{2.6}$$

Similarly for the topic-words parameter ϕ:

$$p(w|z, \lambda, \eta) = \int_{\phi} p(w|z, \phi)p(\phi|\lambda, \eta)\, d\phi \tag{2.7}$$

$$p(w|z, \lambda, \eta) = \prod_{k=1}^{K} p(T|\phi)p(\phi|\lambda, \eta) \tag{2.8}$$

Then to substitute the GD distribution (2.1) and the expression in (2.2) while applying them on the words parameter ϕ instead, along with its hyperparameters λ and η, we get:

$$p(w|z,\lambda,\eta) = \prod_{k=1}^{K} \prod_{i=1}^{V-1} \frac{\Gamma(\lambda_i + \eta_i)}{\Gamma(\lambda_i)\Gamma(\eta_i)} \prod_{i=1}^{V-1} \frac{\Gamma(\lambda_i^k)\Gamma(\eta_i^k)}{\Gamma(\lambda_i^k + \eta_i^k)} \tag{2.9}$$

Now, we define Gibbs sampling in order to infer latent topic assignments z:

$$p\left(z_{wj} = k \middle| z^{-wj}, \alpha, \beta, \lambda, \eta\right) = \frac{p(w|z,\lambda,\eta)p(z|\alpha,\beta)}{p(w|z^{-wj},\lambda,\eta)p(z^{-wj}|\alpha,\beta)} \tag{2.10}$$

where z_{wj} represents the topic assignment for the word w at document index j while z^{-wj} represents the topic assignments for all the other words except the current word w at index j. By integrating out the parameters, the Gibbs sampling equations for words and topics are obtained as:

$$\frac{p(w|z,\lambda,\eta)}{p\left(w\middle|z^{-wj},\lambda,\eta\right)} = \begin{cases} \dfrac{\lambda_i + N_{k,i}^{-wj}}{\lambda_i + \eta_i + \sum_{x=1}^{V} N_{k,x}^{-wj}}, & i = 1 \\ \dfrac{\lambda_i + N_{k,i}^{-wj}}{\lambda_i + \eta_i + \sum_{x=i}^{V} N_{k,x}^{-wj}} \displaystyle\prod_{y=1}^{i-1} \dfrac{\eta_y + \sum_{x=y+1}^{V} N_{k,x}^{-wj}}{\lambda_y + \eta_y + \sum_{x=y}^{V} N_{k,x}^{-wj}}, & 1 < i < V \\ \displaystyle\prod_{y=1}^{V-1} \dfrac{\eta_y + \sum_{x=y+1}^{V} N_{k,x}^{-wj}}{\lambda_y + \eta_y + \sum_{x=y}^{V} N_{k,x}^{-wj}}, & i = V \end{cases} \tag{2.11}$$

$$\frac{p(z|\alpha,\beta)}{p\left(z^{-wj}\middle|\alpha,\beta\right)} = \begin{cases} \dfrac{\alpha_k + N_{j,k}^{-wj}}{\alpha_k + \beta_k + \sum_{l=1}^{K} N_{k,l}^{-wj}}, & k = 1 \\ \dfrac{\alpha_k + N_{j,k}^{-wj}}{\alpha_k + \beta_k + \sum_{l=k}^{K} N_{j,l}^{-wj}} \displaystyle\prod_{m=1}^{k-1} \dfrac{\beta_m + \sum_{l=m+1}^{K} N_{j,l}^{-wj}}{\alpha_m + \beta_m + \sum_{l=m}^{K} N_{j,l}^{-wj}}, & 1 < k < K \\ \displaystyle\prod_{m=1}^{K-1} \dfrac{\beta_m + \sum_{l=m+1}^{K} N_{j,l}^{-wj}}{\alpha_m + \beta_m + \sum_{l=m}^{K} N_{j,l}^{-wj}}, & i < V \end{cases} \tag{2.12}$$

Equations (2.11) and (2.12) are used to estimate the topic-words and document-topics parameters respectively and by replacing them in equation (2.10), we obtain the inference of the topic assignment of a word. It is worth noting that with the use of the product in equations (2.11) and (2.12), new samples assigned to a topic k will affect all the other topics, and the impact of this assignment is dependent on both hyperparameters of the GD prior, in contrast to LDA which has no effect on the sampling distribution of other topics and that is also dependent on only one hyperparameter.

2.5 Experimental Results

In this section, we empirically evaluate the effectiveness of our F-GDA model on two datasets to determine the ability of our model to better recognize 3D objects and classify images.

2.5.1 Datasets

- **N15** [26][27]: is a gray-scale natural scenes images dataset, which consists of 15 categories. This images dataset is only utilized as an ablation study (section 2.5.4) to compare our model with the baseline approaches. For a fair comparison, we followed the same settings in [39] during the evaluation of this dataset, by randomly choosing 9 categories which are office, mountain, forest, store, street, suburb, coast, highway, and living room.

- **ETH80** [28]: is a real-world 3D multi-views objects dataset comprising 80 objects, divided into 8 classes, with each class containing 10 objects. Additionally, there are 41 uniformly spaced views of each object over the upper viewing hemisphere. For the evaluation of this dataset, the whole dataset is considered with 80% for training and 20% for testing.

2.5.2 Feature Representation

Setting up the dataset in the correct format for training carries huge importance within the process flow as it can directly affect the results if they are misrepresented. Also, an efficient and robust model requires well-represented data in the feature space where key features in the object or image are assured to be the core of the representation and in a compact way as well. Our F-GDA model can be applied to any large-scale dataset such as 3D objects, images, as well as textual data. However, the following steps describe the setup phase for visual data types, represented in the form of BoVWs.

1. Features Extraction

Feature extraction is the first step in our framework that aims to find the most unique features in objects/images such as corners or blobs. Choosing an appropriate feature detection technique is a critical step to maximize the discriminative property in each image or object. According to the dataset types, there is a wide range of descriptors to choose from such as SIFT (Scale Invariant Feature Transform) [46], KAZE [47], and ORB (Oriented FAST and Rotated BRIEF) [48]. SIFT is the most commonly used technique by researchers due to its ability to be invariant to affine transformations and occlusions which is why it was chosen by most works and too for ours to elaborate on the exact effect of our proposed model as compared with other baselines, but not to neglect the high performance of the other methods. With the SIFT descriptor, each local feature is represented by a 128-dimensional descriptor which is a vector of numerical values that describes the feature's surroundings.

2. Codeword Standardization

At this stage, each image or object is represented as a collection of features. In order to enable the representation and recognition of each object in a distinct manner, standardization of all the unique features for the whole dataset is essential which is done by using the K-means clustering algorithm to group similar patches together where the center, depicts the standard mean feature of this cluster. The number of clusters chosen will correspond to the number of standardized centers that will later be defined as what we call the number of words/clusters or dataset vocabulary size.

3. Dictionary Formation

The features dictionary is formed by quantizing each vector of features of an object in the dataset against the standardized codeword. By doing so, each object will only be expressed by the same set of known features (with different frequencies) so that they can be understood by the model. Lastly, we append all those obtained quantized objects together to finally form the dictionary.

2.5.3 Experiments Setup

For both datasets, it is important to note that the whole data is preprocessed and fed to F-GDA whereas the training-testing split only comes after representing the objects as mixtures of topics before evaluating the model using the SVM classifier. The reason for not splitting the data from the beginning is that there is a significant amount of randomness involved in the preprocessing and the topic modeling stages. If the same randomness is not used for individual splits, the results will be erroneous, which could mislead the learning model.

2.5.4 Ablation Study

In this section, we evaluate F-GDA on the N15 dataset and compare its performance with other baseline models as an ablation study before assessing it on the 3D object dataset since not many previous topic models have adapted 3D object classification. We selected the N15 dataset because grayscale datasets tend to reduce computational complexity and improve processing speed, in addition to having been evaluated by most of the baseline models we are comparing our model to. Similar to [39], we assume the number of words and topics to be 900 and 90, respectively. We report in Table 2.2 the performance comparison of our proposed model versus state-of-the-art models in terms of classification accuracy.

Table 2.2: F-GDA accuracy comparison with baseline model on N15.

Model	LDA [34]	CVB-LDA [42]	LGDA [38]	GD-LDA [37]	CVB-LGDA [39]	**F-GDA**
Accuracy %	54.1	60	65.69	68.69	72.78	**72.50**

It can be seen that F-GDA achieves promising and competitive results compared to the baselines with a high accuracy of 72.50%. Although this is slightly lower than the accuracy CVB-LGDA [39], F-GDA outperforms in the overall performance as it is simpler to integrate, has fewer learning parameters, and requires less computational power.

We also assess the performance of the classifier using the ROC curve for each category against the rest and illustrate the results in Figure 2.2 along with the confusion matrix in Figure 2.3 which measures the dependency between each two categories in this classification problem.

