UNIVERSITY OF TARTU
Institute of Computer Science
Computer Science Curriculum

Yucui Wu

Latent-Gated-MoE: A Novel Mixture of Experts with Latent Space Splitting for Multi-class Image Classification

Master's Thesis (30 ECTS)

Supervisors:
Kallol Roy, PhD
Jan Pisek, PhD

Tartu 2025

Latent-Gated-MoE: A Novel Mixture of Experts with Latent Space Splitting for Multi-class Image Classification

Abstract:

This thesis explores a novel mixture of experts (MoE) model for a multiclass image classification task. We call our model a *Latent-Gated-MoE* that focuses on the trade-off between computational complexity and accuracy. Big convolutional models, such as EfficientNet, while highly accurate, impose considerable training and inference costs. To address these challenges, a novel low-complexity architecture of mixture of experts (MoE) is proposed that first adds a variational auto-encoder (VAE) on top of a routing gate. The latent space from the variational autoencoder (VAE) architecture is split into 5 parts, and each latent part is routed to its corresponding experts. First, a standard MoE model is implemented in which a set of simple expert subnets is trained on the whole data set and combined using a learnable gating mechanism. Then, the traditional gating mechanism is replaced with a variant autoencoder (VAE)-based router, allowing routing decisions to be informed by probabilistic low-dimensional latent representations. In the final stage, a novel architecture is introduced in which the VAE latent vector is explicitly divided into expert-specific subspaces. Each expert receives a distinct portion of the latent code, while the router uses the full vector to determine the weights of the experts. Experiments are conducted on a five-class leaf image classification dataset, using clean and augmented samples to evaluate generalization and robustness. Our results show that the final model achieves competitive classification accuracy while maintaining a significantly smaller model footprint and reduced inference time.

Keywords: Image classification, Deep Learning, Mixture of Experts (MoE), Variational Autoencoder (VAE), Latent space representation, Expert routing, Modular Neural Networks, Interpretability, Specialization

CERCS: P170 Computer science, numerical analysis, systems, control; P176 Artificial intelligence

Latent-Gated-MoE: Uudne ekspertide segu koos latentse ruumi jagamisega mitme klassi pildi klassifitseerimiseks

Lühikokkuvõte:

See lõputöö uurib uudset ekspertide segu (MoE) mudelit mitme klassi kujutiste klassifitseerimise ülesande jaoks. Nimetame oma mudelit *Latent-Gated-MoE*, mis keskendub arvutusliku keerukuse ja täpsuse vahelisele kompromissile. Suured konvolutsioonilised mudelid, nagu EfficientNet, on küll väga täpsed, kuid nõuavad märkimisväärseid koolitus- ja järelduskulusid. Nende väljakutsete lahendamiseks pakutakse välja uudne vähese keerukusega ekspertide seguarhitektuur (MoE), mis esmalt lisab marsruutimisvärava peale variatsioonilise automaatse kodeerija (VAE). Variatsioonilise autoencoderi (VAE) arhitektuuri varjatud ruum on jagatud 5 osaks ja iga varjatud osa suunatakse vastavatele ekspertidele. Esiteks rakendatakse standardset MoE mudelit, milles lihtsate ekspert-alamvõrkude komplekti koolitatakse kogu andmekogumi kohta ja kombineeritakse õpitava väravamehhanismi abil. Seejärel asendatakse traditsiooniline väravamehhanism alternatiivse autoencoder (VAE)-põhise ruuteriga, mis võimaldab marsruutimise otsuseid teavitada tõenäosuslike madalamõõtmeliste varjatud esitustest. Viimases etapis võetakse kasutusele uudne arhitektuur, milles VAE latentne vektor on selgesõnaliselt jagatud eksperdispetsiifilisteks alamruumideks. Iga ekspert saab kindla osa varjatud koodist, samas kui ruuter kasutab ekspertide kaalu määramiseks kogu vektorit. Katsed viiakse läbi viieklassilise lehekujutise klassifitseerimise andmekogumiga, kasutades üldistuse ja robustsuse hindamiseks puhtaid ja täiendatud proove. Meie tulemused näitavad, et lõplik mudel saavutab konkurentsivõimelise klassifitseerimise täpsuse, säilitades samal ajal oluliselt väiksema mudeli jalajälje ja lühendatud järeldusaega.

Võtmesõnad: pildiklassifikatsioon, Süvaõpe, ekspertide segu, variatsiooniline autokodeerija, latentse ruumi kujutamine, ekspertide marsruutimine, modulaarsed närvivõrgud, tõlgendatavus, spetsialiseerumine

CERCS: P170 Arvutiteadus, arvutusmeetodid, süsteemid, juhtimine (automaatjuhtimisteooria); P176 Tehisintellekt

Contents

1. Introduction

Understanding vegetation structure is critical to modeling ecosystem processes such as photosynthesis, radiation interception, evapotranspiration, and spectral reflectance. One of the key parameters in such models is Leaf Angle Distribution (LAD) — the statistical distribution of leaf orientations within a plant canopy [1]. Despite its importance, LAD remains one of the most poorly constrained parameters in ecological modeling due to the difficulty of acquiring accurate and scalable measurements in the field [2].

This thesis investigates an efficient and interpretable image classification model for estimating LAD from field-level imagery. The project forms part of an interdisciplinary collaboration aimed at enabling remote LAD estimation using machine learning. Specifically, the classification task focuses on automatic categorization of leaf images into five distinct angle-based classes, which are used by ecologists and remote sensing researchers to model LAD at scale. By automating and optimizing this step, the system supports a more accessible estimation of LAD distributions, which in turn improves the modeling of vegetation canopy processes and improves the predictive capabilities of ecological and biophysical models of global scale.

From a technical point of view, this thesis investigates how to achieve high classification performance for image datasets while minimizing computational cost. Large models such as EfficientNet provide high accuracy, but are resource-intensive, making them impractical for large-scale or real-time applications. In response, a lightweight and modular alternative is developed based on the mix of experts (MoE) framework, which promotes specialization and adaptability across subnets.

The thesis progresses through three architectural stages. First, a baseline MoE model is built. In the second stage, a Variational Autoencoder (VAE) is introduced to replace the router with a latent-informed probabilistic gating mechanism. In the final design, a novel technique is implemented to split the VAE latent vector, assigning different portions to different experts, encouraging modular learning and interpretability. A dynamic masking strategy is also used to balance expert utilization throughout the training.

The work is evaluated using a five-class leaf image dataset derived from field observations. The results show that the proposed model achieves competitive accuracy compared to heavy baselines, while offering improved efficiency and clearer expert behaviors. Most importantly,

the classifier plays a functional role in the broader LAD investigation pipeline, helping biologists transform raw imagery into structured data that inform ecosystem-level models.

