BCSE498J Project-II / CBS1904/CSE1904 - Capstone Project

Cross-Modal Knowledge Distillation for Ultra-Lightweight Edge AI: A Resource-Aware Approach

21BCT0066 - Aaron Mano Cherian

Under the Supervision of

Dr. Manoov R

Assistant Professor Sr Grade 1

School of Computer Science and Engineering (SCOPE)

B.Tech.

in

Computer Science and Engineering

(with specialization in Internet of Things)

School of Computer Science and Engineering

February 2025

ABSTRACT

While edge AI continues to offer promising solutions for real-time applications, the deployment of multi-modal deep learning models remains particularly challenging considering their substantial computational and memory requirements on resource-constrained devices. This paper approaches this problem through the integration of a Resource-Aware Cross-Modal Knowledge Distillation (CMKD). Our approach introduces three key checkmarks: (1) a dynamic resource allocation mechanism that adaptively manages computational resources across different modalities based on real-time device constraints, (2) modality-specific compression techniques that optimize knowledge transfer while minimizing memory footprint, and (3) a lightweight feature alignment strategy that maintains cross-modal performance under varying resource conditions.

TABLE OF CONTENTS

Sl.No	Contents	Page No.
	Abstract	**2**
1.	**INTRODUCTION**	**4-5**
	1.1 Background	**4**
	1.2 Motivations	**4**
	1.3 Scope of the Project	**4-5**
2.	**PROJECT DESCRIPTION AND GOALS**	**5-9**
	2.1 Literature Review	**5-6**
	2.2 Gaps Identified	**6**
	2.3 Objectives	**7**
	2.4 Problem Statement	**7-8**
	2.5 Project Plan	**8-9**
3.	**REQUIREMENT ANALYSIS ***	**9-10**
	3.1 Hardware and Software Specifications	**9-10**
4.	**SYSTEM DESIGN***	**11-12**
	4.1 Software Requirements Specification (SRS)	**11-12**
5.	**REFERENCES**	

INTRODUCTION

1.1 Background

The increasing demand for real-time artificial intelligence applications on edge devices has created a significant challenge in deploying complex deep learning models. Edge devices, which include anything from smartphones and IoT sensors, to embedded systems, typically have limited computational resources, memory, and power capacity. This limitation poses a substantial barrier to implementing sophisticated multi-modal deep learning models that could otherwise provide rich, context-aware insights from various input sources such as visual, audio, and sensor data.

Traditional approaches to edge AI often involve compromising between model performance and resource utilization, leading to either degraded accuracy or impractical resource requirements. While model compression techniques like pruning and quantization have shown promise, they often result in significant performance degradation when applied to multi-modal systems. This creates a critical need for more sophisticated approaches that can maintain high performance while dramatically reducing resource requirements.

1.2 Motivations

The exponential growth in edge computing applications over the past few years has created an urgent need for efficient AI models that can operate within strict resource constraints. Current solutions often fail to balance performance and resource utilization effectively, particularly in multi-modal scenarios where different types of input data must be processed simultaneously.

Privacy and latency concerns are driving the need for more processing to occur directly on edge devices rather than in the cloud. This shift requires novel approaches to model optimization that can maintain performance while significantly reducing computational requirements.

The emergence of Cross-Modal Knowledge Distillation (CMKD) as a promising technique for transferring knowledge between different modalities offers new opportunities for creating ultra-lightweight edge AI systems. By leveraging CMKD, we can potentially create more efficient models that maintain high performance while requiring fewer resources.

1.3 Scope of the Project

This project focuses on developing and implementing a resource-aware CMKD framework specifically designed for edge AI applications. Thus scope of this project encompasses the development of a dynamic resource allocation mechanism for managing computational resources across different modalities. It also considers the implementation of modality-

specific compression techniques optimized for knowledge transfer. The possibility of the creation of a lightweight feature alignment strategy for maintaining cross-modal performance along with the evaluation framework using real-world datasets and edge computing scenarios will be carefully considered and pursued. Finally, a simulated implementation of the system across cloud, fog, and edge computing layers would be executed to quantify the progress of this paradigm.

PROJECT DESCRIPTION AND GOALS

2.1 Literature Review

The field of Cross-Modal Knowledge Distillation has evolved significantly in recent years, with various approaches emerging to address the challenges of transferring knowledge between different modalities. Current research has demonstrated the effectiveness of CMKD in several domains, including action recognition, emotion detection, and image processing.