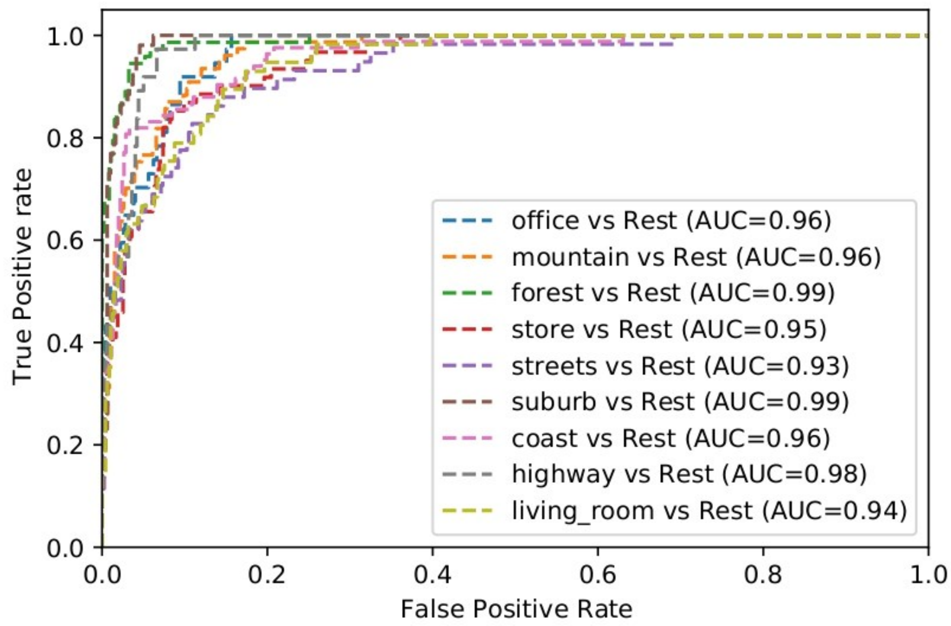
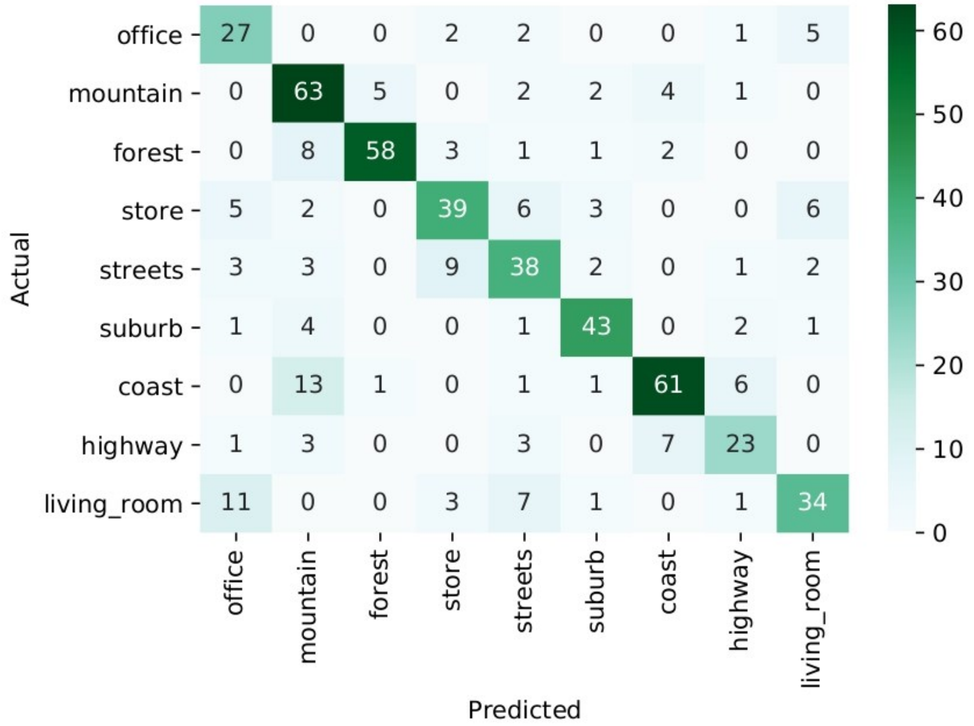


Figure 2.2: ROC curves of the F-GDA model on N15.



	office	mountain	forest	store	streets	suburb	coast	highway	living_room
office	27	0	0	2	2	0	0	1	5
mountain	0	63	5	0	2	2	4	1	0
forest	0	8	58	3	1	1	2	0	0
store	5	2	0	39	6	3	0	0	6
streets	3	3	0	9	38	2	0	1	2
suburb	1	4	0	0	1	43	0	2	1
coast	0	13	1	0	1	1	61	6	0
highway	1	3	0	0	3	0	7	23	0
living_room	11	0	0	3	7	1	0	1	34

Figure 2.3: Confusion matrix of the F-GDA model on N15.

2.5.5 Results and Discussion

As for the 3D multi-view objects dataset, ETH80, we conducted extensive experiments to compare our F-GDA model's performance majorly with LDA to truly showcase the potential of the GD distribution over the Dirichlet distribution. In addition, we performed parameter searches to obtain the optimal model settings and compared it with state-of-the-art models that were evaluated on the same dataset. Figure 2.4 shows the topic search for both LDA and F-GDA. During the parameters search, we observed that F-GDA outperformed other models when the topic number is higher than 60, with the highest accuracy achieved with 90 topics. Figure 2.5 shows the performance of LDA and F-GDA in terms of various metrics using the optimal number of topics from Figure 2.4.

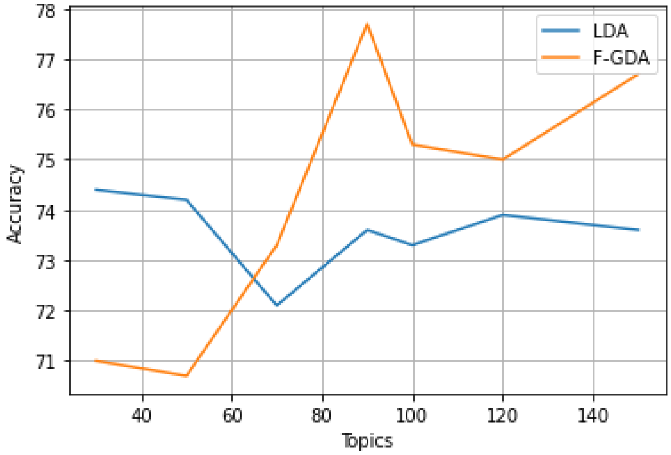


Figure 2.4: Topics search analysis of LDA and F-GDA on ETH80.

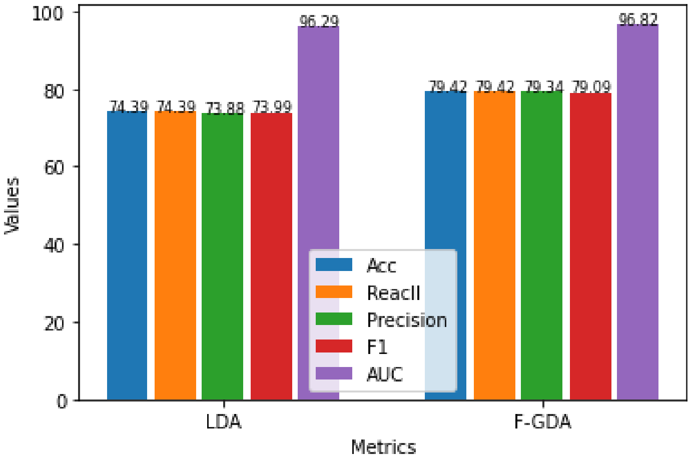


Figure 2.5: Performance comparison between LDA and F-GDA using different metrics on ETH80.

Since F-GDA has shown promising results, further parameter searches were carried out. Figure 2.6 shows a cluster search experiment to identify the highest-performance cluster number. The figure shows that the best number of clusters that captures the key features when using F-GDA on the ETH80 dataset is 2,500 clusters as visual words start to be redundant after that, resulting in lower accuracy. We conducted another search for topics to find the most optimal combination of parameters for the model. We report in Table 2.3 the best combination of these settings for F-LDA evaluated with different metrics. In addition, the ROC curve for each class versus the rest and the confusion matrix of the most optimum model are illustrated in Figure 2.7 and Figure 2.8, respectively, to show the interdependence between each two classes.

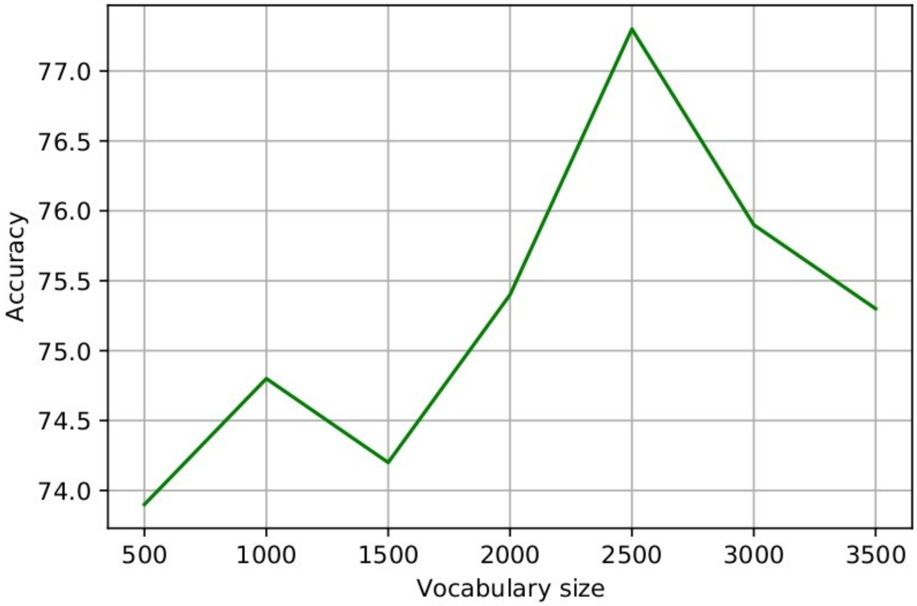


Figure 2.6: Clusters search analysis of the F-GDA model on ETH80.

Table 2.3: F-GDA performance with optimal settings on ETH80.