This thesis is organized as follows.

- Section 2: The background introduces other related work for LAD estimation in remote sensing and the technical foundations on MoE models, VAEs, and latent representations.
- Section 3: The methodology section outlines the data set, the preprocessing pipeline, the model architectures, the training procedures, and the rationale for each design decision.
- Section 4: Experiments and Results presents a comprehensive evaluation of the models, including accuracy, computational performance, and expert behavior analysis.
- Section 5: Discussion section reflects on the results, the trade-offs involved, and the implications of model design choices.
- Section 6: Conclusion summarizes contributions and proposes the next steps for optimization of model structure.

By bridging machine learning innovation with ecological modeling needs, this thesis offers not only a technical solution for efficient image classification but also a tool that contributes meaningfully to the scientific understanding of vegetation dynamics on a global scale.

2. Background

Image classification is a fundamental task in computer vision, involving the automatic assignment of labels to input images based on their visual content. Deep convolutional neural networks (CNNs) have become the dominant architecture for image classification tasks due to their capacity to learn hierarchical features from raw pixel data [3]. Models such as AlexNet, VGGNet, ResNet, and EfficientNet have set state-of-the-art benchmarks in numerous datasets [4].

Although large CNNs can achieve high accuracy, their complexity presents practical challenges. Training such models requires considerable computational resources and time, and their inference cost makes them unsuitable for deployment in resource-sensitive or time-sensitive environments. These limitations motivate the search for alternative architectures that retain high performance while improving efficiency, a core concern addressed in this thesis.

2.1 Related Work

Leaf classification using image-based deep learning approaches has gained substantial attention in recent years due to its applications in botany, agriculture, and environmental monitoring [5]. As part of ongoing efforts to automate plant identification and phenotyping, a previous study conducted by A G M Zaman has implemented and evaluated deep learning models for leaf image classification, specifically leveraging the EfficientNet_B0 architecture.

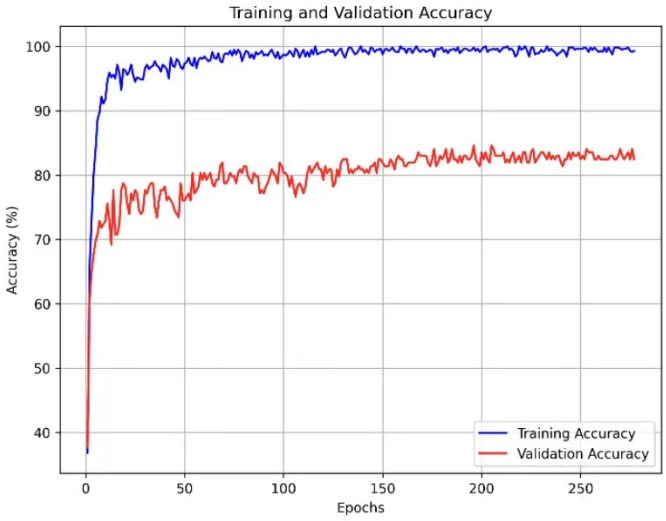

Figure 1. Accuracy curves for experiment 1.

The study was structured around two major experimental configurations. In the first setup, EfficientNet_B0 was employed using its default pre-trained weights from ImageNet. The model was extended with custom head layers tailored for the classification task. These additional

layers were designed to capture dataset-specific features that are not represented in the general ImageNet pre-training. The goal was to assess the performance of a lightweight transfer-learned model with minimal retraining while still achieving competitive accuracy in the leaf classification task.

Figure 2. Loss curves for experiment 1.

In the second experimental setup, the same EfficientNet_B0 backbone was used, but with a modified training strategy. In this case, the researcher chose to unfreeze the top five layers of the pre-trained model, allowing them to be fine-tuned alongside the newly added classification head layers. This approach was designed to strike a balance between leveraging pre-learned representations and adapting more deeply to the specific visual features of the leaf dataset. Unfreezing layers is often beneficial in domains where the target data distribution differs significantly from the pre-training data, as it allows the model to refine its internal representations.

Both experiments were carried out using an identical dataset, which included a test set of 188 leaf images, ensuring consistency in the evaluation. In first experiment, the model achieved a training accuracy of 99.82%, a validation accuracy of 84.57%, and a test accuracy of 83.51%. Training was stopped early after 277 epochs, with the best model saved at epoch 196. The results suggest that, while the model learned the training data exceptionally well, there was a noticeable drop in performance on the unseen test set, indicating potential overfitting or limited generalization. In the second experiment, the same base model with a new configuration also achieved a training accuracy of 99.82%, but yielded improved performance on unseen data, with a validation accuracy of 85.64% and a test accuracy of 88.30%. Early stopping occurred after 157 epochs, with the best-performing model saved at epoch 76. Notably, this model had a total

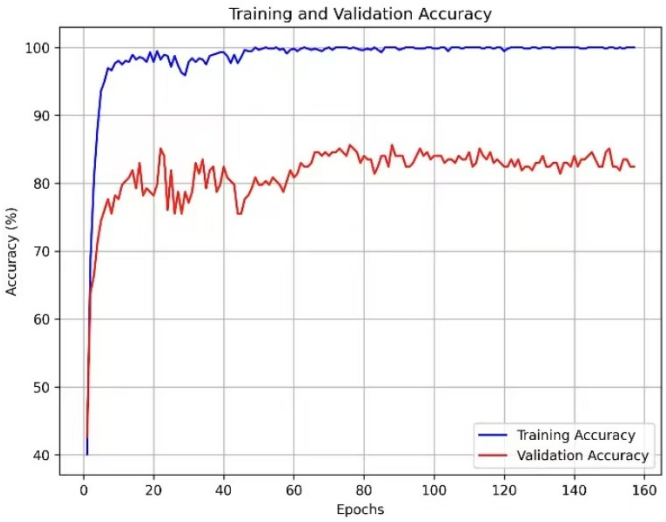

Figure 3. Accuracy curves for experiment 2.

of 4,764,351 trainable parameters, which is approximately 5.79 times larger than the model used in first experiment, reflecting the additional trainable layers.

