School of Computer Science
Faculty of Science and Engineering
University of Nottingham
Malaysia

UG FINAL YEAR DISSERTATION REPORT

ALPETNet: A Hybrid Approach to Lip Reading for Enhanced Security and Surveillance

Student's Name	: Royceton Yeoh Tze Jian
Student Number	: 20509678
Supervisor Name	: Dr. Tissa Chandesa
Year	: 2025

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF BACHELOR OF SCIENCE IN COMPUTER SCIENCE WITH ARTIFICIAL INTELLIGENCE (HONS)
THE UNIVERSITY OF NOTTINGHAM

University of Nottingham
UK | CHINA | MALAYSIA

ALPETNet: A Hybrid Approach to Lip Reading for Enhanced Security and Surveillance

Submitted in May 2025, in partial fulfillment of the conditions of the award of the degrees B.Sc.

Royceton Yeoh Tze Jian
School of Computer Science
Faculty of Science and Engineering
University of Nottingham
Malaysia

I hereby declare that this dissertation is all my own work, except as indicated in the text:

Signature _____

Date 2 / 5 / 2025

Acknowledgements

I would like to express my heartfelt appreciation to my Final Year Project supervisor, Dr Tissa Chandesa, for his exceptional guidance, unwavering support, and constant encouragement throughout both semesters of my dissertation journey. His decision to assign me this particular topic has significantly enhanced my understanding of the computer vision field. I deeply value his dedicated efforts in reviewing my dissertation proposal, ethics form, interim report, and final report. His thoughtful feedback and recommendations have substantially improved the quality of my dissertation, and I am thankful for his steadfast dedication to helping me succeed.

My sincere thanks also goes to all faculty members in the School of Computer Science who conducted Final Year Project lectures and provided clear guidance on dissertation execution, documentation, and presentation. Their commitment to education and student support has been fundamental to my academic achievements.

Additionally, I wish to acknowledge the invaluable support and assistance rendered to me by my fellow classmates and friends while embarking on this dissertation. Their helpfulness and patience in troubleshooting challenges have continuously motivated me.

Lastly, I would like to express my heartfelt gratitude to my family for their immense encouragement and support throughout my final year. Their constant presence, understanding, and love have been a great source of strength to me, especially during the most challenging moments of this journey. Last but not least, I am sincerely grateful to everyone who has contributed to my academic journey, and I will always cherish the support I have received.

Abstract

This dissertation explores a hybrid approach to Visual Speech Recognition (VSR), commonly known as lip reading, which is crucial for enhancing communication in environments with compromised audio, such as surveillance, noisy settings, or for individuals with hearing impairments. The proposed system integrates convolutional and transformer-based modules to capture detailed spatial and temporal features from silent video frames of lip movements.

Two novel architectures are developed: LPETNet (Light Patch Embedding Transformer Network) and ALPETNet (Attention-reinforced Light Patch Embedding Transformer Network). LPETNet focuses on efficiency through patch embeddings and depthwise separable 3D convolutions, while ALPETNet enhances performance further by incorporating gated transformers and channel attention mechanisms. Both models fuse spatial and temporal features and refine sequence modeling using BiGRUs.

The final system successfully balances robust performance and computational efficiency and yields promising results for real-time, interpretable lip-reading applications. This dissertation advances the field of hybrid feature fusion in deep learning and highlights the potential of such architectures for practical deployment in VSR systems.

Table of Contents

List of Figures

List of Tables

1 Introduction

Lip reading, also referred to as visual speech recognition (VSR), is an essential area of research within security, surveillance, and communication technologies. It enables the decoding of spoken messages solely through visual cues in scenarios where audio signals are unavailable or unreliable. This capability has significant applications in environments with high noise interference, communication with individuals who are deaf or hearing-impaired, and forensic investigations. Early studies, such as Kumar et al. (2018), demonstrated the feasibility of reconstructing intelligible speech from silent video feeds, a breakthrough that proved valuable for surveillance and forensic analysis where audio streams are compromised. Building upon this foundation, Bulzomi et al. (2023) proposed a neuromorphic lip-reading framework capable of operating under constrained computational environments, reinforcing VSR's suitability for edge-based surveillance systems. Similarly, Hong et al. (2023) introduced a robust multimodal speech recognition system designed for security-critical applications, emphasising resilience in scenarios where audio modalities are unreliable or intentionally suppressed. Further advancing the field, Wang et al. (2024a) developed a multi-layer cross-attention fusion approach for audiovisual speech recognition (AVSR), enhancing performance in noisy environments, which is particularly pertinent to real-world surveillance and communication systems. These advancements affirm that VSR continues to play an increasingly critical role in modern security infrastructures, extending its relevance beyond controlled settings to real-world surveillance, emergency response, and privacy-preserving communication technologies.

Speech perception is inherently multimodal, involving both auditory and visual information. The McGurk effect, introduced by McGurk and MacDonald (1976), highlights how mismatched auditory and visual stimuli can lead to the perception of entirely different phonemes. This phenomenon underscores the importance of visual information in understanding speech, particularly in noisy environments where auditory signals are difficult to discern. Additionally, individuals with hearing impairments rely heavily on visual cues in the form of lip movements to interpret spoken words. However, Fisher (1968) revealed that human accuracy in lip reading remains limited, with hearing-impaired individuals achieving only 17% accuracy for monosyllabic words and 21% for compound words. These findings highlight the need for automated systems to bridge this gap.

To address challenges in VSR, this dissertation will introduce the Attention-reinforced Light Patch Embedding Transformer Network (ALPETNet), a hybrid lip-reading system designed for improved accuracy and efficiency. The architecture will integrate a convolutional pathway, employing depthwise separable 3D Convolutional Neural Networks (CNNs) and channel attention to extract spatial features efficiently, with a transformer pathway that will use multi-scale patch embedding and gated transformer blocks to model temporal dependencies. Features from both pathways will be fused through cross-attention fusion, enabling complementary information integration. The fused features will then be processed by a Bidirectional Gated Recurrent Unit (BiGRU) for sequence modelling and classified through a fully connected layer. This design is intended to ensure robust performance under challenging conditions, including variable lighting.