In the context of action recognition, researchers have successfully shown that networks trained on RGB videos can be adapted to recognize actions from 3D human pose sequences. This transfer of knowledge between modalities has proven particularly effective in scenarios where data in the target modality is limited or difficult to obtain.

Emotion recognition research has demonstrated the potential of CMKD in transferring knowledge from complex, multi-modal systems to simpler, single-modal implementations. For instance, the EmotionKD framework has shown promising results in transferring knowledge from EEG-based systems to simpler GSR-based implementations, maintaining high performance while significantly reducing complexity.

The application of CMKD in image processing has revealed interesting approaches to supervision transfer, particularly in scenarios where direct supervision in certain modalities is expensive or impractical to obtain. These approaches have demonstrated the potential for creating more efficient, lightweight models while maintaining acceptable performance levels.

Topic	Description
Cross-Modal Knowledge Distillation for Action Recognition	CMKD allows networks trained on one modality, such as RGB videos, to be adapted to recognize actions in another modality, such as sequences of 3D human poses. The process steps include: 1) Extract the knowledge of the trained teacher network (Source Modality), 2) Transfer it to a small ensemble of student networks (Target Modality). Commonly used loss: KL-loss. Preferred loss: Cross-entropy loss + Mutual learning. **Link**: https://sci-hub.ru/https://ieeexplore.ieee.org/abstract/document/8802909
Knowledge as Priors: Cross-	In cases of inadequate data, one can generalize distilled cross-modal knowledge learned from a source dataset, containing paired examples

Modal Knowledge Generalization for Datasets Without Superior Knowledge	from both modalities, to a target dataset by modeling knowledge as priors on parameters of the student network. **Link**: https://openaccess.thecvf.com/content_CVPR_2020/papers/Zhao_Knowledge_As_Priors_Cross-Modal_Knowledge_Generalization_for_Datasets_Without_Superior_CVPR_2020_paper.pdf
EmotionKD: A Cross-Modal Knowledge Distillation Framework for Emotion Recognition Based on Physiological Signals	This approach targets emotion detection using EEG and GSR data signals. GSR is easy to access, but EEG requires more time. Through CMKD, multi-modal features are fused and transferred to an unimodal GSR model, improving performance. **Link**: https://github.com/YuchengLiu-Alex/EmotionKD
Cross Modal Distillation for Supervision Transfer	This study focuses on CMKD applied to image data, investigating various supervision transfer methods. It may provide a framework for use in our own paper. **Link**: https://openaccess.thecvf.com/content_cvpr_2016/html/Gupta_Cross_Modal_Distillation_CVPR_2016_paper.html

2.2 Gaps Identified

Current CMKD approaches primarily focus on performance optimization without explicitly considering resource constraints. This creates a significant gap in the context of edge AI applications, where resource awareness is crucial. Existing solutions often fail to address the dynamic nature of edge computing environments, where available resources may fluctuate based on device conditions and concurrent applications.

Another significant gap exists in the integration of CMKD with resource-aware computing paradigms. While both fields have been extensively studied independently, their intersection remains largely unexplored. This creates an opportunity for developing novel approaches that combine the benefits of both domains.

The lack of standardized approaches for measuring and optimizing resource utilization in CMKD systems presents another gap. Current research typically focuses on accuracy metrics without considering the practical implications of deployment on resource-constrained devices.

2.3 Objectives

The primary aim of this research is to develop a resource-aware CMKD framework that effectively bridges the gap between theoretical CMKD approaches and practical edge AI deployment. This overarching goal can be broken down into several specific objectives:

To design and implement a dynamic resource allocation mechanism that can adaptively manage computational resources across different modalities based on real-time device constraints. This system will monitor available resources and adjust the distribution of computational load accordingly, ensuring optimal performance under varying conditions.

To develop modality-specific compression techniques that optimize knowledge transfer while minimizing memory footprint. This objective includes creating specialized compression algorithms that consider the unique characteristics of each modality while maintaining the essential information required for effective knowledge transfer.

To create a lightweight feature alignment strategy that can maintain cross-modal performance under varying resource conditions. This involves developing efficient methods for aligning features across modalities while adapting to the available computational resources.