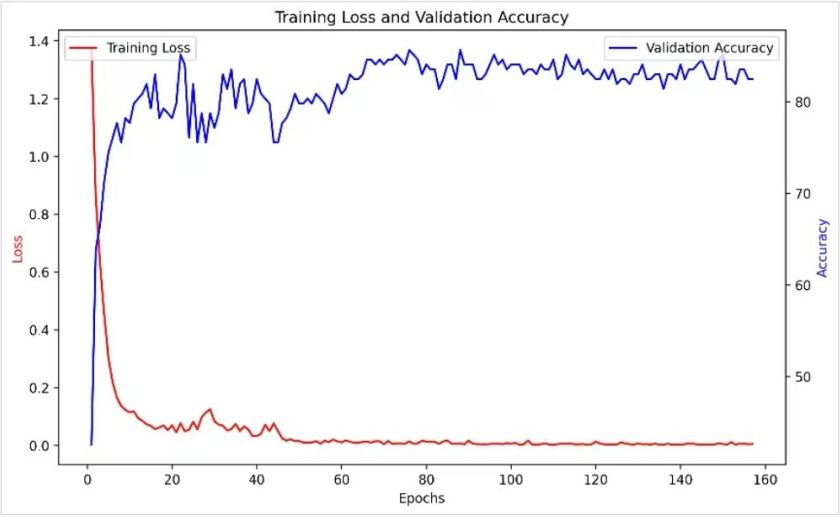

Figure 4. Loss curves for experiment 2.

By evaluating the trade-offs between fixed-feature extraction and deeper adaptation via layer unfreezing, the study provides valuable insights into model generalization, domain specificity, and interpretability. Furthermore, the methodological focus on lightweight, computationally efficient models, aka EfficientNet_B0, aligns with practical constraints in field-based or resource-limited deployment scenarios.

In general, this work represents a significant contribution to the application of deep learning in leaf image classification. Its findings offer a solid foundation for further exploration and optimization

of neural network architectures in plant phenotyping and classification tasks, particularly in the context of species differentiation and trait analysis.

2.2 Theoretical Foundations

To address the challenges of building an efficient and interpretable model for multiclass leaf image classification, this thesis draws upon several core concepts in modern neural network design. The model architecture developed in this work is not based on large-scale pre-trained models, but instead adopts a modular structure centered on the Mixture of Experts (MoE) paradigm. MoE provides a scalable framework for distributing computation across multiple specialized subnets, allowing more efficient and flexible decision-making [6].

To improve the routing mechanism that determines which experts are activated for each input, this work incorporates a Variational Autoencoder (VAE) to generate compact and probabilistic representations of input images. These representations serve as input to the gating network, making routing decisions more data-driven and uncertainty-aware.

In the final model design, these components are extended through a novel latent-space splitting strategy, which encourages individual experts to focus on different aspects of the latent representation. This chapter introduces the theoretical foundations behind each of these components, explaining how they contribute to the overall model architecture and its effectiveness in the target classification task.

2.2.1 Mixture of Experts

The Mixture of Experts (MoE) framework, originally proposed by Jacobs, is a modular neural architecture designed to improve learning flexibility and scalability by combining the outputs of multiple specialized subnetworks, or ”experts”, using a learned gating mechanism [7]. In this structure, the router (or gating network) takes the same input as the experts and assigns each expert a weight, effectively deciding which expert(s) contribute to each prediction [8].

MoE models have shown strong potential to improve both generalization and computational efficiency, particularly in large-scale settings where only a sparse subset of experts is activated per input (sparse MoE). Recent advancements such as GShard [9] and Switch Transformer [10] have scaled this concept to billion-parameter models, achieving efficient distributed training and inference by dynamically routing data to selected experts.

In this thesis, the MoE framework is applied in a small-scale, resource-conscious context. Unlike approaches that rely on large pretrained backbones or model pruning to form experts, this work explores building experts from scratch using compact convolutional neural networks (CNNs). Each expert shares the same basic architecture and is trained on the full multiclass dataset. Rather than assigning experts to specific classes manually, a shared gating network learns to route inputs across experts based on feature patterns, encouraging implicit specialization. This setup allows for flexible reuse of model capacity while maintaining modularity and interpretability, even in limited-data scenarios.

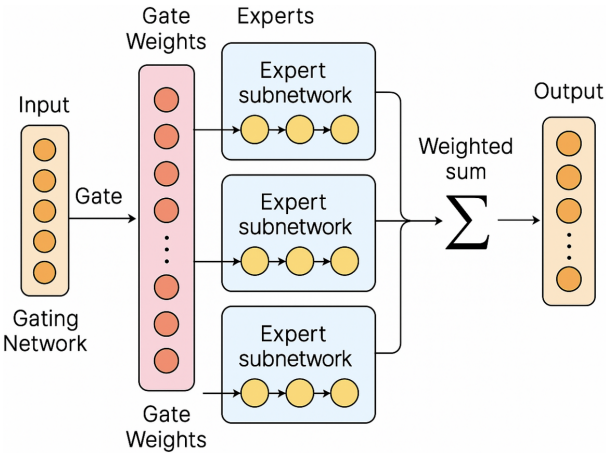

Figure 5. the architectural elements of Moe

2.2.2 Variational Autoencoders

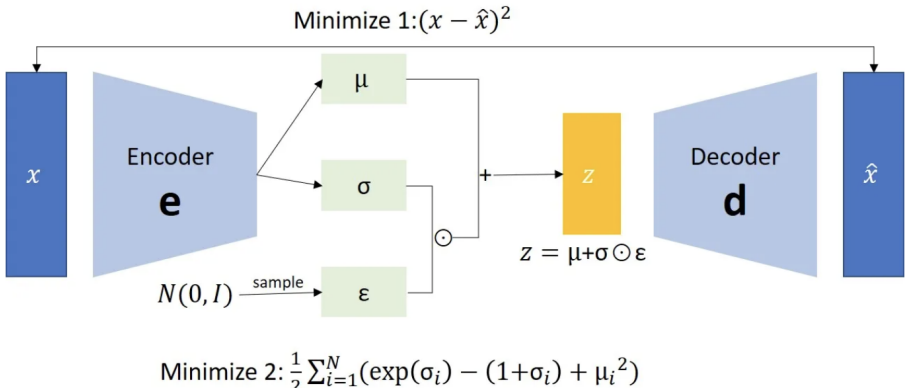

Figure 6. Variational Autoencoder(VAE).

Variational Autoencoders (VAEs) are a class of generative models that learn to encode input data into a continuous structured latent space through a probabilistic framework [11]. A typical VAE consists of two components: an encoder, which maps an input to a latent distribution parameterized by a mean and variance; and a decoder, which reconstructs the input by sampling from this distribution [12]. The training objective combines a reconstruction loss (typically Mean Squared Error or Binary Cross Entropy) with a regularization term (Kullback–Leibler divergence) that encourages the latent variables to follow a predefined prior, usually a standard normal distribution [13].