1.1 Problems and Motivation

Lip reading poses significant challenges due to the complexity of visual cues and inherent limitations in human performance when recognising speech using only visual information. Fisher (1968) highlight that even experienced professionals achieve limited accuracy rates when interpreting lip movements. Machine lipreading systems offer a promising solution by automating this process and achieving higher accuracy rates through deep learning-based approaches. However, real-world applications continue to face persistent obstacles in the form of variable lighting conditions and computational inefficiencies.

Homophonic ambiguity is particularly challenging because many words produce nearly identical lip movements despite differing phonemes. For example, Wang et al. (2022) emphasise that words like `pack`, `back`, and `mac` are visually indistinguishable without contextual information. Additionally, short-duration words often provide insufficient visual data for accurate recognition. These limitations underscore the need for advanced architectures capable of capturing fine-grained spatiotemporal features from video sequences.

Recent advancements in hybrid architectures have shown promise in addressing these challenges. Jeon et al. (2021) proposed a system combining 3D CNNs with densely connected layers to improve feature extraction while reducing computational overhead. Similarly, Wang et al. (2022) introduced 3D CvT, which integrated convolutional layers for local feature extraction with transformers for global context modelling. More recently, Park et al. (2024) presented SwinLip, an efficient visual speech encoder utilising the Swin Transformer to enhance both performance and computational efficiency in lip-reading tasks. Furthermore, Xue et al. (2023) proposed a fine-grained sequence-to-sequence lip reading framework that incorporated self-attention mechanisms and self-distillation to improve recognition accuracy across complex sequences.

The motivation for this dissertation stems from these evolving advancements and limitations in traditional VSR systems. By integrating hybrid transformer models with spatiotemporal feature extraction mechanisms, this dissertation will seek to address existing gaps in VSR technology while enhancing its applicability across security and communication domains.

1.2 Aim

The dissertation aims to develop a robust lip-reading system through a hybrid architecture that integrates CNNs, Recurrent Neural Networks (RNNs), and Transformer-based mechanisms. The system will be designed to address key challenges in VSR, including variable lighting conditions and computational inefficiencies, while enabling accurate translation of lip movements into spoken language.

1.3 Objectives

The dissertation is guided by the following objectives:

1. Design a system to effectively capture both detailed and broader patterns in lip movements, ensuring accurate recognition of speech.
2. Develop a method to merge information from multiple components, to improve the understanding of lip movements over time.
3. Address real-world challenges, including variable lighting conditions, to ensure reliable system performance in diverse environments.

1.4 Dissertation Overview

This dissertation is structured as follows. Chapter 2 will review relevant works in VSR. Chapter 3 will outline the proposed Light Patch Embedding Transformer Network (LPETNet) and ALPETNet architectures. Chapter 4 will present the experimental results and provide analysis. Chapter 5 will consolidate the key contributions of the dissertation and propose directions for future work. It will also reflect on the technical, ethical, and personal aspects encountered during the dissertation.

2 Related Work

This chapter reviews prior work in the field of VSR, tracing its progression from traditional machine learning methods to advanced deep learning frameworks. Section 2.1 discusses foundational techniques such as Hidden Markov Models and handcrafted feature extraction. Section 2.2 discusses the use of CNNs for spatial modelling. Section 2.3 discusses the application of RNNs for capturing temporal dependencies. Section 2.4 discusses recent developments in Transformer-based architectures for spatiotemporal representation. Section 2.5 discusses key real-world challenges in VSR deployment. Section 2.6 discusses benchmark datasets that have enabled empirical progress in the field. Section 2.7 discusses the broader impact of these technologies and their applications across domains.

2.1 Traditional Methods and Early Approaches

Early studies in automatic lip reading primarily relied on traditional machine learning techniques for feature extraction and classification. Makhlouf et al. (2013) demonstrated the use of Hidden Markov Models (HMMs) for temporal modelling and Discrete Cosine Transform (DCT) for visual feature extraction. These methods provided foundational insights but were limited by their inability to generalise across varying conditions such as occlusions and noise.

An important advancement in VSR was the introduction of Active Shape Models (ASMs) by Luettin et al. (1996). ASMs use a statistical approach to model the shape of an object, such as the lips, by learning patterns of variability from a training set. This author developed ASMs to robustly detect, track, and parameterise lip movements while avoiding user-imposed constraints or thresholds. This method proved effective in handling variations in lighting, rotation, scale, and translation. By modelling both the outer and inner contours of the lips, ASMs enhanced their ability to capture viseme-related variances while ignoring irrelevant linguistic or image variabilities.

Building upon ASMs, Active Appearance Models (AAMs) extended the concept by incorporating both shape and texture information into a unified framework. Katsamanis et al. (2009) proposed AAMs for facial analysis and speech inversion tasks. This author demonstrated that AAMs effectively captured non-rigid shape deformations and texture variations simultaneously. Their approach facilitated robust lip tracking and articulatory trajectory estimation under diverse conditions. Unlike ASMs, which focused solely on shape constraints, AAMs integrated texture information across the target object, providing a richer representation of visual speech features.

Notably, Goldschen et al. (1997) performed visual-only sentence-level lip reading using HMMs on a limited dataset with hand-segmented phones. This was followed by Neti et al. (2000), who introduced sentence-level AVSR by combining HMMs with hand-engineered features on the IBM ViaVoice dataset. Their approach improved speech recognition performance in noisy environments by fusing visual features with audio ones. However, their visual-only results were limited as they relied on rescoring noisy

audio lattices rather than pure visual recognition.