Optimal setting	2500 clusters – 100 topics – 160 iterations				
Metric	Accuracy	Recall	Precision	F1	AUC
Performance %	79.42	79.42	79.34	79.09	96.82

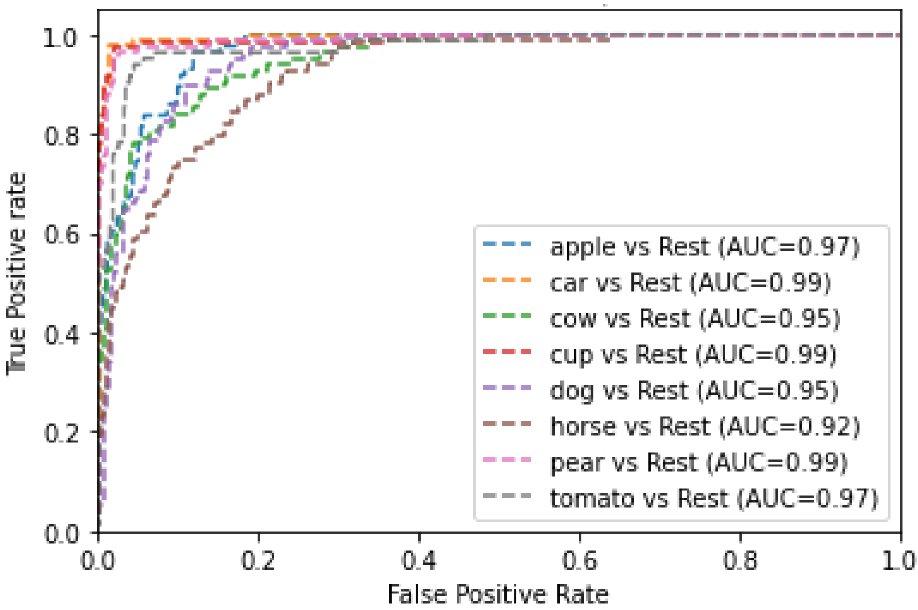
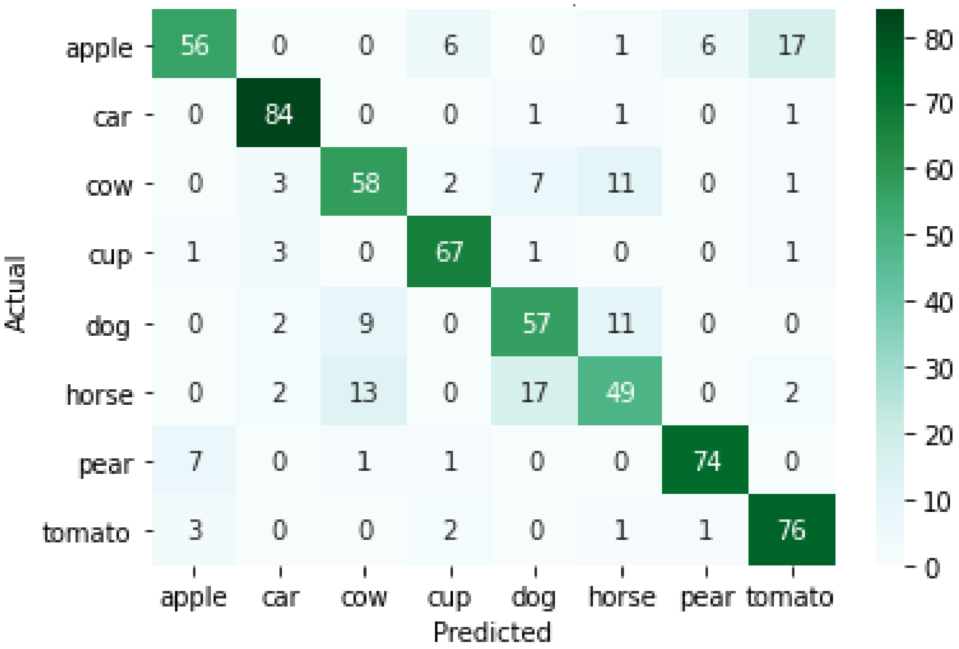


Figure 2.7: ROC curves of the F-GDA model on ETH80.



	apple	car	cow	cup	dog	horse	pear	tomato
apple	56	0	0	6	0	1	6	17
car	0	84	0	0	1	1	0	1
cow	0	3	58	2	7	11	0	1
cup	1	3	0	67	1	0	0	1
dog	0	2	9	0	57	11	0	0
horse	0	2	13	0	17	49	0	2
pear	7	0	1	1	0	0	74	0
tomato	3	0	0	2	0	1	1	76

Figure 2.8: Confusion matrix of F-GDA on ETH80.

Finally, Figure 2.9 provides the precision versus recall curve for the most optimal F-GDA model and is fitted against LDA, S-LDA and other baseline models in [43] that were evaluated on ETH80. This further confirms that our model has surpassed all of them in terms of precision-recall performance.

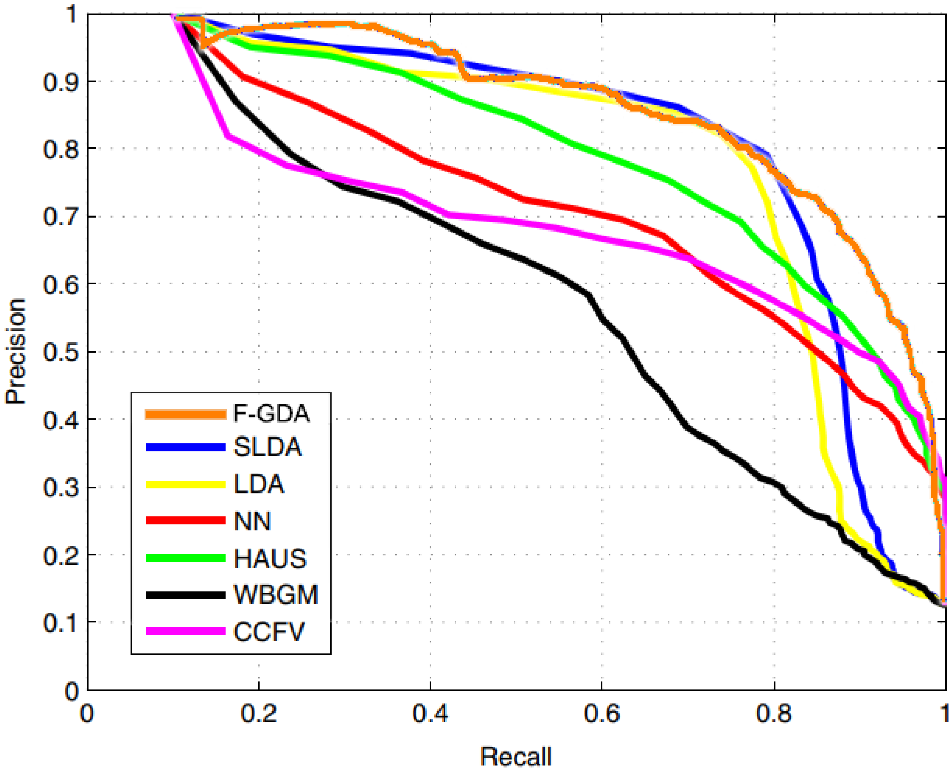


Figure 2.9: Precision-Recall performance of F-GDA and all the baseline models on ETH80.

Chapter 3

Enhanced Vision Transformer Model with a Preceded Variational Autoencoder

3.1 Background

In this section, we present a background study on major components of our model; Variational Autoencoder (VAE) and Vision Transformer (ViT) in which we present their architectural flow and major components since we will be utilizing them in our novel architecture.

3.1.1 Variational Autoencoder

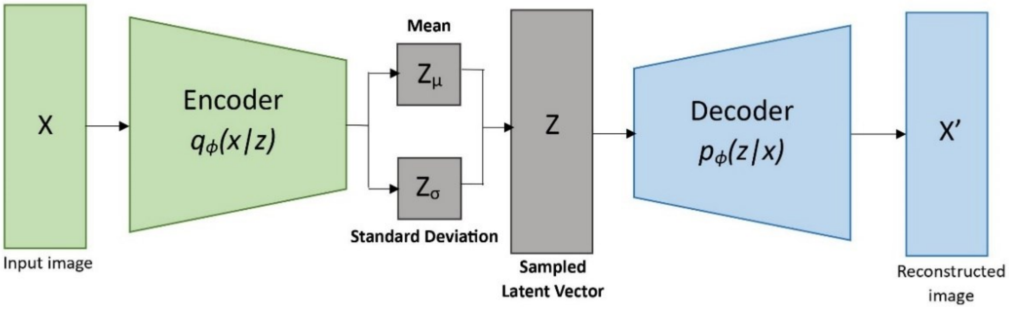


Figure 3.1: Variational Autoencoder architecture.

The Variational Autoencoder (VAE) [49][50] plays a pivotal role in feature extraction within the realm of deep learning. VAE structure consists of two major components; an encoder, denoted as $q_{\phi}(z|x)$, and a decoder, denoted as $p_{\theta}(x|z)$. The encoder approximates the function that maps the input image X from its original form into a lower meaningful dimension after it has been structured. The decoder then attempts to recreate the original input using the output from the encoder, generating output image X'. Instead of mapping the output of the encoder to a fixed latent vector as the case in Autoencoders, we map it to a probability distribution. The encoder outputs two parameters which are typically the mean Z_μ, and standard deviation Z_σ for each dimension of the latent space. Subsequently, these parameters are utilized to define a probability distribution (often Gaussian) from which we can sample to obtain the latent vector Z in the bottleneck layer. This allows the model to capture more nuanced information about the image and represent it in a more expressive and powerful manner as the ensures that neighboring data points are likely to have similar latent representations in the continuous latent space.