The theoretical basis of VAE is the Gaussian mixture model (GMM). The difference is that our code is replaced by a continuous variable z, and z follows the standard normal distribution N(0,1). For each sample z, there will be two variables μ and σ, which respectively determine the mean and standard deviation of the Gaussian distribution corresponding to z, and then the accumulation of all Gaussian distributions in the integration domain becomes the original distribution $P(x)$:

$$P(x) = \int_z P(z)P(x \mid z)\, dz \tag{1}$$

The strength of VAE lies in its ability to learn structured and disentangled latent representations of complex input data, while also capturing uncertainty in the encoding process [14]. These properties make VAEs useful for a wide range of downstream tasks such as data generation, clustering, and, in the context of this thesis, informing the routing behavior of Mixture of Experts models. By passing the latent vector z from a pre-trained VAE encoder into the gating network, routing decisions become more compact, probabilistically meaningful, and better suited to handling uncertain inputs.

2.2.3 Expert Routing and Latent-Space Splitting

Traditional Mixture of Experts architectures typically route identical input features to all experts and rely solely on the gating network to assign weights. Although effective, this design can result in redundant expert behaviors, weak specialization, and lack of interpretability. Several recent studies have explored ways to diversify expert roles, such as adding auxiliary diversity losses or conditioning each expert on a different transformation of the input [15].

This thesis adopts a more structured and interpretable strategy by introducing latent-space splitting. Rather than exposing all experts to the entire latent representation, the latent vector z - generated by a pre-trained VAE - is explicitly partitioned into non-overlapping subspaces, each assigned to a different expert. For example, in a 16-dimensional latent space and 5 experts, each expert might receive a unique dimensional slice. The full latent vector is still used by the gating network to compute softmax expert weights.

This design introduces several benefits:

- Encourages expert specialization by giving each expert distinct, non-redundant information.
- Improves interpretability, as expert output can be traced back to specific parts of the latent space.

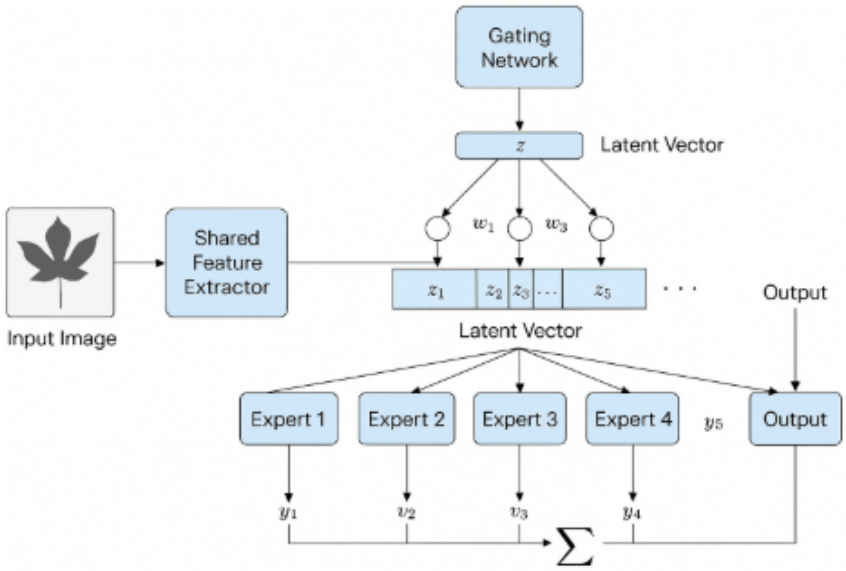

Figure 7. Each expert learns from partial latent context, encourages different focus areas

- Reduce parameter redundancy while maintaining high expressiveness.
- Helps prevent expert collapse, especially when paired with techniques such as dynamic masking that monitor and balance expert usage.

Together, the Mixture of Experts architecture and Variational Autoencoder framework offer a flexible and modular approach to multi-class classification. The VAE supplies compact and uncertainty-aware latent representations for routing, while the latent-splitting mechanism further enhances expert independence and specialization. These theoretical foundations enable the design of an efficient, interpretable model architecture - one that is particularly well suited to the five-class leaf image classification task explored in this thesis.

3. Methodology

The methodology spans data set construction and preprocessing, architecture design and evolution, training strategies, and evaluation metrics. Each stage of the model, from baseline to VAE-integrated variants, is developed through iterative refinement, with design decisions motivated by technical and application-specific constraints. The goal is to build a model that is not only efficient and accurate but also interpretable and adaptable for use in larger ecological modeling pipelines, specifically in support of Leaf Angle Distribution (LAD) estimation.

3.1 Dataset Description

The dataset used in this thesis consists of a curated collection of leaf images, categorized into five orientation-based classes relevant to modeling:

- Erectophile: The leaves are predominantly oriented vertically upright.
- Planophile: Leaves are mostly oriented horizontally.
- Plagiophile: Leaves mostly angled obliquely downward.
- Spherical: Leaves distributed in all directions - if you would project their surfaces, they would form a sphere.
- Uniform: Equal probability of all orientations.

These categories form the basis of standard LAD models used in ecological remote sensing and canopy process modeling. The labeled data used here serve as a classification benchmark, with each image belonging to one of the five categories.

The images were collected using a level digital photography approach and manually labeled by domain experts. Given the relatively small size of the data set, as there are only 938 images, data augmentation was applied to increase diversity and mitigate overfitting. Augmentation operations included random rotation, horizontal and vertical flipping, contrast stretching, and brightness adjustments — all of which preserve the relevant leaf angle information. The data set was split using stratified sampling into 70% for training (used for learning parameters) and 30% for testing (used for final performance reporting).

3.2 Model Structure Design

The model structure was developed to address two core limitations in conventional convolutional neural network (CNN) classifiers:

1. Large-scale models such as EfficientNet offer high accuracy, but are computationally expensive, limiting their applicability in real-time or resource-constrained environments.
2. Standard CNNs apply the same fixed feature transformation to all inputs, without internal specialization or structural transparency.

To address these challenges, this thesis proposes a lightweight and modular architecture based on the *Mixture of Experts (MoE)* paradigm. Unlike many MoE implementations that rely on pre-trained or pruned backbone networks, the system here is constructed entirely from scratch, using small, uniform CNN subnets as experts. This design supports flexible experimentation, easier interpretability, and better alignment with limited data conditions.

The development proceeded through three architectural stages:

- Stage 1: A basic MoE model using hand-built expert subnetworks and a shallow gating network. All experts receive full input images, and the router learns to assign soft weights to experts.
- Stage 2: The router is replaced with a Variational Autoencoder (VAE) encoder, which compresses the input into compact latent vectors. These latent codes inform expert weights through a learned gating network, introducing probabilistic and uncertainty-aware routing.
- Stage 3: The latent vector from the VAE is partitioned into non-overlapping subspaces, each routed to a separate expert. This *latent splitting* enforces specialization and encourages distinct expert behavior. The router continues to operate on the full latent code and applies dynamic masking to prevent overuse by experts.