The integration of visual and audio modalities has proven effective in enhancing speech recognition systems, particularly in noisy environments. Potamianos et al. (2003) demonstrated this by achieving significant improvements on the connected DIGIT corpus. Their audiovisual approach achieved word error rates (WERs) of 30.29% for audio-only recognition, 23.6% for visual-only recognition, and 10.35% for AVSR using global fusion weights. These results highlight the potential of combining modalities to improve robustness in automatic speech recognition, particularly under challenging acoustic conditions.

In addition to HMM-based methods, other traditional approaches explored geometric and appearance-based modelling for lip reading. Matthews et al. (2002) utilised Principal Component Analysis to model lip shapes through eigenvectors, often referred to as `eigenlips`, to parameterise lip image sequences for speech recognition. Their approach demonstrated the effectiveness of PCA in reducing dimensionality while preserving key visual features, enabling robust performance in isolated letter recognition tasks.

While these methods laid the groundwork for automated lip reading, they faced significant limitations in generalisation across speakers and handling motion features dynamically. Gergen et al. (2016) addressed these issues by employing speaker-dependent training on a linear discriminant analysis transformed version of DCT features in an HMM/GMM system, achieving an accuracy of 86.4% on the GRID corpus for speaker-dependent tasks.

Early approaches to lip-reading often relied on extensive preprocessing techniques, such as handcrafted pipelines for extracting image features or motion-based features like optical flow. Zhou et al. (2014) summarised these efforts, noting that traditional methods faced significant challenges, including speaker dependency, pose variation, and the effective encoding of temporal information. While statistical models like HMMs were able to capture temporal dynamics to some extent, they struggled to fully model the spatiotemporal complexities inherent in video sequences. These limitations underscored the need for more robust and integrated approaches in VSR. These foundational works paved the way for modern deep learning approaches, which have largely replaced handcrafted pipelines with end-to-end architectures capable of learning robust features directly from raw data.

2.2 Convolutional Neural Networks

CNNs have played a pivotal role in advancing lip-reading systems by automating the extraction of spatial features from video frames. Assael et al. (2016) introduced LipNet, an end-to-end architecture that utilised spatiotemporal convolutions for feature extraction, achieving remarkable improvements in sentence-level lip reading. Similarly, Gutierrez and Robert (2017) demonstrated the effectiveness of VGG-based CNN architectures for word-level classification tasks on the MIRACL-VC1 dataset, setting a benchmark for modern lip-reading systems.

Recent studies have explored optimising CNN architectures for lip-reading tasks, focusing on computational efficiency and performance. Fu et al. (2023) proposed a lightweight model combining ShuffleNet with the Convolutional Block Attention Module, achieving competitive accuracy while reducing computational costs. Similarly, Arakane and Saitoh (2023) investigated EfficientNet-based models tailored for word-level lip-reading, demonstrating their effectiveness in balancing accuracy and efficiency. These efforts showcase the potential of lightweight CNN architectures, such as ShuffleNet and EfficientNet, for real-time applications in lip-reading systems.

In addition to these advancements, NadeemHashmi et al. (2018) proposed a pure CNN-based lip-reading model featuring a twelve-layer architecture with batch normalisation. This approach aimed to reduce variances caused by external factors such as lighting conditions and speaker accents. Despite its simplicity and reduced computational requirements, the model achieved a validation accuracy of only 52.9% on the MIRACL-VC1 dataset. While this performance lags behind more complex hybrid models, it underscores the challenges of relying solely on spatial feature extraction without incorporating temporal modelling.

Despite their success, CNNs are inherently limited in modelling temporal dependencies due to their focus on spatial information. This limitation has necessitated the integration of temporal modelling techniques, such as RNNs and Transformers, to capture sequential patterns in lip movements.

2.3 Recurrent Neural Networks

RNNs have been pivotal in modelling temporal dependencies in lip-reading tasks due to their ability to process sequential data effectively. Amongst RNN variants, Long Short-Term Memory (LSTM) networks have been widely adopted for their capability to address the vanishing gradient problem, which is common in standard RNNs. Fung and Mak (2018) demonstrated the effectiveness of LSTMs combined with Maxout CNNs for end-to-end lip reading in low-resource settings, achieving robust performance on small datasets. Similarly, Garg et al. (2016) utilised a combination of CNNs and LSTMs for word-level classification tasks, highlighting the strength of LSTMs in capturing temporal patterns in visual speech data.

Gated Recurrent Units (GRUs) were introduced by Cho et al. (2014) as a simplified yet effective alternative to LSTM networks. The author developed GRUs to address challenges such as the vanishing gradient problem in traditional RNNs, enabling efficient training while maintaining robust performance in sequence modelling tasks. GRUs incorporate gating mechanisms that adaptively control the flow of information, making them particularly suitable for capturing temporal dependencies across varying time scales.

Subsequently, Chung et al. (2014) conducted an empirical evaluation of GRUs, comparing them against LSTMs and traditional tanh units on tasks such as polyphonic music modelling and speech signal modelling. The dissertation demonstrated the effectiveness of GRUs in sequence modelling, highlighting their comparable performance

to LSTMs in terms of generalisation and computational efficiency.

Building on these foundational works, BiGRUs have been extensively employed in lip-reading models due to their ability to capture both forward and backward temporal contexts. This bidirectional capability allows BiGRUs to process sequential data, such as video frames, more effectively by utilising information from both past and future time steps. As a result, BiGRUs have become a popular choice for modelling temporal dynamics in lip-reading systems. For example, Sarhan et al. (2021) incorporated BiGRUs into their HLR-Net model, combining them with inception layers to enhance temporal modelling capabilities. This integration allowed the model to achieve significant improvements in recognition accuracy on the GRID corpus.

In LipNet, Assael et al. (2016) employed BiGRUs alongside CNNs to aggregate visual features over time. This approach enabled the model to achieve state-of-the-art performance on the GRID corpus by leveraging the RNN's ability to process sequential data effectively. Additionally, Devi et al. (2023) explored the use of BiGRUs combined with 3D CNNs for silent speech recognition, demonstrating their effectiveness in capturing spatiotemporal dependencies for lip-reading tasks.