Finally, to validate the effectiveness of the proposed framework through comprehensive experimentation and evaluation on real-world edge devices, ensuring practical applicability and performance benefits.

2.4 Problem Statement

The deployment of multi-modal deep learning models on edge devices presents several critical challenges that this research aims to address:

Edge devices operate under severe resource limitations, including restricted computational power, limited memory capacity, and constrained energy resources. Current multi-modal deep learning models, which typically require substantial computational resources, are often impractical for deployment on these devices. This creates a fundamental conflict between the desire for sophisticated AI capabilities and the practical limitations of edge hardware.

Existing approaches to model optimization for edge deployment often result in significant performance degradation. While techniques such as model compression and pruning can reduce resource requirements, they frequently lead to unacceptable decreases in model accuracy, particularly in multi-modal scenarios where complex feature interactions are crucial for performance.

Edge devices operate in dynamic environments where available resources fluctuate based on various factors such as concurrent applications, battery levels, and thermal conditions. Current solutions lack the ability to dynamically adapt to these changing conditions, leading to either resource overflow or underutilization.

The transfer of knowledge between different modalities (such as visual, audio, and sensor data) while maintaining model performance is particularly challenging in resource-constrained environments. Existing CMKD approaches are not optimized for edge deployment and often require substantial computational resources.

The deployment and maintenance of multi-modal AI systems across numerous edge devices present significant challenges in terms of model updates, performance monitoring, and resource optimization. Current solutions lack efficient mechanisms for managing these aspects at scale.

This project addresses these challenges through the development of a resource-aware CMKD framework specifically designed for edge AI applications. The proposed solution aims to maintain high model performance while significantly reducing resource requirements and providing dynamic adaptation capabilities.

2.5 Project Plan

I have structured the approach to this concept in the following phases:

Phase 1: Foundation and Research establishes the groundwork for the entire project. This crucial initial phase begins with an extensive literature review of CMKD techniques, coupled with a thorough analysis of existing edge AI deployment strategies. Through this comprehensive research, we will identify key performance metrics and constraints that will guide our development process. All research findings and gap analysis will be meticulously documented to inform subsequent phases. During this phase, we will also undertake the critical task of architecture design, where the theoretical framework of our system will be developed and finalized. The final component of this phase involves setting up the development environment, including the installation and configuration of all necessary software tools. This setup process will be thoroughly documented to ensure reproducibility and maintain consistency throughout the project lifecycle.

Phase 2: Core Development represents the heart of our implementation work. This phase begins with the development of the teacher model, which involves implementing the base architecture and integrating multi-modal processing capabilities. This sophisticated model will serve as the foundation for knowledge distillation. Following the teacher model, we will focus on implementing the student model, with particular emphasis on resource efficiency and compression techniques. These models will be integrated into our CMKD framework, which involves creating robust knowledge distillation mechanisms and implementing resource-aware optimization techniques. Throughout this phase, we will develop and integrate performance monitoring systems to ensure our implementation meets our specified requirements. Comprehensive documentation will be maintained for all components of the framework.

Phase 3: Integration and Testing focuses on bringing all components together into a cohesive system. The integration process begins with combining all system components and implementing necessary communication protocols between different parts of the system. We will develop and document detailed deployment procedures to ensure smooth

implementation. This phase also encompasses extensive testing and validation, including the development of a comprehensive test suite that will evaluate the system under various operating conditions. Resource utilization analysis will be conducted to ensure our system meets efficiency requirements. The optimization stage of this phase involves fine-tuning system performance, improving resource utilization, and enhancing scalability. All optimization procedures will be thoroughly documented to enable future improvements and maintenance.

Phase 4: Deployment and Evaluation represents the final stage of our project implementation. During this phase, we will develop comprehensive deployment procedures and implement robust monitoring systems to track system performance. Maintenance protocols will be established to ensure long-term system stability and reliability. A thorough performance evaluation will be conducted, including comprehensive system evaluation and detailed analysis of performance metrics and resource utilization. The final stage of this phase involves completing all technical documentation, preparing user manuals, and developing maintenance guides. The project culminates in the preparation of a final project report that documents all aspects of the implementation, from initial research through to deployment and evaluation.