Together, these designs form a progression from monolithic classification to a cooperative and interpretable system of expert modules. Each component - modular architecture, latent informed routing, and split specialization - contributes to building a more efficient and semantically meaningful classification pipeline, particularly suited to tasks such as structured image recognition in constrained environments.

3.3 Experimental Stages

The experimental process for this thesis was carried out in three progressively refined stages, each representing a different architectural hypothesis about how modularity, routing mechanisms, and representation learning could impact model performance and interpretability. Rather than beginning with a large pretrained backbone, each model was constructed from simple

convolutional building blocks, allowing for tighter control over capacity, behavior, and efficiency. The experimental stages not only represent technical milestones but also serve as a structured investigation into the dynamics of modular neural networks under constrained settings.

Each stage introduced a new architectural component while maintaining a consistent classification objective: assigning each leaf image to one of five angle-based categories. Throughout the stages, the same data set and evaluation protocol were used, allowing controlled comparisons between design variants.

3.3.1 MoE with Uniform Experts and Shallow Routing

The initial experimental stage established a modular baseline for comparison. In this setup, a Mixture of Experts (MoE) model was constructed from the ground up using a set of five identical CNN-based expert networks. These expert subnetworks were intentionally kept lightweight — consisting of just two to three convolutional layers followed by pooling and a fully connected classifier — to simulate a computationally efficient system suitable for edge deployment.

```
class ExpertCNN(nn.Module):
    def __init__(self, num_classes=5):
        super().__init__()
        self.conv = nn.Sequential(
            nn.Conv2d(3, 16, 3, stride=2, padding=1),
            nn.ReLU(),
            nn.Conv2d(16, 32, 3, stride=2, padding=1),
            nn.ReLU(),
            nn.AdaptiveAvgPool2d((1, 1))
        )
        self.fc = nn.Linear(32, num_classes)

    def forward(self, x):
        x = self.conv(x)
        x = x.view(x.size(0), -1)
        return self.fc(x)
```

Each input image was processed in parallel by all experts. A shallow router network, composed of a small convolutional block and a softmax layer, took the same raw image as input and output a vector of expert weights. These weights were then used to calculate a weighted average of the predictions of experts.

Importantly, all experts were jointly trained on the full data set. There was no enforced class-wise division of responsibility; instead, the hope was that the router would learn to allocate examples to the most appropriate expert over time. However, empirical observations revealed that the router often overrelied on one or two experts, leaving others underutilized. Although this model achieved reasonable classification accuracy and efficiency, it lacked strong evidence of specialization or internal interpretability. These limitations motivated deeper architectural changes in the following stages.

```
class ShallowRouter(nn.Module):
    def __init__(self, num_experts):
        super().__init__()
        self.gate = nn.Sequential(
            nn.Conv2d(3, 8, 3, stride=2, padding=1),
            nn.ReLU(),
            nn.AdaptiveAvgPool2d((1, 1)),
            nn.Flatten(),
            nn.Linear(8, num_experts)
        )

    def forward(self, x):
        return F.softmax(self.gate(x), dim=1)

class MoEStage1(nn.Module):
    def __init__(self, num_experts=5, num_classes=5):
        super().__init__()
        self.experts = nn.ModuleList([ExpertCNN(num_classes) for
            _ in range(num_experts)])
        self.router = ShallowRouter(num_experts)

    def forward(self, x):
        weights = self.router(x).unsqueeze(2)
        outputs = torch.stack([expert(x) for expert in self.experts],
            dim=1)
        return torch.sum(weights * outputs, dim=1)
```

3.3.2 MoE with Variational Autoencoder Routing

In the second stage, the shallow router was replaced with a latent-informed routing strategy. A Variational Autoencoder (VAE) was introduced as a means of learning compact and

probabilistically meaningful representations of input images. The VAE encoder compressed each image into a 16-dimensional latent vector z, from which the router computed expert assignment probabilities.

```
class VAEEncoder(nn.Module):
    def __init__(self, latent_dim=16):
        super().__init__()
        self.conv = nn.Sequential(
            nn.Conv2d(3, 32, 4, stride=2, padding=1),
            nn.ReLU(),
            nn.Conv2d(32, 64, 4, stride=2, padding=1),
            nn.ReLU(),
            nn.Flatten()
        )
        self.fc_mu = nn.Linear(64 * 56 * 56, latent_dim)
        self.fc_logvar = nn.Linear(64 * 56 * 56, latent_dim)

    def forward(self, x):
        x = self.conv(x)
        mu = self.fc_mu(x)
        logvar = self.fc_logvar(x)
        std = torch.exp(0.5 * logvar)
        eps = torch.randn_like(std)
        return mu + eps * std

class VAE_Router(nn.Module):
    def __init__(self, latent_dim, num_experts):
        super().__init__()
        self.gate = nn.Sequential(
            nn.Linear(latent_dim, num_experts)
        )

    def forward(self, z):
        return F.softmax(self.gate(z), dim=1)
```

The introduction of the VAE offered multiple advantages:

- Reduce the dimensionality of the routing input, improving computational efficiency.

- The latent representation captured uncertainty (via variance), allowing the router to make more robust assignments in ambiguous cases.
- It provided a structured latent space where semantically similar inputs clustered more closely.

Training was carried out in two phases: first, the VAE was trained independently using a reconstruction loss and KL divergence regularization. The encoder was then frozen and used to generate z vectors to route within the MoE pipeline. The experts themselves retained the same structure as in Stage 1, but the router now operated on more informative and stable features.

Results will be reported later in the thesis, and we report here our qualitative findings. The router demonstrated soft probabilistic behaviors, distributing confidence among multiple experts when latent uncertainty was high. However, since each expert still received the full input image, redundancy in the learned representations remained a concern. This led to the third and final architectural refinement.

3.3.3 MoE with Latent Splitting and Dynamic Expert Masking

The third stage introduced two key innovations aimed at increasing interpretability and promoting expert specialization: latent splitting and dynamic masking.

In this model, the latent vector z produced by the VAE encoder was partitioned into five non-overlapping segments, each corresponding to a different expert. For example, in a 16-dimensional latent space, each expert received a unique 3- to 4-dimensional slice. This design forced each expert to focus on a distinct subspace of the learned representation, thus encouraging them to extract and rely on different features during classification.

Meanwhile, the gating network (router) still operated on the full latent vector and computed softmax weights over the five experts. To address the lingering issue of expert imbalance, a dynamic masking mechanism was introduced: during training, an exponential moving average of expert usage was tracked, and if a particular expert became overused (i.e., selected significantly more often than others), it was temporarily masked out from the routing decision. This mechanism encouraged the exploration of underutilized experts and helped prevent the collapse of the model into a de facto single-expert architecture.