While BiGRUs have proven sufficient for many lip-reading applications, they often face challenges related to scalability and computational efficiency when applied to large-scale datasets. Transformer-based architectures have recently emerged as an alternative, offering improved scalability and parallelisation while maintaining strong performance in sequence modelling tasks.

2.4 Transformer-Based Architectures

The introduction of Transformer-based models has addressed many limitations of traditional RNNs in lip reading. Transformers excel at capturing long-range dependencies through self-attention mechanisms, making them well-suited for video-based tasks. The Transformer architecture, introduced by Vaswani et al. (2017), replaced recurrent connections with multi-head self-attention mechanisms, significantly improving scalability and parallelisation. This innovation laid the foundation for applying Transformers to computer vision tasks like lip reading.

In the field of lip reading, Vision Transformers have been adapted to process spatiotemporal data effectively. For instance, Liu et al. (2021) proposed the Swin Transformer, which uses shifted window attention mechanisms to reduce computational complexity while retaining global context information. Building on this foundation, Park et al. (2024) developed SwinLip by introducing a 3D Temporal Embedding Module and a 1D Convolutional Attention Module to enhance temporal modelling capabilities. These innovations allowed SwinLip to achieve state-of-the-art performance on datasets such as Lip Reading in the Wild (LRW) while reducing computational load. SwinLip demonstrated robust performance across multiple languages and benchmarks, showcasing its ability to efficiently capture both local and global dependencies in lip movements.

Another notable Transformer variant is the Convolutional Vision Transformer (CvT), introduced by Wu et al. (2021). CvT integrates convolutional layers into the token embedding process to combine hierarchical feature extraction with self-attention mechanisms. This approach enhances spatial feature learning while maintaining the scalability of Transformers. Wang et al. (2022) applied CvT to lip-reading tasks and demonstrated its potential to capture global information effectively. However, its performance was found to be comparable to CNN-based methods due to challenges in reducing computational complexity.

Resformer represents another advancement in Transformer-based architectures for lip reading. Xue et al. (2023) proposed Resformer as a fine-grained sequence-to-sequence model that combines CNN frontends with self-attention modules for pixel-wise learning and temporal encoding. By leveraging self-distillation techniques, Resformer further improves model performance by refining predictions using auxiliary loss functions. Experiments on datasets such as GRID and LRW-1000 demonstrated Resformer's ability to reduce WER significantly compared to existing methods.

Despite their advantages, Transformer-based models often require large-scale datasets and significant computational resources for training. This has led researchers to explore hybrid architectures that combine the strengths of CNNs, RNNs, and Transformers. These approaches integrate spatial feature extraction with temporal modelling and global attention mechanisms to address critical challenges in lip reading, such as occlusions, lighting variations, and diverse speaking styles.

In summary, Transformer-based architectures like SwinLip, CvT, and Resformer have revolutionised lip reading by enabling efficient processing of spatiotemporal data while maintaining high accuracy across diverse datasets. Their adaptability and scalability make them promising candidates for advancing VSR systems.

2.5 Real-World Challenges

Real-world applications of VSR systems face numerous challenges that hinder their robustness and accuracy. These challenges include homophonic ambiguity, occlusions, variable lighting conditions, short-duration words, head pose variations, and speaker dependency.

One of the primary challenges in VSR systems lies in visual ambiguities caused by homophenes—phonemes that produce identical or nearly indistinguishable lip movements. For instance, Fernandez-Lopez and Sukno (2018) emphasised that phonemes /b/ and /p/ are visually indistinguishable because voicing occurs at the glottis, which is not visible. This makes it difficult to differentiate words like `bat` and `pat` without additional contextual information. Similarly, velar consonants such as /k/ and /g/ can alter their visual appearance based on the preceding or following phoneme. These ambiguities highlight the lack of a one-to-one correspondence between phonemes and visemes, as discussed by Fernandez-Lopez and Sukno (2018), leading to significant challenges in designing robus VSR systems.

Another critical limitation is the insufficient visual data provided by short-duration words such as a,an, eight, and bin, which last less than 0.02 seconds, as discussed by Xiao (2018). These brief utterances make it challenging for models to extract meaningful features within such a limited timeframe. Additionally, occlusions caused by facial obstructions such as masks or hands further complicate recognition tasks. Bear and Harvey (2017) highlighted that occlusions disrupt the visibility of key articulatory features, reducing model performance. Variable lighting conditions also impair VSR systems by degrading the clarity of visual features, making consistent feature extraction difficult across different environments.

Head pose variations and speaker dependency introduce additional complexities. Changes in head orientation can obscure parts of the mouth region, while inter-speaker variability in lip shapes and speaking styles requires models to generalise effectively across diverse populations, as noted by Wang et al. (2024b). Furthermore, poor temporal resolution compared to audio systems limits the ability of VSR models to capture fine-grained spatiotemporal dynamics.

To address these challenges, researchers have proposed advanced hybrid architectures and methods. For example, Sarhan et al. (2021) developed HLR-Net, which integrates inception layers with BiGRU to enhance recognition accuracy under diverse conditions. Hybrid architectures like LipNet by Assael et al. (2016) combine CNNs with RNNs for sentence-level lip reading, while Park et al. (2024) introduced SwinLip, which integrates Swin Transformers with CNN-based temporal embeddings to reduce computational complexity. Some studies advocate for incorporating global facial features beyond just the mouth region, arguing that facial expressions provide contextual cues that aid in decoding speech, as demonstrated by Shirakata and Saitoh (2020).