Since the gradient cannot be pushed through a sampling node during the backpropagation process of the training, a technique called reparameterization trick is applied. In the reparameterization trick, the sampling equation being employed is as below:

$$Z = \mu + (\sigma \odot \varepsilon) \tag{3.1}$$

where $\varepsilon \sim Normal(0,1)$. From the above equations, it can be noticed that ε is the only stochastic node and it is fixed, therefore, the training can be run through μ and σ normally.

During the training, the VAE model aims to minimize two losses; reconstruction loss in equation (3.2) which intends to minimize the reconstruction error between the original image and the generated image, and KL-divergence loss in equation (3.3) which forces the distribution to be as close as possible to the standard normal distribution that is centered around 0. Doing so will provide continuous data or a range of data in the latent space so that we are able to access the targeted distribution correctly and construct meaningful output. The overall loss function of the VAE model is therefore the summation of both losses (equation (3.4)).

$$L(X, X') = -E_{z \sim q_\phi(z|x)}[log P_\theta(x|z)] \tag{3.2}$$

$$D_{KL}(q_\phi(z|x)||p_\theta(z|x)) = -\frac{1}{2}\sum(1 + log(\sigma^2) - \mu^2 - \sigma^2) \tag{3.3}$$

$$L_{VAE}(\theta, \phi; x) =$$

$$\frac{1}{2}\sum(\mu^2 + \sigma^2 - log(\sigma^2) - 1) - E_{z\,\sim\, q_{\phi}(z|x)}[logP_{\theta}(x|z)] \tag{3.4}$$

By learning a probabilistic mapping from the data space to the latent space, VAE effectively capture the underlying structure of the input images and thus is able to represent them in a lower dimension while preserving the essential characteristics. This capability makes VAEs invaluable for tasks such as feature extraction and data compression [51], anomaly detection [52], and generation [53], where they excel at disentangling and representing complex features. Through probabilistic modeling and encoding, VAE contribute significantly to unsupervised learning and feature extraction, which is why it is a perfect fit in our model to precede ViT in order to embed the object views in a lower meaningful dimension.

3.1.2 Vision Transformer

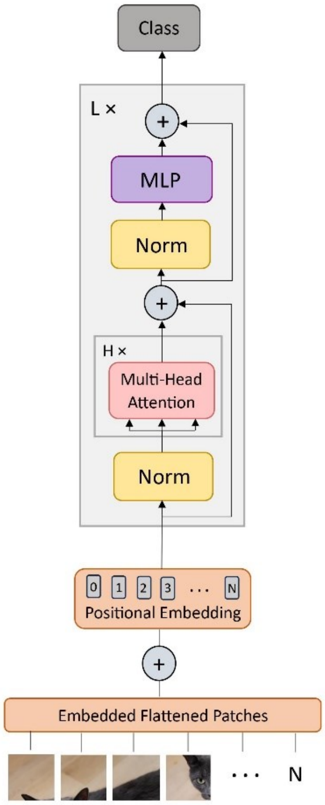


Figure 3.2: Vision Transformer architecture.

Vision Transformers (ViT) [25] have emerged as a pioneering paradigm in computer vision, representing a departure from traditional Convolutional Neural Networks (CNNs). ViT replaces the grid-based operations of CNNs with self-attention mechanisms presented in [24], enabling a more flexible and holistic understanding of visual data. This architectural refinement draws upon the established reliability of Transformers within the field of natural language processing or sequential data types to be specific. However, the major difference in architecture between Transformers and Vision Transformers is that the ViT does not have a decoder since we are not interested in generating data but, the primary goal is to extract meaningful features and to understand the spatial relationships in the image so the encoder only in the ViT performs this task.

ViT models have demonstrated exceptional capabilities across various computer vision tasks, including image classification [25], object detection [54], and semantic segmentation [55]. Their hierarchical self-attention mechanism allows for capturing long-range dependencies and global context, making them particularly well-suited for tasks requiring holistic scene understanding. The encoder views the encoded input representation as a set of key K, value V, and query Q vectors of the same size as the input, and then, the Transformer adopts the scaled dot-product attention to generate an output that is the weighted sum of the value vectors, where the weight assigned to each value slot is determined by the dot-product of the query and its corresponding key:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \qquad (3.5)$$

where d_k is the dimension of key vectors. Self-attention mechanism is deployed in the multi-head attention layer which consists of h number of cascaded heads where each of them works independently and then they all get concatenated to construct an output that is able to attend to information from different representation subspaces at various positions.

A pivotal component of ViT's success is the integration of positional encoding, where each patch of an image is supported by positional information enabling the model to understand not just the content of the image but also its spatial arrangement.

Layer Normalization, usually referred to as the Norm block, is a type of normalization that is fatal to the model to improve the training and generalization of the model, which is unlike the batch normalization, it applies the normalization to each feature in the layer independently. This means that for each feature x in the input embedded vector $X = [x_1, \dots, x_D]$ where D is the vector size, the mean μ and standard deviation σ are computed as follow and used to normalize the feature values.

$$\mu = \frac{1}{D}\sum_{i=1}^{D} x_i \tag{3.6}$$

$$\sigma = \sqrt{\frac{1}{D}\sum_{i=1}^{D}(x_i - \mu)^2} \tag{3.7}$$

After the mean μ and standard deviation σ are computed, a linear operation is conducted on each element of x as below:

$$x_i' = \alpha\frac{x_i - \mu}{\sigma} + \beta \tag{3.8}$$

where α and β are learnable parameters for affine transform.

Apart from the number of attention heads in the multi-head attention layer, the Transformer encoder itself is also constructed by an l number of stacked layers then they all get concatenated to further enhance the capability of the model in capturing the useful features. Figure 3.2 displays the architecture of the Vision Transformer where N is the number of patches per image, $H \times$ is the number of cascaded self-attention heads, and $L \times$ is the number of stacked Transformer layers. By pre-training on huge-scale datasets, ViT has achieved comparable accuracy compared with its CNN counterparts.

The innovative structure of ViT was an inspiration within the computer vision community to enhance existing models to overcome certain limitations. For instance, these works [56]–[58] incorporated the Transformer encoder within the VAE architecture for anomaly detection and accompaniment generation in music due to the ViT capabilities of capturing sequential information while acquiring the most important features of the data. These hybrid models are, however, computationally expensive, require a huge amount of data to be trained, and are very hard to fine-tune because of the complexity of the model.

In this work, we are proposing VAeViT, a hybrid Transformer-VAE model for 3D multi-view object classification, to overcome the limitations of ViT such as the necessity for huge datasets that require expensive resources, and the inadequate feature representation by the embedding layer.

3.2 Related Work

Many successful CNN-based 3D multi-view object classification methods have been introduced in the past several years. MVCNN [12] and MVCNN-MultiRes [59] utilize max-pooling with the conventional 2D CNN, retaining the maximal activations from only a specific view while discarding non-maximal elements which potentially leads to loss of vital visual information. Alternatives like sum-pooling have been explored but have not proven more effective. Subsequent works, such as RCPCNN [21] and GVCNN [20] presented innovative strategies for view feature aggregation, organizing views into sets, and conducting pooling within each set. Additionally, Seqviews2seqlabels [60] and 3D2SeqViews [61] introduced RNNs to model view order. However, they are still sensitive to viewpoint variations. While MHBN [62] and MVLADN [63] recognized limitations in view-based pooling, shifting towards set-to-set matching with patch-level pooling. Nevertheless, both view-based and patch-based pooling primarily fuse visual features from different views in the final pooling layer, lacking interactions between visual features from different views in preceding layers. Notably, View-GCN [9] employed Graph Convolution Networks (GCNs) to capture view-based relations and has achieved great results. Yet, constructing an effective graph that represents view-based relations can be challenging since determining which views should be connected in the graph and how to weigh them requires careful consideration and domain knowledge.

On the other hand, RotationNet [11] considers the discrete variance of rotation by taking multi-view images of a 3D object as input and jointly estimates its pose, and the object category. Nonetheless, RotationNet has the limitation that each image should be observed from one of the pre-defined viewpoints. In contrast, Relation Network [64] enhanced each patch feature by considering patches from all views, yielding superior performance compared to prior view-based and patch-based pooling methods but it may struggle to generalize well to new objects or unseen scenes and may be sensitive to variations in the viewpoint, or object appearance. Another very promising new method is CAR-Net [10] which explicitly identifies prospective intra-view and cross-view correspondences through kNN search within the semantic space. The model then integrates shape features from these correspondences through acquired transformations. However, selecting the appropriate k-value in the kNN search is critical, as an improper choice can introduce noisy or irrelevant correspondences, impacting the performance. Although the kNN search scores excellent outcomes towards the intended use, it can be computationally intensive within the semantic space, especially with a large number of views or complex scenes.

MVT [8], the current state-of-the-art, adapts the ViT model for 3D multi-view object representation. It divides each 2D view into multiple patches, performs a low-level computation via the local Transformer layers, and then merges the patches' features from all views in a set and feeds that into a stack of Transformer layers to create a global representation of the object. While MVT has achieved competitive results, there is still room for improvement, especially regarding its dependence on pre-training on large datasets. In this chapter, we introduce VAeViT, an enhanced ViT model that incorporates a preceding VAE. Our model leverages VAE's expertise in feature representation, along with ViT's ability to capture deep features and positional information from multi-views, to comprehensively represent 3D objects.