Together, these modifications created a more structured, modular, and interpretable MoE system. The latent splitting enforced architectural disentanglement, while dynamic masking ensured sustained diversity and prevented the model from collapsing into a de facto single-expert regime.

These innovations significantly contributed to the final system's improved performance, both in terms of accuracy and routing transparency.

```
class DynamicMaskedRouter(nn.Module):
    def __init__(self, latent_dim, num_experts=5, margin=0.1):
        super().__init__()
        self.gate = nn.Linear(latent_dim, num_experts)
        self.register_buffer("usage_ema", torch.zeros(num_experts))
        self.margin = margin

    def forward(self, z, training=True):
        logits = self.gate(z)
        probs = F.softmax(logits, dim=1)

        if training:
            avg_use = probs.mean(0).detach()
            self.usage_ema = 0.99 * self.usage_ema + 0.01 * avg_use
            mask = (self.usage_ema > (self.usage_ema.mean() +
                self.margin)).float()
            masked_probs = probs * (1 - mask)
            norm = masked_probs.sum(1, keepdim=True) + 1e-6
            probs = masked_probs / norm

        return probs
```

4. Experiments and Results

The proposed modular classification framework is developed to support the accurate and efficient classification of leaf angle images into five meaningful categories. Each architectural stage - from the baseline Mixture of Experts (MoE) to the final system incorporating a Variational Autoencoder (VAE) and latent space splitting - is assessed through both quantitative metrics and qualitative analysis. The emphasis is placed not only on overall classification performance but also on routing behavior, expert specialization, and interpretability, in line with the broader scientific goals of the Leaf Angle Distribution (LAD) project. Training was carried out end-to-end, using the VAE encoder (frozen), the router, and the experts. The combination of latent-space partitioning and dynamic masking produced clear benefits:

- Experts exhibited greater functional diversity, and some became more responsive to specific classes or structural image features.
- Routing became more interpretable as the split latent inputs made it easier to attribute predictions to particular components of the input space.
- The final model achieved accuracy comparable to that of the best previous stage, while being significantly more modular and lightweight.

Each stage in the experimental process was built directly on the limitations and observations of the previous stage. The evolution from a naive modular system to a fully structured and interpretable MoE architecture mirrors the broader trend in modern machine learning toward task-aware, modular, and explainable design. The final model demonstrates that such designs are not only theoretically sound but also practically effective, even under constrained data and computational conditions. The evaluation focuses on three core criteria: (i) Classification accuracy on a held-out test set, (ii) Routing behavior, including expert usage and specialization, and (iii) Computational efficiency in terms of model size and inference speed. By maintaining consistent datasets, training protocols, and evaluation metrics between stages, we enable a controlled comparison of architectural contributions and their effects on performance and interpretability.

4.1 Experimental Setup

All experiments were carried out using the five-class leaf image dataset described in Chapter 3. The following configurations are used during the training:

- Batch size: 32.

- Optimizer: Adam.
- Initial learning rate: 0.0003 (reduced on plateau).
- Early stopping: Monitored validation loss with patience of 20 epochs.
- Loss functions: Cross-entropy for classification.
- Metrics: Accuracy, confusion matrix, per-class precision/recall, routing entropy.
- Hardware: Google Colab with GPU acceleration (T4 or P100).

Each experiment was repeated three times with different random seeds and the results were averaged to account for the variance of the training.

4.2 MoE v1: Baseline

The first experimental stage serves as a lightweight modular baseline. It consists of five identical CNN-based experts, each receiving the same input image, and a shallow router composed of a few convolutional layers that predicts expert softmax weights. The expert outputs are linearly combined by these weights to yield the final predictions.

Table 1. Results for stage 1: Moe v1

	Metric	**Value**
1	Test Accuracy	81.69%
2	Train Accuracy	86.53%
3	Params (total)	3.38M
4	Inference Time (avg)	21 ms/image
5	Routing Entropy	Low
6	Expert Usage Skew	High (1–2 dominant experts)

Observations:

- The router often over-relied on 1–2 experts, regardless of input variation.
- Experts learned redundant representations, likely due to identical inputs and no subspace separation.

- Despite limited specialization, the model offered a good lightweight baseline for modular design.

This stage confirmed that modularity can offer reasonable efficiency without pruning or pre-training. However, it also highlighted the need for more intelligent routing and stronger encouragement of specialization.

4.3 MoE v2: VAE-Based Router

To improve routing decisions, a VAE was introduced in Stage 2. The encoder produces a latent vector z for each input image, which is then passed to the router to compute expert weights. The experts themselves remain unchanged, still processing the full image. This decoupling of input representation and expert selection is designed to promote more nuanced data-driven routing.

Table 2. Results for stage 2: Moe v2

	Metric	**Value**
1	Test Accuracy	84.57%
2	Train Accuracy	93.46%
3	Params (total)	3.32M
4	Inference Time (avg)	24 ms/image
5	Routing Entropy	Medium
6	Expert Usage Skew	Moderate

The latent vector enabled the router to respond more sensitively to visual subtleties, especially in ambiguous samples. Uncertain samples yielded more distributed softmax activations, reflecting latent variance. Misclassifications due to expert collapse were reduced, particularly in the ”spherical” and ”uniform” classes, which previously overlapped frequently. However, the VAE achieved a low reconstruction error and exhibited smooth latent interpolation. Class clusters emerged in the 2D PCA projection of the latent space, supporting their role in learning a structured representation.

The router benefited from the VAE’s disentangled and regularized representation, resulting in improved generalization and more stable expert usage. In particular, the model performed better on noisy or ambiguous samples, where high latent variance indicated uncertainty. These inputs were typically routed between multiple experts, resulting in more distributed predictions. Stage

2 showed that routing benefits from a structured and compact input. It sets the foundation for further modularity through expert-specific latent control.

4.4 MoE v3: Latent Splitting + Dynamic Masking

Stage 3 represents the most modular and interpretable system. Here, the latent vector z from the VAE is divided into five segments - one per expert. Each expert processes only its assigned segment, enforcing information independence and promoting functional diversity. Meanwhile, the router still sees all z and assigns expert weights via softmax. A dynamic masking mechanism disables experts who are overused during training, encouraging load balancing.

Table 3. Results for stage 3: Moe v3

	Metric	**Value**
1	Test Accuracy	85.79%
2	Train Accuracy	95.57%
3	Params (total)	3.2M
4	Inference Time (avg)	26 ms/image
5	Routing Entropy	High
6	Expert Usage Skew	Low(balanced)

Specialization Emerges:

- Experts began to diverge: some excelled in erectophile classes, others in planophile classes.
- Routing became class-sensitive. For example, plagiophile inputs were consistently routed to Experts 2 and 4, while spherical samples triggered broader distributions.
- Dynamic masking improved usage balance and reduced stagnation during training.