Despite these advancements, designing VSR systems that are robust to visual ambiguities remains a significant challenge. While some researchers propose phoneme-to-viseme mappings to address these ambiguities, as highlighted by Fernandez-Lopez and Sukno (2018), others argue that leveraging contextual information from neighbouring characters or language models can resolve these issues more effectively, as noted by Wang et al. (2022). Furthermore, advancements in deep learning architectures like SwinLip have demonstrated promising results in reducing computational complexity while maintaining high accuracy across diverse datasets, as shown by Park et al. (2024).

In summary, addressing real-world challenges in VSR requires a multifaceted approach that combines robust feature extraction techniques with advanced temporal modelling and contextual analysis. By leveraging innovations such as hybrid architectures which have been done in various works such as those described in Wang et al. (2022); Park et al. (2024); Sarhan et al. (2021) and incorporating global facial features, future VSR systems can achieve greater accuracy and reliability across diverse applications.

2.6 Datasets for Lip Reading

Datasets play a pivotal role in advancing lip-reading research by providing structured benchmarks for model evaluation. These datasets, broadly categorised into controlled environments and in-the-wild collections , have been empirically procured from the survey paper by Oghbaie et al. (2025), which provides a comprehensive review of automatic deep lip-reading datasets and their associated challenges.

The GRID corpus, introduced by Cooke et al. (2006a), remains a cornerstone for sentence-level lip reading, featuring 34 speakers delivering 1,000 syntactically structured sentences under controlled conditions. Similarly, the OuluVS2 dataset by Anina et al. (2015) provides phrase-level data with multi-view recordings, enabling pose-invariant model training. MIRACL-VC by Rekik et al. (2014), includes synchronised depth maps for occlusion analysis, and CUAVE by Patterson et al. (2002), designed for multi-speaker and motion scenarios, have been instrumental in studying homophenes and speaker-dependent variations.

In-the-wild datasets, such as the LRW corpus by Chung and Zisserman (2017), leverage real-world variability by extracting over 500 target words from BBC broadcasts. LRW's large-scale, diverse speaking styles make it a benchmark for word-level tasks, despite challenges like co-articulation effects and temporal biases. The Lip Reading Sentences (LRS) dataset by Son Chung et al. (2017) extends this to sentence-level recognition, while LRS2-BBC by Afouras et al. (2018a) and LRS3-TED by Afouras et al. (2018b) expand coverage to diverse BBC genres and unscripted TED talks, respectively. MV-LRS by Son and Zisserman (2017) further addresses pose robustness by incorporating multi-view recordings from dramas and factual programs.

Non-English datasets have also emerged to support multilingual research. LRW-1000 by Yang et al. (2019) offers Mandarin word-level data with tonal variations, while Greek-words by Kastaniotis et al. (2019) and Wild LRRo by Jitaru et al. (2020) provide resources for under-resourced languages like Greek and Romanian. Despite these advancements, dataset creation faces challenges such as labor-intensive annotation, short-duration words, and environmental variability. Synthetic data generation techniques, including Generative Adversarial Networks-based frameworks by Kumar et al. (2020) and viseme concatenation by Wang et al. (2024b), are increasingly explored to mitigate data scarcity and enhance model generalisation.

2.7 Impact

The evolution of lip-reading technologies reflects a significant transition from traditional machine learning methods to advanced deep learning architectures. Early approaches, such as those based on HMMs and DCT, laid the foundation for VSR but were constrained by their inability to handle complex real-world conditions, as discussed in the study by Makhlouf et al. (2013). These methods often relied on handcrafted feature extraction pipelines, which struggled with challenges like speaker dependency, pose variation, and environmental variability.

CNNs have been instrumental in advancing lip-reading systems by automating spatial feature extraction processes. The study by Assael et al. (2016) introduced LipNet, combining CNNs with RNNs to achieve state-of-the-art performance in sentence-level lip reading. Similarly, Gutierrez and Robert (2017) demonstrated the effectiveness of VGG-based CNN architectures for VSR tasks using the MIRACL-VC1 dataset.

RNNs have played a pivotal role in modelling temporal dependencies in visual speech data. The study by Fung and Mak (2018) demonstrated how LSTM networks could be combined with Maxout CNNs to achieve robust performance in low-resource settings. GRUs, introduced by Cho et al. (2014), further advanced temporal modelling by capturing both forward and backward dependencies in speech sequences.

Transformer-based architectures have emerged as a transformative force in lip-reading research. CvT, proposed by Wu et al. (2021), integrates convolutional layers with self-attention mechanisms to enhance spatial feature learning. Building on this, Xue et al. (2023) introduced Resformer, leveraging self-distillation techniques for improved performance. Innovations like SwinLip by Park et al. (2024) have enabled significant reductions in WER on benchmark datasets such as GRID and LRW-1000 while improving scalability and robustness.

The impact of these advancements extends beyond academic research into practical applications. Lip-reading systems are increasingly deployed in security and surveillance for VSR in noisy environments, aiding individuals with hearing impairments through augmented lip views, and enabling silent dictation for privacy-sensitive scenarios. Furthermore, these systems are being used to transcribe silent films, synthesise speech for individuals with disabilities, and resolve multi-talker simultaneous speech scenarios.

Despite these achievements, challenges remain in designing robust VSR systems that can generalise across diverse populations and real-world conditions. By addressing existing gaps through innovations in feature extraction, temporal modelling, and contextual analysis using cutting-edge architectures like CNNs, RNNs, and Transformers, VSR systems have the potential to revolutionise communication technologies across various domains.

2.8 Chapter Summary

This chapter has reviewed the evolution of VSR, from early HMM-based and handcrafted approaches to deep learning models including CNNs, RNNs, and Transformer-based architectures. It has highlighted key advancements in spatial and temporal modelling, the development of benchmark datasets, and the challenges posed by real-world conditions such as homophonic ambiguity and lighting variability. The review has also underscored the limitations of existing models in balancing accuracy, efficiency, and robustness.

These insights form the basis for the methodological choices made in this dissertation. The following chapter, Chapter 3 introduces the proposed architectures—LPETNet and ALPETNet—designed to address the limitations identified in prior work by combining lightweight patch-based convolutional modules with gated transformer mechanisms for efficient and accurate lip reading.