3.3 Model Architecture

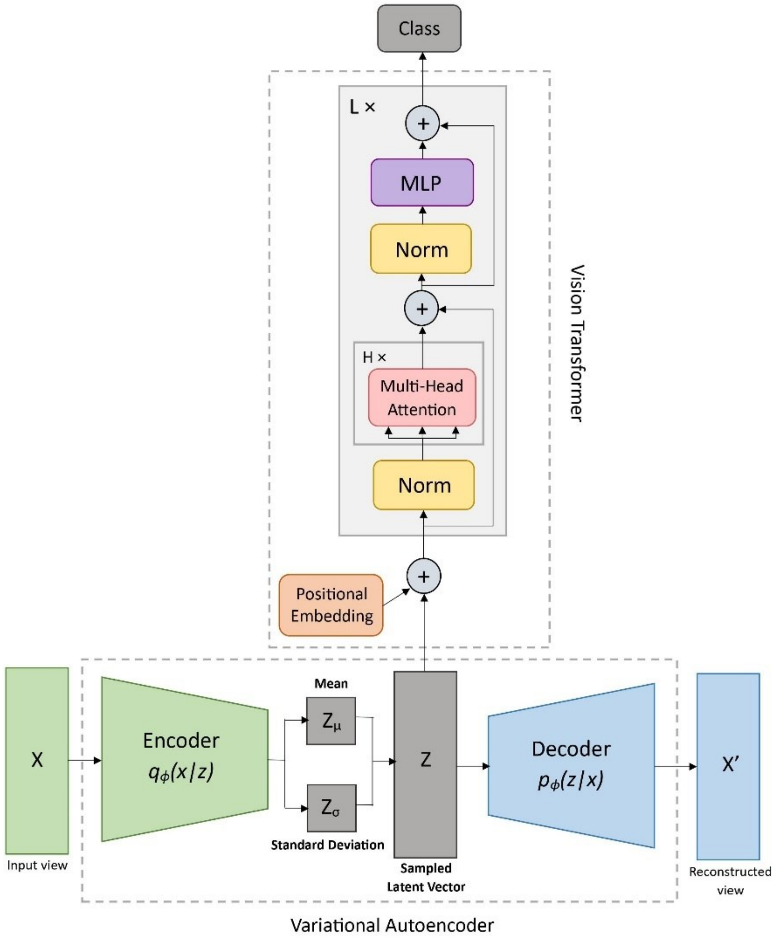


Figure 3.3: VAeViT architecture.

In this section, we introduce the flow and major components of the VAeViT structure illustrated in Figure 3.3. As may be seen, the architecture of VAeViT incorporates the structure of VAE, extracts the learning component of interest as an output, and adapts it to fit in the ViT's input via some tweaks in the interconnection layer. The model takes input $X \in R^{V \times N}$ where X denotes a 2D view from a specific view, V is the number of views per object, and N is the number of objects in the dataset and passes it to the VAE to be condensed into a low-dimension feature representation so that this output gets aggregated with the spatial information in the positional embedding interconnection layer and then gets advanced to the last layer which is ViT where further processing on the data will occur before each object gets classified by the last layers of ViT. The next three subsections will illustrate in detail the methodology of each of those components and its role in constructing a comprehensive 3D object recognition model.

3.3.1 Views Latent Feature Embedding

The first component of the VAeViT model that the input is fed to is a latent feature embedding module which is mainly based on a VAE model. The probabilistic encoder $q_{\phi}(z|x)$ accepts the input views and processes them through all its blocks and outputs two variables which are the mean Z_{μ} and the standard deviation Z_{σ}. The reparameterization trick illustrated in equation (3.1) is then deployed to sample the latent vector Z. Subsequently, the probabilistic decoder $p_{\theta}(x|z)$ takes the sampled vector Z, processes it through its blocks, and tries to reconstruct the original object view. With each iteration, the model learns new weights, enabling it to generate an output X', that closely approximates the original input to the best of the model's ability. The model keeps on backpropagating aiming to minimize the 2 losses in equation (3.4) until optimum results are reached.

Since the VAE model is utilized in this framework to represent the 2D object views in a lower dimension, we are only interested in the sampled vector of each view X from the last epoch. This obtained output represents the embeddings $Z \in R^{V \times N}$ that will be combined together with the positional embeddings in the next phase to represent a global perception of the 3D object before it is fed to the ViT model.

3.3.2 Views Positional Embedding

Positional embeddings as illustrated in Figure 3.2 are basically the spatial information added to the embedded 2D object views to give insights about the order of the views. We use learnable embeddings instead of fixed embeddings as in [24], where we have a number of embeddings as the number of views per object (12 or 20 in our case) in which each number represents the location of a 2D view. These positional embeddings are mapped into an embedding table where each discrete number representing the view order is transformed into a vector of the same size as the latent vectors Z. Each view embedding and its corresponding position embedding are summed up together to form the complete input $G \in R^{V \times N}$ which carries both spatial and semantic features. Positional embeddings play a crucial role by offering contextual information about the input segment being processed. This aids the model in determining not only the object's class but also its pose from any angle.

3.3.3 3D Objects Global Feature Learning

Finally, in order to effectively capture global context information and account for long-range dependencies within complex 3D object views, we integrate Vision Transformers (ViTs) into our approach. Specifically, the Vision Transformer takes the input G, partitions all the views for each object together, and feeds the Transformer encoder with an input of the format $[[G] \times V] \times N$.

The first block of the Transformer encoder is the layer normalization which is a widely used component in Transformer-based architecture for training stability and reducing the training time necessary. The output vectors are fed to the multi-head attention layer to capture the deep semantic representations. The output of the multi-head attention layer is added to the original input $[[G] \times V]$ via a residual connection that helps the network in training by allowing gradients to flow through the networks directly. The output of that goes to another normalization layer for further optimization. The normalized residual output gets fed into a Multi-layer perceptron (MLP) layer that takes input tokens separately and applies a linear transformation to them where embeddings get multiplied by a learned weight matrix and get added to the learned bias vector. A non-linear activation function (GELU in our case) is employed subsequently to allow more complex pattern learning. Adding a dropout layer was proved to improve the performance by as much as 4\% on recognition by propagating representations across layers. Another residual connection is employed and the constructed output of class scores is utilized for classification through a SoftMax function which converts those raw scores into probabilities, indicating the likelihood of each input belonging to which class.

3.4 Experimental Results

In this section, we conduct extensive ablation experiments to learn about the effect of the model's attributes on the overall performance and investigate the effectiveness of the VAeViT model as compared with other state-of-the-art models in recognizing and classifying 3D objects.

3.4.1 Datasets

To assess our model's efficacy, we conducted a series of experiments using VAeViT on two 3D multi-view benchmark datasets, each with two variants based on the object count and the number of captured views. We also examined smaller subsets to evaluate the model's performance on smaller datasets.

- **ModelNet40** [15] is a 3D CAD models dataset that originally contains 12,311 objects from 40 categories. Below are the two assessed variants:
    - 12 views of 3,200 objects resulting in 38,400 images.
    - 20 views of 9,843 objects resulting in 196,860 images.

- **ModelNet10** [15] is a subset of ModelNet40 that contains 4,899 objects from only 10 categories. The two assessed variants are:
    - 12 views of 800 objects resulting in 9,600 images.
    - 20 views of 4,899 objects resulting in 97,980 images.

Gray-scale style is deployed in both datasets to lower the computation cost and speed up the testing. We have followed a split of 80% for training and 20% for testing all through the experiments. In the next 3 subsections, we will only be utilizing the first variant of ModelNet10 since it is the smallest in size and will ease the computation process.

3.4.2 The influence of VAE number of epochs

Table 3.1: Accuracy comparison across different numbers of VAE epochs.

Model variant	50 epochs VAE	100 epochs VAE
VAE	78.0	77.0
VAeViT try1	96.25	98.12
VAeViT try2	98.75	98.12
VAeViT try3	95.0	97.5
VAeViT avg.	96.67	**97.91**

As mentioned previously in section 3.3, VAeViT trains VAE and ViT independently and sequentially. The component that utilizes the most computation resources is the VAE model which is why it is crucial to examine the effect of epochs' number on the overall performance in terms of accuracy and consistency. Table 3.1 shows that both 50 epochs and 100 epochs models have represented the dataset quite similar (from the VAE accuracy). However, when these latent vectors are fed to the ViT model, it can be clearly seen that the 50 epochs model demonstrates an accuracy range of 3.75%, spanning from its lower to upper limits which is quite wide for such a small dataset while the 100 epochs model is much more consistent, and has a higher average accuracy.

3.4.3 Latent Vector dimension effect

Latent vectors are the only common element in both models and they are the only component we are interested in after the VAE training because they will serve the ViT's input. The dimension of latent vectors plays a vital role in the performance as well as the comprehensiveness of the representation. Table 3.2 demonstrates the effect of various vector dimensions on the overall accuracy and illuminates the level of complexity that each model possesses. Based on the experiments, the dimension of 512 is proven the most practical dimension, and it is noticeable that reducing the dimension d_k hurts model quality while increasing it too much makes the training process more sophisticated and consumes more power.

Table 3.2: Comparison of accuracy for different latent dimension d_k.

Model variant	256 d_k	384 d_k	**512 $\boldsymbol{d_k}$**	768 d_k
VAE	75.36	74.4	**77.0**	75.36
VAeViT	97.5	88.13	**98.12**	92.5

3.4.4 ViT model architecture influence

Models' architecture has always been an open area for innovation where developers respond to the application requirements with the most adequate corresponding structure. As for the Vision Transformer, there are a few parameters that can be adjusted which can potentially lead an enhanced accuracy such as; the number of self-attention heads, number of Transformer layers, and MLP size. In this section, we will investigate the effect of each on the overall accuracy, expressiveness, and computation power.