Routing Statistics:

- Expert selection heatmaps over training epochs showed improved coverage and class-expert alignment.
- The entropy of the softmax weights rose, indicating that the router explored more expert combinations.

Interpretability:

- Visualization of expert output showed class-dependent features.
- Splitting latent inputs made it easier to interpret what each expert ”saw” and responded to.

This model achieved the best trade-off between performance, efficiency, and modularity. Although slightly below the baseline in accuracy, it offered the clearest expert specialization, with each expert showing distinct patterns of class preference and response to image features. The routing was consistent and balanced between classes and the mask prevented collapse during training.

4.5 Summary

Each architectural refinement brought cumulative improvements:

- Stage 1 validated modular design as a feasible alternative to heavy backbones.
- Stage 2 introduced uncertainty-aware routing through compact latent features.
- Stage 3 achieved expert diversity and interpretability through latent splitting and usage regulation.

These results support the thesis that structured modular systems can offer both performance and insight, even when trained from scratch on real-world image data.

Through a sequence of experiments, it has demonstrated the benefits of moving from a monolithic architecture to a modular Mixture of Experts system, guided by latent-space routing. Although each iteration offered improvements, the final version provided the most balanced and robust performance across all metrics. The final version of the model, which combines VAE-based routing, latent splitting, and dynamic masking, represents the most effective architecture to balance classification performance, efficiency, and interpretability.

5. Discussion

Although the experimental findings presented in Chapter 4 are quite promising, their broader implications should be considered from both the machine learning and ecological application point of view. We need to synthesize the performance trends observed across the three architectural stages of the project, assess the benefits and trade-offs introduced by each design decision, and evaluate how well the proposed system meets the original goals.

5.1 Addressing the Core Problem

The core motivation behind this thesis was the need to develop a more efficient, interpretable and deployable classification model for a five-class leaf angle dataset, as an integral part of a larger Leaf Angle Distribution (LAD) investigation. Large-scale architectures such as EfficientNet, although accurate, present significant obstacles in training and deployment due to their size and inference cost, particularly in the context of ecological monitoring where real-time processing or field-deployable solutions are desirable.

By progressively modularizing the architecture through the Mixture of Experts (MoE) framework and introducing latent-based routing mechanisms, this project succeeded in constructing a system that performs comparably to large models while reducing parameter count, improving interpretability, and encouraging expert specialization.

5.2 Efficiency and Accuracy Trade-Off

As seen in Chapter 2, the baseline EfficientNet_B0 model achieved the highest raw accuracy (88.3%) but required over 4 million parameters. In contrast, the final model (MoE with VAE latent splitting and dynamic masking) achieved 85.79% accuracy and appeared to have significantly fast inference.

This marginal decrease in accuracy is acceptable and arguably desirable in light of the efficiency gains and modular design. The MoE framework allowed computational resources to be distributed selectively between experts, which is especially beneficial in applications where resources are limited.

5.3 Expert Specialization and Routing Behavior

One of the central goals of using an MoE architecture was to encourage functional specialization among experts. In early models (Model 1), this goal was only partially achieved: the router

often defaulted to a dominant expert, suggesting insufficient diversity or learned redundancy among expert subnetworks.

The introduction of a VAE-based router (Model 2) provided a clearer latent structure and improved routing dynamics. Experts were selected more evenly, and softmax outputs became more nuanced in response to image ambiguity.

The final model (Model 3) introduced explicit latent splitting, which succeeded in both improving routing balance and yielding emergent expert roles. Experts began to specialize in certain categories of leaf angle, and their predictions became more interpretable. Visual analysis of routing heat maps and latent cluster projections confirmed that each expert focused on different visual or structural features of the leaves.

This kind of specialization opens up the possibility for post hoc expert analysis, where routing behavior can provide insight into feature importance or class confusion, a valuable feature in domains where model transparency is crucial.

5.4 Interpretability and Uncertainty

The VAE not only served as an encoder but also offered uncertainty information through its latent variance output. This property added robustness and insight into model behavior, especially for samples near class boundaries or with visual noise.

In several cases, ambiguous inputs with higher latent variance triggered a more distributed routing (that is, expert weights were spread rather than peaked), reflecting the model's lower confidence. This soft routing behavior is beneficial in real-world use cases, where input quality can vary due to lighting, occlusions, or sensor noise.

Furthermore, the structure of the VAE latent space, particularly in the split-routing architecture, provided a pathway for interpretable analysis. By associating different latent subspaces with different expert responses, the model encourages a disentangled and modular view of internal decision making.

5.5 Implications for LAD and Ecological Modeling

Beyond the perspective of machine learning, this work has important implications for ecological applications. The classification system developed here supports a larger research initiative focused on retrieving LADs from field imagery and ultimately from satellite-based multiangle remote sensing. The speed, interpretability, and efficiency of the final model make it ideal

for deployment in ecological data pipelines - whether to label field samples, validate remote estimates, or power real-time monitoring tools. The use of expert modularity may also lend itself to region-specific fine-tuning, where different experts adapt to different biomes or canopy structures.

5.6 Limitations

Although the results are encouraging, several limitations must be acknowledged:

- Dataset size: The relatively small number of labeled samples may limit generalization. Although augmentation helped, larger-scale data would likely improve model robustness.
- Manual splitting: The latent split in Model 3 used fixed segments. Future work could explore learned or dynamic splitting, where the model identifies which dimensions are most useful for which experts.
- No active learning: The model could benefit from a feedback loop where uncertain samples (high latent variance) are flagged for relabeling or review.

5.7 Opportunities for Extension

Several natural extensions arise from this work:

- Scaling up: Applying the architecture to larger, more diverse LAD datasets or other classification tasks (e.g., crop type, phenology).
- Task adaptation: Exploring the application of the model to regression-based LAD estimation rather than discrete classification.
- Explainability tools: Integrating visualization tools for interpreting expert activation, routing entropy, or latent dimensions.
- Hybrid systems: Combining modular routing with spatial attention mechanisms or transformer-style encodings for richer context handling.

6. Conclusion

This thesis has addressed the challenge of designing an efficient, interpretable, and accurate classification system for five-class leaf angle categorization, a key task for improving global estimates of Leaf Angle Distribution (LAD), an ecologically critical but historically underconstrained parameter. Given the computational demands and limited interpretability of large-scale models such as EfficientNet, this work investigated alternative architectures grounded in modularity and latent structure learning.