3 Methodology

This chapter outlines the methodology adopted for building and evaluating the proposed VSR models. Section 3.1 discusses the choice and characteristics of the GRID dataset, along with the rationale behind its selection. Section 3.2 describes the preprocessing pipeline used to isolate and normalise the mouth region, enabling temporally consistent visual inputs. Section 3.3 introduces the architectural components of the two proposed models—LPETNet and ALPETNet—including depthwise separable convolutions, patch transformers, channel attention, and BiGRU-based temporal encoders. Section 3.4 details the training configuration, including data augmentation, CTC loss formulation, and the use of the Adam optimiser. Finally, Section 3.4 concludes with an analysis of training stability, providing evidence of smooth convergence for both models.

3.1 Dataset Preparation

This dissertation utilises the GRID corpus, a dataset originally introduced by Cooke et al. (2006a), which consists of video recordings from 34 speakers, each delivering 1,000 six-word sentences following a fixed syntactic structure: **command + colour + preposition + letter + digit + adverb**. An example sentence is “set blue with h seven again.” The dataset is recorded under controlled laboratory conditions and includes speakers of varying genders and accents, enabling the development and evaluation of VSR systems across diverse speaker characteristics. Each video is recorded at 25 frames per second and lasts approximately 3 seconds, resulting in a uniform sequence of 75 frames per sample.

Although the dataset nominally includes 34 speakers (labelled t01 to t34), it is important to note that no video recordings are available for speaker t21 due to an oversight during data collection. As documented in the official Zenodo release of the GRID corpus by Cooke et al. (2006b), speaker 21 is absent from the dataset and thus excluded from this dissertation.

The choice of the GRID corpus is motivated by its suitability for sentence-level lipreading tasks. While larger and more diverse datasets such as LRS2 and LRS3 exist, they are not employed in this dissertation due to their substantial computational requirements, which exceed the capabilities of the system, as discussed in Section 3.4. Moreover, the GRID corpus offers complete and syntactically consistent sentences, providing an appropriate structure for evaluating sequence modelling architectures. Further justification and comparisons with alternative datasets can be found in Section 2.6.

In alignment with the evaluation protocols adopted by Assael et al. (2016), Son Chung et al. (2017), and Xu et al. (2018), 255 sentences from each speaker are reserved for evaluation, and the remaining samples are used for training. This evaluation set serves both validation and testing roles, consistent with prior studies.

3.2 Data Preprocessing

The preprocessing methodology adopted in this dissertation closely follows the approach proposed by Sarhan et al. (2021), which includes three key stages: mouth region extraction, frame normalisation, and sentence preparation. Each video is first decomposed into individual frames at a frame rate of 25 frames per second, corresponding to a sequence length of 75 frames per utterance. As illustrated in Figure 1, the `dlib` and OpenCV libraries are employed to detect 68 facial landmarks per frame, of which points 49 to 68 correspond to the mouth region. These landmarks are used to isolate and crop the mouth area from each frame, removing extraneous facial features and background content. The left column of the figure displays original frames, while the right column shows the corresponding cropped mouth regions.

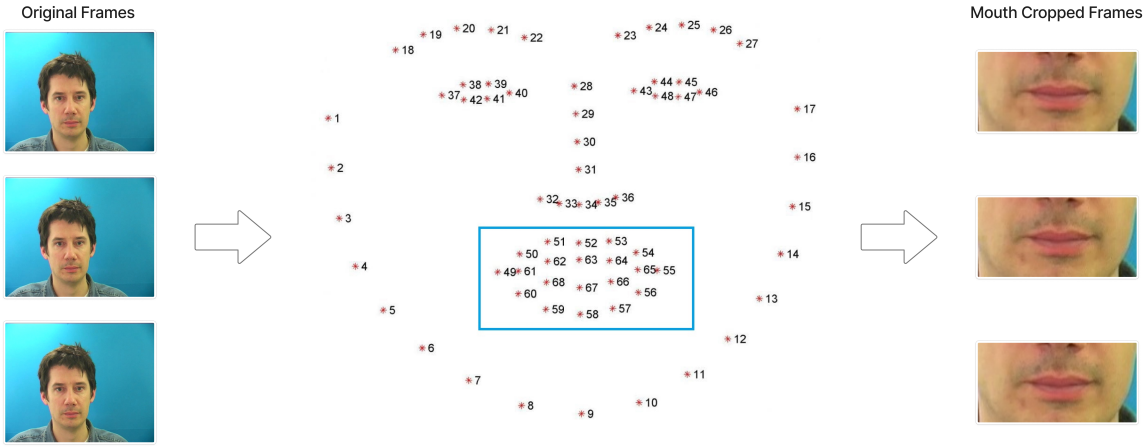

Figure 1: Mouth crop localisation based on facial landmarks.

By isolating the mouth region, this preprocessing step ensures that only the most relevant visual information—namely, the lip contours and articulation patterns—is retained for model input, thereby enhancing spatial focus and mitigating background and lighting variability across recordings. Although the GRID corpus exhibits relatively uniform recording conditions, this focused approach further reduces any residual inconsistencies or artefacts.

In the original pipeline by Sarhan et al. (2021), the cropped mouth regions are resized to 50×100 pixels (height $\times$ width), maintaining a 1:2 aspect ratio. However, in this dissertation, the crops are resized to 80×160 pixels to preserve the same aspect ratio while ensuring compatibility with the input dimensions required by the proposed architecture. Specifically, this modification enables subsequent downscaling of the input to 64×128, a format that would not be feasible if the crops remained at the lower 50×100 resolution. This adjustment ensures a smoother alignment with the model's convolutional and spatiotemporal processing requirements, while preserving sufficient spatial detail for effective lip motion representation.