Table 3.3 and Table 3.4 describe the effect of the number of heads, and Transformer layers, respectively, on the model's overall accuracy. For both experiments, we used the same number of ViT epochs. In Table 3.3, we fixed the number of transformer layers to 4 layers and varied the number of attention heads. The lowest obtained accuracy out of all the attention head variations is for the 12 heads model which is simply because the model becomes too condensed and needs an unnecessarily high number of epochs which consumes high power. On the other hand, lowering the number of heads too much as per the case in the 2 heads model, had a low accuracy too but that is because the model has a smaller space of expressiveness. The 4 heads model scored the most optimum results.

Table 3.3: Accuracy comparison of ViT with varying numbers of attention heads.

Model Variant	2 heads	**4 heads**	6 heads	12 heads
VAeViT	95.0	**98.12**	97.5	93.9

Subsequently, we fixed the number of attention heads to 4, as it scored the highest, and experimented with varying the number of transformer layers as shown in Table 3.4. Following the same principle, the 12 layers model scored extremely low accuracy since the model becomes too complicated. The highest performance recorded was for the model of 6 layers.

Table 3.4: Accuracy comparison of ViT with varying numbers of transformer layers.

Model Variant	2 layers	**6 layers**	8 layers	12 layers
VAeViT	95.83	**98.12**	94.79	81.25

Also, Table 3.5 illustrates the effect of the MLP size on the classification accuracy where it is shown that having a large MLP size (as per the case in the 3072 model), is not a good idea since it will confuse the model because it adds on unnecessary nodes to the layer. Although the low number of perception models performed adequately in this case, they might not be sufficient with other applications. That is why having multiple consecutive layers was proven to be more reliable and produce consistent results. The most reasonable size as of our implementation is the 2048,1024 layers model since it has scored the highest.

Table 3.5: Accuracy comparison of ViT with varied MLP sizes..

Model variant	1024	2048	3072	**2048,1024**	3072,2048,1024
VAeViT	97.5	97.5	94.38	**98.12**	97.5

3.4.5 ViT epochs number influence on views number

VAE significantly streamlines the ViT's task by condensing 2D images into lower-dimensional latent vectors. However, for extensive datasets with numerous categories, VAE may not perform optimally, as it requires more time to discern the semantic features for each class. Since VAE takes a much longer time to train than ViT, it's advisable to allocate more epochs to ViT, especially for larger datasets. See Table 3.6 for the impact of ViT's number of epochs on each dataset variant (refer to section 3.4.1), and the influence of increasing the number of views and dataset size on the overall accuracy.

Table 3.6: VAeViT epochs accuracy comparison.

Model variant	ModelNet10		ModelNet40	
	12v	20v	12v	20v
VAE	77.0	95.9	67.0	50.36
VAeViT 100 ViT epochs	**98.12**	99.0	90.31	93.71
VAeViT 200 ViT epochs	98.12	**99.67**	94.22	96.1
VAeViT 300 ViT epochs	-	-	**96.88**	**98.02**

As for the number of views, it can be asserted that the 20-view dataset variants are generally better than the 12-views because they have more information about the object. However, more epochs are required to fully grasp the diverse characteristics of each category in extensive datasets. Also, from the model's perspective, a good size of datasets is $V \times$ any other 3D multi-view object classification model's average dataset size simply because our model sees a 9,600 3D multi-view images dataset of 12 views as $9{,}600 \div 12 = 800$ objects. Of those 800 objects, only 80% are for training which means that VAeViT perceives this 9,600-image dataset as 640 of $[[d_k \times 12]]$ vectors for training.

The least well-represented variant is the largest dataset, which is ModelNet40 of 20 views, but it may be observed that the model's performance is consistently advancing by increasing the number of ViT epochs before it reached a tremendously high accuracy for ModelNet40 of 98.02%. On the other hand, with the ViT epochs increase for ModelNet10 of 12 views, the model suffered from overfitting and resulted in a lower accuracy. In short, the larger the dataset and/or higher the number of views, the more ViT epochs the model will require to achieve good results but the more consistent and higher in accuracy it will be.

3.5 Model Settings

Table 3.7: Settings of the optimum VAeViT model.

VAeViT-small	VAE enc blocks 5	VAE dec blocks 5	VAE epochs 100	Latent vector dimension 512
	ViT heads 4	ViT layers 6	ViT MLP size 2048,1024	ViT epochs 100-300

As stated previously, there is a wide space of adjustments in the model according to the dataset's characteristics and the application requirements. Based on demonstrated experiments, the state-of-the-art VAeViT settings are expressed in Table 3.7. Figure 3.4 and Figure 3.5 represent a sample of the generated images and the confusion matrix respectively after the VAE training only on ModelNet10 of the 12-views variant based on the settings stated in the first row of Table 3.7 which yielded an accuracy of 77.0%. The bar plot in Figure 3.6 illustrates the accuracy enhancement of all the dataset variants when using VAE only and if VAeViT was instead employed.

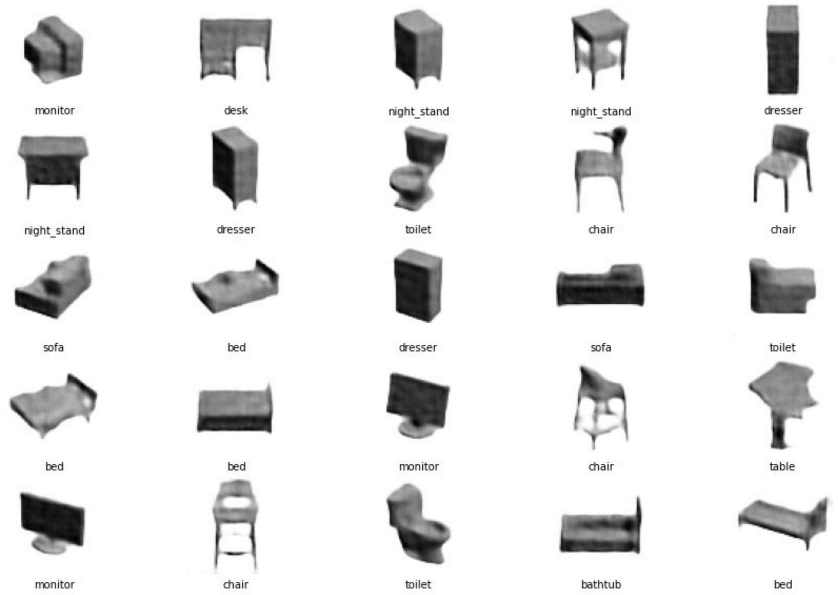
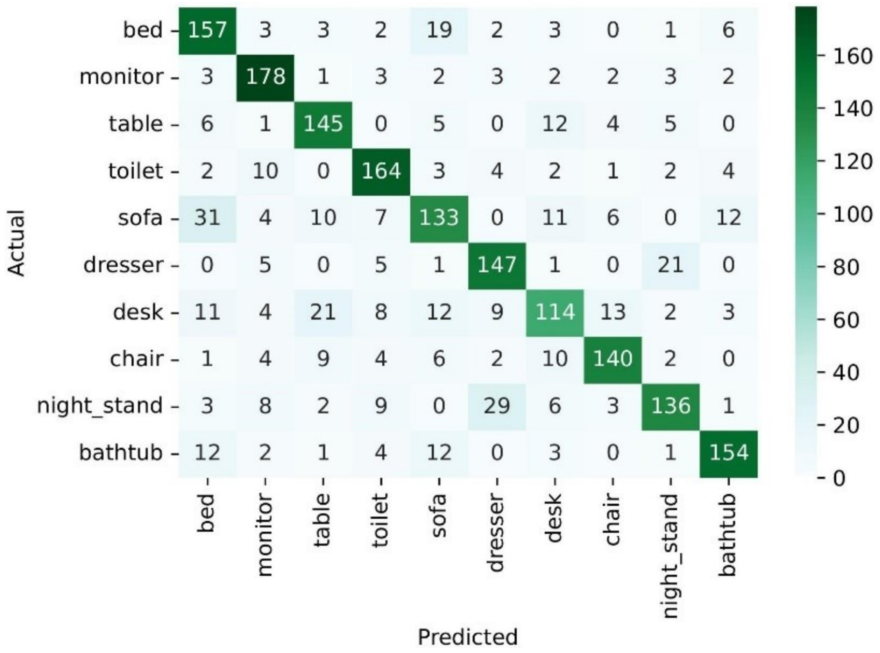


Figure 3.4: Reconstructed sample of ModelNet10 12 views after VAE training only.



	bed	monitor	table	toilet	sofa	dresser	desk	chair	night_stand	bathtub
bed	157	3	3	2	19	2	3	0	1	6
monitor	3	178	1	3	2	3	2	2	3	2
table	6	1	145	0	5	0	12	4	5	0
toilet	2	10	0	164	3	4	2	1	2	4
sofa	31	4	10	7	133	0	11	6	0	12
dresser	0	5	0	5	1	147	1	0	21	0
desk	11	4	21	8	12	9	114	13	2	3
chair	1	4	9	4	6	2	10	140	2	0
night_stand	3	8	2	9	0	29	6	3	136	1
bathtub	12	2	1	4	12	0	3	0	1	154

Figure 3.5: Confusion matrix of ModelNet10 12 views after VAE training only.