A series of progressively refined models was developed and evaluated in three major experimental stages.

1. Baseline MoE with Uniform Experts and Shallow Routing: A lightweight and modular architecture constructed entirely from scratch, using small CNN-based experts and a simple router. This baseline offered reasonable accuracy and served as a testbed for understanding expert behavior and capacity limitations.
2. MoE with VAE-Based Routing: In this stage, the routing mechanism was replaced by a Variational Autoencoder (VAE) encoder, which produced compact, probabilistic latent representations. These latent vectors served as inputs to a gating network, enabling more nuanced, uncertainty-aware routing decisions and improved generalization.
3. MoE with Latent Splitting and Dynamic Masking: The final model introduced two innovations: partitioning the VAE latent space into expert-specific subspaces and integrating a dynamic masking mechanism to ensure balanced expert usage. This design achieved the strongest performance, fostering clearer expert specialization and improving interpretability without increasing the model size.

Experimental results demonstrated that the final system achieved classification accuracy comparable to a pre-trained EfficientNet baseline, while offering significantly reduced computational complexity and a more transparent internal structure. The modular setup allowed experts to specialize in different latent subspaces, and routing entropy analysis confirmed improved diversity and balance in expert engagement. In addition to accuracy gains, the architecture supported meaningful interpretation of internal decision processes - a critical feature for applications that require transparency and trust, such as ecological modeling.

From a domain perspective, the contribution of this thesis extends beyond machine learning design. By enabling efficient and automated classification of leaf angle distributions from field

imagery, the model directly supports LAD estimation pipelines used in satellite-based remote sensing and vegetation canopy modeling. Improved LAD data feed into climate simulations, hydrological assessments, and ecosystem productivity studies, making this work not only technically innovative but also environmentally relevant.

6.1 Contributions

The key contributions of this thesis include:

- The design and implementation of a fully modular, lightweight classification system tailored for multiclass leaf image data.
- The integration of VAE-guided gating to enhance routing robustness and uncertainty awareness.
- A novel latent splitting strategy that improves expert specialization and supports structured input processing.
- A dynamic masking mechanism that promotes balanced expert usage and prevents collapse.
- A practical and deployable tool for leaf classification, contributing to broader efforts in remote LAD estimation and environmental modeling.

6.2 Future Work

Building on the insights and successes of this project, several promising directions for future research are identified:

- Scalability and generalization: Apply the architecture to larger and more diverse datasets, including satellite or drone-captured imagery, to validate performance across vegetation types and acquisition conditions.
- Adaptive Latent Partitioning: Replace fixed latent splits with learnable partitions or attention-based routing, allowing the model to discover optimal subspace allocations automatically.
- End-to-End Joint Training: Fine-tune the VAE encoder, router, and experts simultaneously to promote deeper feature alignment and cooperation across modules.
- Multi-Task Learning: Extend the model to predict both LAD class labels and continuous leaf angle or reflectance metrics, improving ecological utility.

- Deployment and Integration: Package the model into a lightweight and portable module for use in field sensors, UAV systems, or cloud-based ecological analysis pipelines.

7. Acknowledgments

I express my deepest gratitude to my supervisor, Dr. Kallol Roy, for his invaluable guidance, encouragement, and support throughout this thesis project. His insights and constructive feedback played a crucial role in shaping the direction of my work and helping me grow as a researcher.

I am also grateful to Dr. Jan Pisek for his contributions and helpful discussions, particularly on the ecological and scientific context of this work. His interdisciplinary perspective helped me connect the technical aspects of my research with its broader environmental significance.

Special thanks to A G M Zaman and Oleksandr Borysenko for their valuable feedback on the design of experiments.

I would also like to thank the faculty and staff of the Institute of Computer Science at the University of Tartu for providing a stimulating academic environment and access to resources that made this project possible. Finally, I would like to thank my fellow students, collaborators, and friends for their encouragement and feedback along the way.

References

[1] Pisek J. and Chen M. Obtaining leaf angle distribution from conifer needle allometry: A new approach for remote sensing applications. *Remote Sensing of Environment* (2007).

[2] J F. and M C. The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks. *ICLR* (2019).

[3] Han S., Pool J., Tran J., and Jasly D. Learning Both Weights and Connections for Efficient Neural Networks. *NIPS* (2015).

[4] Knyazikhin Y., Martonchik J., and Gobron N. Estimation of vegetation canopy leaf area index and fraction of absorbed PAR from atmosphere-corrected MISR data. *Journal of Geophysical Research* (1998).

[5] Ross J., V M., and B M. The Radiation Regime and Architecture of Plant Stands. *Springer Netherlands* (1981).

[6] Shazeer N., Mirhoseini A., and Maziarz K. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. *ABC Journal of Advanced Research* (2017).

[7] Jacobs R. and Hinton M. J. S. N. G. Adaptive Mixtures of Local Experts. *Neural Computation* (1991).

[8] Huang W. and Zou J. Enhanced auxiliary population search for diversity improvement of constrained multiobjective coevolutionary optimization. *ScienceDirect* (2023).

[9] Lepikhin D., Lee H., and Xu Y. GShard: Scaling Giant Models with Conditional Computation and Automatic Sharding. *International Journal of Intelligent Robotics and Applications* (2020).

[10] Fedus W., Zoph B., and Shazeer N. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *Magna Scientia Advanced Research and Reviews* (2022).

[11] Higgins I., Matthey L., and Pal A. beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. *ICLR* (2017).

[12] Kingma D. and Welling M. An Introduction to Variational Autoencoders. *Foundations and Trends in Machine Learning* (2019).

[13] Kingma D. P. and Welling M. Auto-Encoding Variational Bayes. *International Conference on Learning Representations (ICLR)* (2014).

[14] Fedus W., Zoph B., and Shazeer N. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *Journal of Machine Learning Research (JMLR)* (2022).

[15] Rahimi A. and Alahi A. A Multi-Loss Strategy for Vehicle Trajectory Prediction: Combining Off-Road, Diversity, and Directional Consistency Losses. *In Computer Vision – ECCV 2020,* (2024).

License

Non-exclusive licence to reproduce thesis

I, Yucui Wu,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to reproduce, for the purpose of preservation, including for the purpose of preservation in the DSpace digital archives until expiry of the term of validity of the copyright, **Latent-Gated-MoE: A Novel Mixture of Experts with Latent Space Splitting for Multi-class Image Classification**, supervised by Dr. Kallol Roy and Dr. Jan Pisek.
2. Making the thesis available to the public is not allowed.
3. I am aware of the fact that the author retains the right refered to in point 1.
4. This is to certify that granting the non-exclusive licence does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Yucui Wu

12/05/2025