Following cropping and resizing, each frame undergoes colour normalisation, where pixel values are scaled from the range $[0, 255]$ to $[0, 1]$ to standardise intensity dis-

tributions across the RGB channels. This normalisation step facilitates more stable convergence during training by reducing input variability.

To facilitate batch training and ensure uniform tensor shapes, both the visual input sequences and their corresponding text transcriptions are zero-padded to fixed lengths. Each video sequence is padded to exactly 75 frames, while each character-level transcription is padded to a maximum of 200 characters. These values are selected to maintain compatibility with the model's architectural constraints while balancing GPU memory usage and representational capacity.

Sentence preparation is conducted using the corresponding `.align` files provided with the GRID corpus, where frame-wise alignment is parsed to construct ground-truth transcriptions. An example of such an alignment is shown in Table 1, where each row corresponds to a segment of time during which a specific word (or silence) is spoken.

Table 1: Example of frame tuples extracted for sentence preparation.

Frame start-time (ms)	Frame end-time (ms)	Word
0	12750	`sil`
12750	19500	`bin`
19500	26750	`white`
26750	31250	`in`
31250	36750	`r`
36750	43500	`four`
43500	58500	`please`
58500	74500	`sil`

During preprocessing, non-verbal tokens such as `sil` (silence) and `sp` (short pause) are removed, and the remaining spoken words are concatenated and converted into character-level labels. These labels are subsequently used as supervision targets for training under the Connectionist Temporal Classification (CTC) loss function, allowing the model to learn flexible alignments between the visual input and the text output. This preprocessing pipeline ensures that the model receives a clean, colour-normalised, and temporally consistent sequence of maouth movements, thereby enhancing its ability to learn discriminative features necessary for accurate VSR.

3.3 Model Architecture

3.3.1 Depthwise Separable Convolutional Neural Network

The first component of the proposed architecture adopts a Depthwise Separable Convolutional Neural Network, specifically tailored for spatiotemporal data such as lip movements captured in video. This model is structured to enhance efficiency and feature disentanglement by factorising the standard convolutional operation into two

separate processes: depthwise convolution and pointwise convolution. The use of this formulation aligns with the design proposed by Guo et al. (2019), who demonstrated that such an approach significantly reduces parameter count while preserving the model's capacity to learn both spatial and channel-wise dependencies.

In contrast to standard 3D convolutions, which simultaneously capture both spatial and inter-channel correlations, depthwise separable convolutions perform these operations independently. The depthwise convolution applies a distinct 3D filter to each input channel to extract spatial-temporal features, while the pointwise convolution combines these outputs across channels via a $1 \times 1 \times 1$ convolution to model inter-channel relationships.

Formally, a standard 3D convolution applies a filter $\mathbf{K} \in \mathbb{R}^{W \times H \times D \times M \times N}$ to an input feature map $\mathbf{F} \in \mathbb{R}^{W' \times H' \times D' \times M}$ to produce an output $\mathbf{O} \in \mathbb{R}^{W' \times H' \times D' \times N}$, as shown in Equation (1):

$$O_{x,y,z,n} = \sum_{i,j,k,m} K_{i,j,k,m,n} \cdot F_{x+i,\, y+j,\, z+k,\, m} + b_n \tag{1}$$

where $O_{x,y,z,n}$ denotes the output value at spatial location (x, y, z) in output channel n, $K_{i,j,k,m,n}$ represents the filter weight connecting input channel m to output channel n at offset (i, j, k), $F_{x+i,y+j,z+k,m}$ is the input feature value, and b_n is the bias term associated with the output channel n.

In the depthwise separable formulation, this standard convolution is decomposed into two stages. First, the depthwise convolution applies a separate spatial filter to each input channel individually, as shown in Equation (2):

$$\hat{O}_{x,y,z,m} = \sum_{i,j,k} \hat{K}_{i,j,k,m} \cdot F_{x+i,\, y+j,\, z+k,\, m} + \hat{b}_m \tag{2}$$

where $\hat{O}_{x,y,z,m}$ is the intermediate output corresponding to input channel m, $\hat{K}_{i,j,k,m}$ is the depthwise kernel weight for channel m at spatial offset (i, j, k), $F_{x+i,y+j,z+k,m}$ denotes the input feature at the given location, and $\hat{b}_m$ is the bias term for each input channel.

Following the depthwise operation, a pointwise convolution is applied across channels to produce the final output, as shown in Equation (3):

$$O_{x,y,z,n} = \sum_{m} \tilde{K}_{m,n} \cdot \hat{O}_{x,y,z,m} + \tilde{b}_n \tag{3}$$

where $\tilde{K}_{m,n}$ represents the scalar pointwise kernel weight connecting input channel m to output channel n, $\hat{O}_{x,y,z,m}$ is the intermediate depthwise output, and $\tilde{b}_n$ denotes the bias term associated with each output channel.

This decomposition allows for a substantial reduction in the number of parameters compared to traditional convolutions. It also supports the design philosophy of separately modelling spatial correlations, which are captured by the depthwise filters, and cross-channel interactions, which are captured by the pointwise filters, a structure that Guo et al. (2019) argue more accurately reflects how information is organised across domains.

In the context of this dissertation, where video sequences encode subtle spatial and temporal variations in mouth movements, this formulation is particularly advantageous. It enables efficient extraction of low-level motion cues while retaining global channel-wise abstractions. Moreover, the separable design aligns with the overall goal of constructing a lightweight and modular visual encoder that integrates seamlessly with higher-level temporal modelling components.

3.3.2 Lightweight Patch Transformer

To efficiently capture both spatial and temporal dependencies in the input sequence, the proposed architecture integrates a Lightweight Patch Transformer block, composed of three primary components: efficient patch embedding, linear attention, and a compact feedforward module. The design is informed by principles from transformer-based architectures, but is specifically adapted to the visual speech domain where model size and computational cost are key constraints.