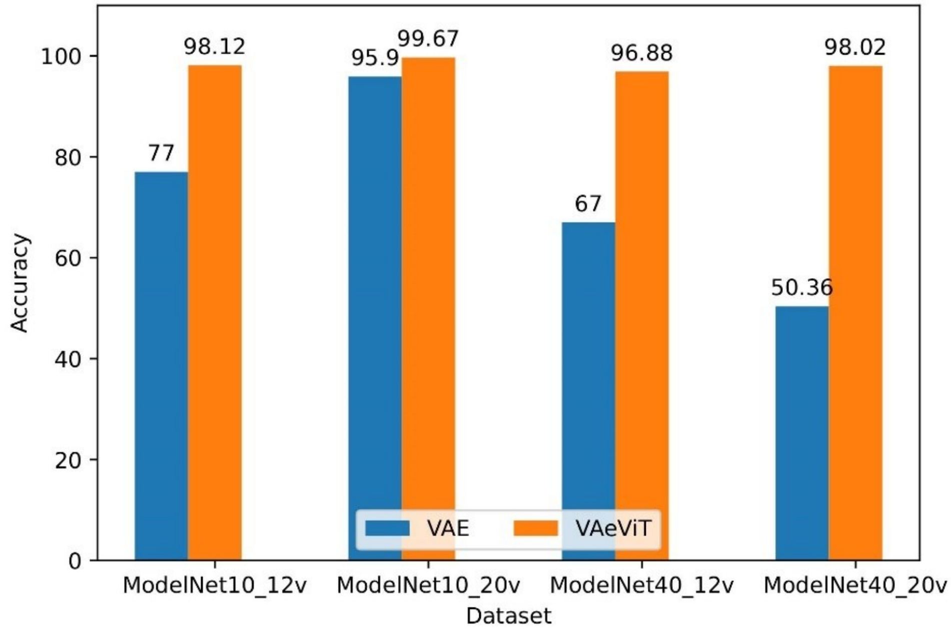


Figure 3.6: Comparison of accuracy between VAE-only and VAE-ViT across all dataset variants.

3.6 Comparison with State-Of-The-Art methods

We conducted a comprehensive comparison between VAeViT and leading models for each 3D object representation type, as detailed in section 3.2 and summarized in Table 3.8.

Table 3.8: 3D object classification accuracy comparison with state-of-the-art models.

Model	Input Modality	ModelNet10	ModelNet40
3DShapeNets [15]	Voxel-based	83.5	77.0
LightNet [65]		93.94	88.93
3D-A-Nets [66]		-	90.5
LP-3DCNN [14]		94.4	92.1
VRN Ensemble [13]		97.14	95.5
PointNet++ [18]	Point cloud	-	91.9
Kd-Networks [67]		94.0	91.8
DeepCCFV [68]		-	92.5
LDGCNN [17]		-	92.9
RS-CNN [16]		-	93.6
MVCNN [12]	12 views	-	89.0
MVCNN [12]	80 views	-	90.1
MVCNN-MultiRes [59]	20 views	-	91.4
GVCNN [20]	3 views	-	93.1
GVCNN [20]	12 views	-	92.6
RCPCNN [21]	12 views	-	93.8
3D2SeqViews [61]	12 views	94.7	93.4
SeqViews2SeqLabels [60]	12 views	94.8	93.4
MHBN [62]	6 views	95.0	94.7
MHBN [62]	12 views	-	93.4
MVLADN [63]	6 views	94.9	94.6
RotationNet [11]	12 views	94.0	91.0
RotationNet [11]	20 views	98.5	97.4
Relation Network [64]	12 views	95.3	94.3
CAR-Net [10]	12 views	95.8	95.2
CAR-Net [10]	20 views	99.0	97.7
MVT [8]	12 views	95.3	94.4
MVT [8]	20 views	99.3	97.5
View-GCN [9]	20 views	-	97.6
VAeViT (Ours)	**12** views	**98.12**	**96.88**
VAeViT (Ours)	**20** views	**99.67**	**98.02**

Voxel-based models generally exhibit lower recognition accuracy compared to their multi-view counterparts, as observed in the first section of the table. While VRN Ensemble [13] outperformed some multi-view methods, this exceptional performance can be attributed to its ensemble approach and the incorporation of a more advanced base model. The second part shows point-based methods, which demonstrate comparable performance to voxel-based models. Most recent approaches that are based on voxels and point clouds tend to achieve similar accuracies, typically falling within the range of 90%-93%. This convergence may be attributed to the inherent complexity or computational demands of these representations.

As for the 3D multi-view types, we have assessed our model against all the pioneering methods in the field where each has its own advantages but suffers from some limitations. Each model utilizes a specific number of views that they excel at according to the approach's methodology. We notice from models such as GVCNN [20] and MHBN [62] that the higher number of views does not always yield a better performance. However, it could be perceived from the state-of-the-art models; RotationNet [11], CAR-Net [10], MVT [8], and View-GCN [9] that 20 view models always perform the highest. Although some of these methods have been pioneering for a few years, VAeViT has the potential to dominate the field after it has demonstrated a competitive performance of 99.67% and 98.02% on the two most well-known datasets; ModelNet10 and ModelNet40 respectively.

Chapter 4

Conclusion

In this thesis, we have introduced two novel 3D multi-view object classification models that are inspired by the NLP methods' significance in extracting the inherent traits within texts and discerning the semantic relationships among sequential data types. The thesis started by demonstrating the advancements of 3D objects in our daily lives and the remarkable limitations encountered by most 3D object recognition methods. Furthermore, we elaborated on the types of 3D object representation and the most utilized methodologies in the field that most models are built based upon.

In chapter 2, we discussed the background of topic models, their working principles, and the most remarkable models in the domain before we introduce our newly designed approach. F-GDA, short for Fully Generalized Dirichlet Allocation model, is a novel Bayesian model that derives all its priors from a Generalized Dirichlet distribution to assure a completely flexible model that is capable of generalizing well. The model is also distinct from other models of a complete generative process that it is simple-to-infer and compatible with multiple applications due to the Gibbs Sampling's unbiased estimates capabilities and its robustness and reliability, especially when dealing with complex, high-dimensional data. A detailed derivation of the core model is demonstrated in section 2.4.2. In further sections of the chapter, we illustrated the preprocessing phases of shaping 3D multi-view objects in the model's compatible input form which is the Bag of Visual Words (BoVWs) format. We proved the robustness of F-GDA and its outperformance against baseline topic models in an ablation study on the N15 images dataset, and showed its capabilities with 3D object classification on the 3D mutli-views dataset, ETH80 where it scored promising results as compared to baseline methods.

In chapter 3, we deviated towards Deep Learning where we presented the working principle of two of the most dominant structures in the field; Variational Autoencoder (VAE) and Vision Transformer (ViT), that is also inspired from the famous NLP architecture, the Transformers. After that we presented the strengths and weaknesses of the current leading paradigms and sequentially VAeViT to overcome those limitations. VAeViT is a pure VAE architecture preceding ViT sequentially where each of them work independently to achieve a certain task. This architectural enhancement builds on the proven effectiveness of VAEs in feature representation and additionally capitalizes on ViT's prominent capability in capturing semantic features from sequential data types. By doing so, we have enhanced the classification of 3D objects by augmenting all the 2D views together via the utilization of the positional embeddings preceding the ViT input and have eliminated the conventional ViT's limitation of necessitating very large dataset in order to perform better by the incorporation of VAE. We conducted extensive amount of experiments on two benchmark 3D multi-view datasets to learn the influence of the model's attributes on the overall performance and illustrated the tremendously high performance of our model against the state-of-the-art models and that is has the potential to dominate the realm of Computer Vision.

For future work, we are planning to incorporate additional flexible priors into our Bayesian model that would allow for a more accurate representation of the underlying data while utilizing less computation time. Also, we would attempt to develop an online learning algorithm to allow our model to handle and learn from continuous data streams in real-time as in [29] which will enhance the performance of the learned model as new data is observed. As for the Deep Learning model, we plan to investigate the effect of our framework on other 3D object representation types such as the point cloud.

List of References

[1] Y. Tang *et al.*, “Recognition and localization methods for vision-based fruit picking robots: A review,” *Front Plant Sci*, vol. 11, p. 510, 2020.

[2] E. Arnold, O. Y. Al-Jarrah, M. Dianati, S. Fallah, D. Oxtoby, and A. Mouzakitis, “A survey on 3d object detection methods for autonomous driving applications,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 10, pp. 3782–3795, 2019.

[3] A. Hatamizadeh *et al.*, “Unetr: Transformers for 3d medical image segmentation,” in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2022, pp. 574–584.

[4] N. Varoquaux, F. Ay, W. S. Noble, and J.-P. Vert, “A statistical approach for inferring the 3D structure of the genome,” *Bioinformatics*, vol. 30, no. 12, pp. i26–i33, 2014.

[5] S. Bourouis, Y. Laalaoui, and N. Bouguila, “Bayesian frameworks for traffic scenes monitoring via view-based 3D cars models recognition,” *Multimed Tools Appl*, vol. 78, pp. 18813–18833, 2019.

[6] D. E. Ruineihart, G. E. Hint, and R. J. Williams, “LEARNING INTERNAL REPRESENTATIONS BERROR PROPAGATION two,” 1985.

[7] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, 1997.

[8] S. Chen, T. Yu, and P. Li, “Mvt: Multi-view vision transformer for 3d object recognition,” *arXiv preprint arXiv:2110.13083*, 2021.

[9] X. Wei, R. Yu, and J. Sun, “View-gcn: View-based graph convolutional network for 3d shape analysis,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1850–1859.

[10] Y. Xu, C. Zheng, R. Xu, Y. Quan, and H. Ling, “Multi-view 3D shape

recognition via correspondence-aware deep learning," *IEEE Transactions on Image Processing*, vol. 30, pp. 5299–5312, 2021.

[11] A. Kanezaki, Y. Matsushita, and Y. Nishida, "Rotationnet: Joint object categorization and pose estimation using multiviews from unsupervised viewpoints," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5010–5019.

[12] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3d shape recognition," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 945–953.