Throughout all phases, we maintain a strong focus on documentation and quality assurance. Each component and process will be thoroughly documented to ensure knowledge transfer and enable future maintenance and improvements. Regular reviews and assessments will be conducted to ensure the project remains aligned with its objectives and maintains high quality standards. This structured approach ensures a systematic and thorough implementation of our resource-aware CMKD framework, while maintaining flexibility to address challenges and opportunities as they arise during the development process.

REQUIREMENT ANALYSIS

3.1 Hardware and Software Specifications

Hardware Requirements:

Cloud Layer:

CPU: 8+ cores (Ryzen 7)

RAM: 32GB

Storage: 1TB SSD

GPU: NVIDIA Tesla V100

Network: High-speed internet connection

Fog Layer:

CPU: 4+ cores (Ryzen 7)

RAM: 16GB

Storage: 512GB SSD

GPU: NVIDIA RTX 2080

Network: Reliable internet connection (100Mbps+)

Edge Layer: (need-dependent)

CPU: Dual-core processor

RAM: 4GB

Storage: 64GB

GPU: Integrated graphics or mobile GPU

Network: Standard wireless connectivity

Software Requirements

Operating System: Windows 11

Container Platform: Docker 20.10+

Orchestration: Kubernetes 1.21+

Deep Learning Framework: PyTorch 1.9+

Distributed Computing: Ray 1.4+

Database: MongoDB 4.4+

Storage: MinIO

Development Tools: Python 3.8+, Git, Docker

4. SYSTEM DESIGN*

4.1. Software Requirements Specification (SRS)

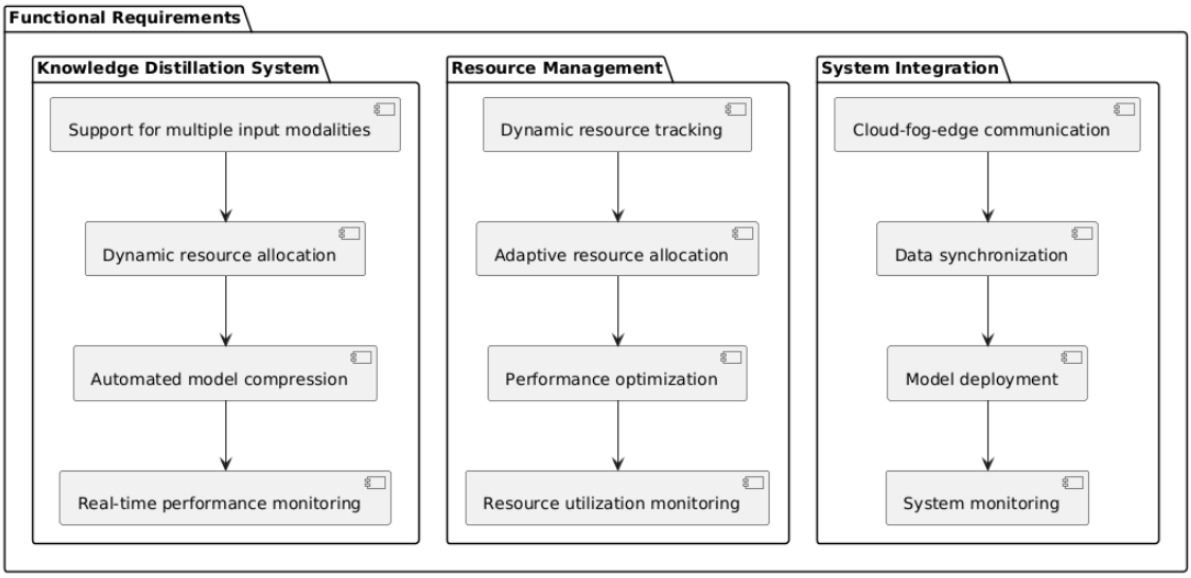

Fig 1: Functional Requirements

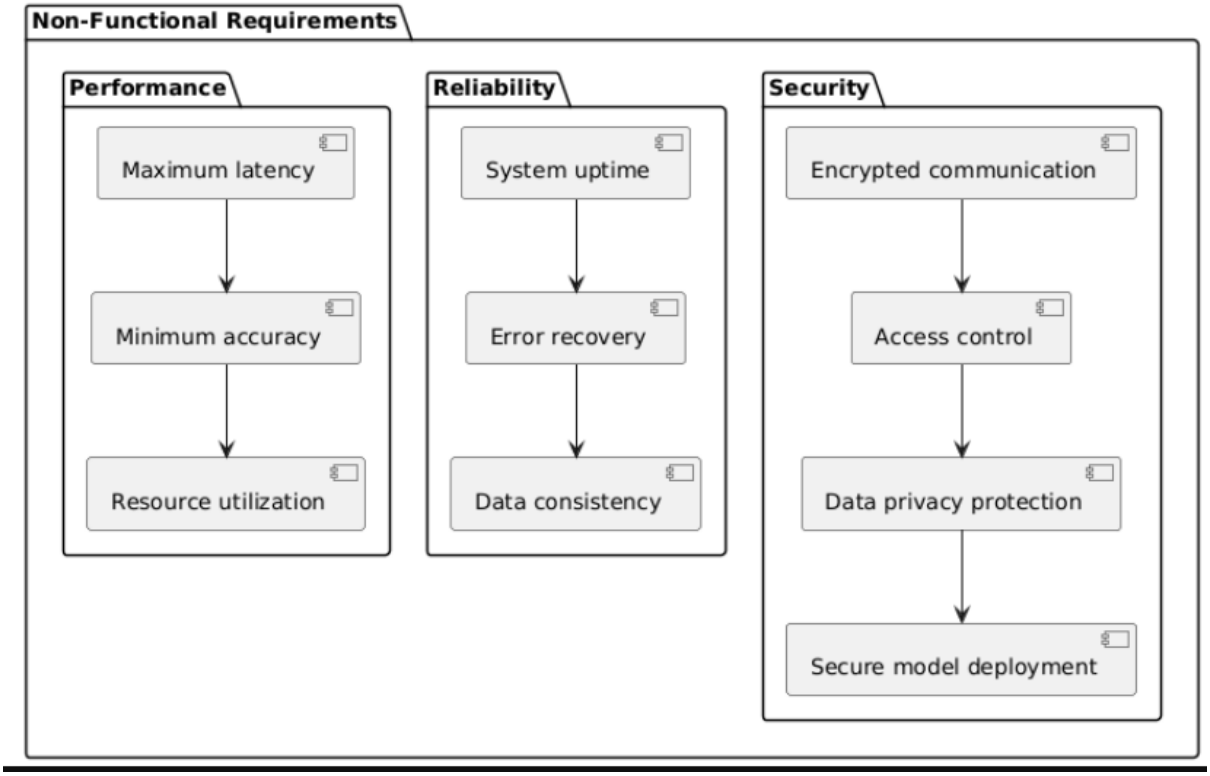

Fig 2: Non-Functional Requirements

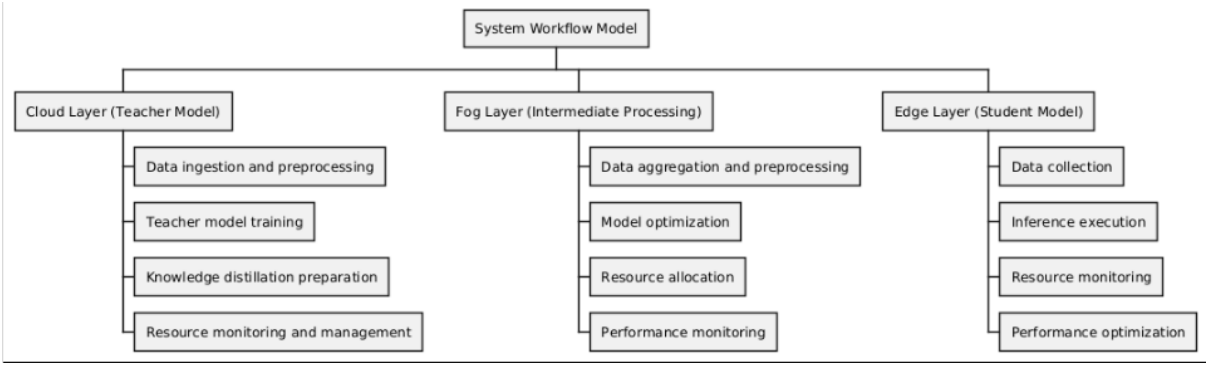

Fig 2: Work Breakdown Structure