[13] A. Brock, T. Lim, J. M. Ritchie, and N. Weston, "Generative and discriminative voxel modeling with convolutional neural networks," *arXiv preprint arXiv:1608.04236*, 2016.

[14] S. Kumawat and S. Raman, "Lp-3dcnn: Unveiling local phase in 3d convolutional neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4903–4912.

[15] Z. Wu *et al.*, "3d shapenets: A deep representation for volumetric shapes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1912–1920.

[16] Y. Liu, B. Fan, S. Xiang, and C. Pan, "Relation-shape convolutional neural network for point cloud analysis," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 8895–8904.

[17] K. Zhang, M. Hao, J. Wang, C. W. de Silva, and C. Fu, "Linked dynamic graph cnn: Learning on point cloud via linking hierarchical features," *arXiv preprint arXiv:1904.10014*, 2019.

[18] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Adv Neural Inf Process Syst*, vol. 30, 2017.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[20] Y. Feng, Z. Zhang, X. Zhao, R. Ji, and Y. Gao, "Gvcnn: Group-view convolutional neural networks for 3d shape recognition," in *Proceedings*

of the IEEE conference on computer vision and pattern recognition, 2018, pp. 264–272.

[21] C. Wang, M. Pelillo, and K. Siddiqi, "Dominant set clustering and pooling for multi-view 3d object recognition," *arXiv preprint arXiv:1906.01592*, 2019.

[22] J. Yang, Y.-G. Jiang, A. G. Hauptmann, and C.-W. Ngo, "Evaluating bag-of-visual-words representations in scene classification," in *Proceedings of the international workshop on Workshop on multimedia information retrieval*, 2007, pp. 197–206.

[23] N. Bouguila and D. Ziou, "A Dirichlet process mixture of generalized Dirichlet distributions for proportional data modeling," *IEEE Trans Neural Netw*, vol. 21, no. 1, pp. 107–122, 2009.

[24] A. Vaswani *et al.*, "Attention is all you need," *Adv Neural Inf Process Syst*, vol. 30, 2017.

[25] A. Dosovitskiy *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[26] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int J Comput Vis*, vol. 42, pp. 145–175, 2001.

[27] L. Fei-Fei and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2005, pp. 524–531.

[28] B. Leibe and B. Schiele, "Analyzing appearance and contour based methods for object categorization," in *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, 2003, pp. II–409.

[29] W. Fan, H. Sallay, and N. Bouguila, "Online learning of hierarchical Pitman–Yor process mixture of generalized Dirichlet distributions with feature selection," *IEEE Trans Neural Netw Learn Syst*, vol. 28, no. 9, pp. 2048–2061, 2016.

[30] N. Bouguila and W. Fan, *Mixture models and applications*. Springer, 2020.

[31] W. Fan, H. Sallay, N. Bouguila, and S. Bourouis, "Variational learning of hierarchical infinite generalized Dirichlet mixture models and

applications," *Soft Comput.*, vol. 20, no. 3, pp. 979–990, 2016.

[32] E. Epaillard and N. Bouguila, "Variational Bayesian learning of generalized Dirichlet-based hidden Markov models applied to unusual events detection," *IEEE Trans Neural Netw Learn Syst*, vol. 30, no. 4, pp. 1034–1047, 2018.

[33] W. Fan and N. Bouguila, "Variational Learning of Dirichlet Process Mixtures of Generalized Dirichlet Distributions and Its Applications," in *Advanced Data Mining and Applications, 8th International Conference, ADMA 2012, Nanjing, China, December 15-18, 2012. Proceedings*, S. Zhou, S. Zhang, and G. Karypis, Eds., in Lecture Notes in Computer Science, vol. 7713. Springer, 2012, pp. 199–213.

[34] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[35] W. Li and A. McCallum, "Pachinko allocation: DAG-structured mixture models of topic correlations," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 577–584.

[36] D. Blei and J. Lafferty, "Correlated topic models," *Adv Neural Inf Process Syst*, vol. 18, p. 147, 2006.

[37] K. L. Caballero, J. Barajas, and R. Akella, "The generalized dirichlet distribution in enhanced topic detection," in *Proceedings of the 21st ACM international conference on Information and knowledge management*, 2012, pp. 773–782.

[38] A. S. Bakhtiari and N. Bouguila, "A variational Bayes model for count data learning and classification," *Eng Appl Artif Intell*, vol. 35, pp. 176–186, 2014.

[39] K. E. Ihou and N. Bouguila, "A new latent generalized dirichlet allocation model for image classification," in *2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA)*, 2017, pp. 1–6.

[40] N. Bouguila and D. Ziou, "A countably infinite mixture model for clustering and feature selection," *Knowl. Inf. Syst.*, vol. 33, no. 2, pp. 351–370, 2012.

[41] D. Putthividhya, H. T. Attias, and S. Nagarajan, "Independent factor topic models," in *Proceedings of the 26th Annual International*

Conference on Machine Learning, 2009, pp. 833–840.

[42] Y. Teh, D. Newman, and M. Welling, “A collapsed variational Bayesian inference algorithm for latent Dirichlet allocation,” *Adv Neural Inf Process Syst*, vol. 19, 2006.

[43] W. Nie, X. Li, A. Liu, and Y. Su, “3D object retrieval based on Spatial+ LDA model,” *Multimed Tools Appl*, vol. 76, pp. 4091–4104, 2017.

[44] T. P. Minka and J. Lafferty, “Expectation-propogation for the generative aspect model,” *arXiv preprint arXiv:1301.0588*, 2012.

[45] N. Bouguila and W. ElGuebaly, “A generative model for spatial color image databases categorization,” in *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2008, pp. 821–824.

[46] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int J Comput Vis*, vol. 60, pp. 91–110, 2004.

[47] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, “KAZE features,” in *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12*, 2012, pp. 214–227.

[48] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “ORB: An efficient alternative to SIFT or SURF,” in *2011 International conference on computer vision*, 2011, pp. 2564–2571.

[49] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” *arXiv preprint arXiv:1312.6114*, 2013.

[50] D. Rezende and S. Mohamed, “Variational inference with normalizing flows,” in *International conference on machine learning*, 2015, pp. 1530–1538.

[51] I. Higgins *et al.*, “beta-vae: Learning basic visual concepts with a constrained variational framework,” in *International conference on learning representations*, 2016.

[52] J. An and S. Cho, “Variational autoencoder based anomaly detection using reconstruction probability,” *Special lecture on IE*, vol. 2, no. 1, pp. 1–18, 2015.

[53] A. Hawkins-Hooker, F. Depardieu, S. Baur, G. Couairon, A. Chen, and D. Bikard, “Generating functional protein variants with variational

autoencoders," *PLoS Comput Biol*, vol. 17, no. 2, p. e1008736, 2021.

[54] J. Beal, E. Kim, E. Tzeng, D. H. Park, A. Zhai, and D. Kislyuk, "Toward transformer-based object detection," *arXiv preprint arXiv:2012.09958*, 2020.

[55] R. Ranftl, A. Bochkovskiy, and V. Koltun, "Vision transformers for dense prediction," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 12179–12188.

[56] K. Song, X. Liang, and J. Wu, "ViT-based VQ-VAE Generative Network for Accompaniment Generation," in *Proceedings of the 2021 4th International Conference on Algorithms, Computing and Artificial Intelligence*, 2021, pp. 1–5.

[57] B. Choi and J. Jeong, "ViV-Ano: Anomaly detection and localization combining vision transformer and variational autoencoder in the manufacturing process," *Electronics (Basel)*, vol. 11, no. 15, p. 2306, 2022.

[58] T. Chen, B. Li, and J. Zeng, "Learning traces by yourself: Blind image forgery localization via anomaly detection with vit-vae," *IEEE Signal Process Lett*, vol. 30, pp. 150–154, 2023.

[59] C. R. Qi, H. Su, M. Nießner, A. Dai, M. Yan, and L. J. Guibas, "Volumetric and multi-view cnns for object classification on 3d data," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5648–5656.

[60] Z. Han *et al.*, "SeqViews2SeqLabels: Learning 3D global features via aggregating sequential views by RNN with attention," *IEEE Transactions on Image Processing*, vol. 28, no. 2, pp. 658–672, 2018.

[61] Z. Han *et al.*, "3D2SeqViews: Aggregating sequential views for 3D global feature learning by CNN with hierarchical attention aggregation," *IEEE Transactions on Image Processing*, vol. 28, no. 8, pp. 3986–3999, 2019.

[62] T. Yu, J. Meng, and J. Yuan, "Multi-view harmonized bilinear network for 3d object recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 186–194.

[63] T. Yu, J. Meng, M. Yang, and J. Yuan, "3D object representation learning: A set-to-set matching perspective," *IEEE Transactions on Image Processing*, vol. 30, pp. 2168–2179, 2021.

[64] Z. Yang and L. Wang, “Learning relationships for multi-view 3D object recognition,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 7505–7514.

[65] S. Zhi, Y. Liu, X. Li, and Y. Guo, “Toward real-time 3D object recognition: A lightweight volumetric CNN framework using multitask learning,” *Comput Graph*, vol. 71, pp. 199–207, 2018.

[66] M. Ren, L. Niu, and Y. Fang, “3d-a-nets: 3d deep dense descriptor for volumetric shapes with adversarial networks,” *arXiv preprint arXiv:1711.10108*, 2017.

[67] R. Klokov and V. Lempitsky, “Escape from cells: Deep kd-networks for the recognition of 3d point cloud models,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 863–872.

[68] Z. Huang, Z. Zhao, H. Zhou, X. Zhao, and Y. Gao, “Deepccfv: Camera constraint-free multi-view convolutional neural network for 3d object retrieval,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019, pp. 8505–8512.