The first component is the efficient patch embedding layer, which projects high-dimensional spatiotemporal video frames into a compact token-based representation. Instead of densely sliding over the input as in traditional convolutions, this module adopts a large-stride 3D convolution to produce sparse, non-overlapping patches from the video volume. The motivation for this design follows from the findings of Phan et al. (2022), who demonstrated that patch embeddings can serve as effective local descriptors, preserving spatial semantics while enabling positional alignment across frames. By using fewer but larger patches, the model gains computational efficiency without sacrificing the ability to model localised motion features critical for lip-reading.

Once the visual tokens are generated, they are passed to a lightweight transformer block, where the self-attention mechanism is replaced by a linearised variant. Traditional transformers rely on full self-attention, which scales quadratically with respect to sequence length, posing memory and latency challenges during inference. To overcome this, the model incorporates a linear attention mechanism, inspired by the work of Ahn et al. (2023), which reduces the attention complexity to linear time by restructuring the attention computation into kernelised projections. This is achieved by applying a non-linearity to the queries and keys before aggregation, allowing the attention weights to be factorised and normalised efficiently. Such a formulation retains the global receptive field characteristic of transformers, while being significantly more scalable, particularly for long sequences like those encountered in video.

The attention block is then followed by a lightweight Feedforward Network, which consists of two linear layers separated by a Gaussian Error Linear Unit (GELU) activation. The GELU activation is chosen due to its superior mathematical properties and empirical performance in deep learning contexts. Unlike conventional activation functions such as Rectified Linear Unit (ReLU), which suffer from issues like non-differentiability and the dying neuron problem, GELU introduces smooth, non-linear behaviour that facilitates better gradient flow and optimisation stability. As demonstrated by Lee (2023), GELU is differentiable, bounded from below, and Lipschitz continuous, leading to well-behaved gradients and stable convergence during training. Their rigorous evaluation

across multiple benchmark datasets further confirmed GELU's consistent superiority over other activations, particularly in complex architectures involving residual and transformer-like designs. These properties make it a particularly fitting choice for the transformer block in this architecture, where smooth non-linear transformations are essential for learning expressive token-level representations.

The Feedforward Network serves to enrich the token representations by applying a learned transformation that increases the model's expressive power. This modular design mirrors the standard transformer architecture proposed by Vaswani et al. (2017), where self-attention and feedforward layers are alternated and normalised, but with modifications tailored for efficiency. In particular, a reduced Multi-Layer Perceptron (MLP) ratio is used to constrain the number of intermediate dimensions, thereby lowering the parameter count and training cost.

The overall structure of the Lightweight Patch Transformer balances global modelling capacity with architectural simplicity. By combining local patch extraction, linear attention, and compact feedforward processing, this component enables the model to learn effective temporal and spatial dependencies across the sequence. Furthermore, the resultant token embeddings are later fused with deep CNN features, offering a complementary representation that integrates both local motion cues and global contextual patterns.

This hybrid approach leverages the benefits of patch-wise attention and efficient computation, demonstrating that transformer-based representations can be made tractable for lip-reading applications without sacrificing model performance.

3.3.3 Bidirectional Gated Recurrent Unit

To model temporal dependencies across visual speech sequences, the proposed architecture incorporates a BiGRU module following the visual encoding stages. The rationale behind selecting GRUs over other recurrent architectures—such as standard RNNs and LSTMs—is comprehensively discussed in Section 2.3, which outlines the empirical and theoretical motivations drawn from prior research. As reviewed in that section, GRUs offer a simplified yet effective gating mechanism that balances model complexity and performance, while the bidirectional extension enables the architecture to capture both historical and future context in video sequences.

The architectural contrast between standard LSTMs, and GRUs is illustrated in Figure 2, which visualises the simplified gate structure of GRUs in comparison to the more complex LSTM design.

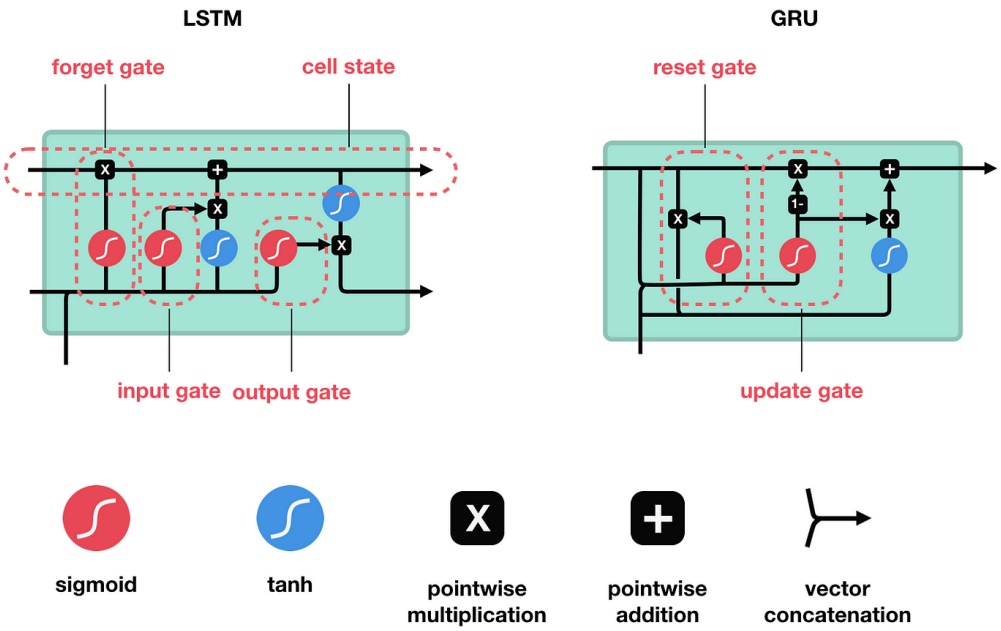

Figure 2: Comparison between LSTM, and GRU architectures.

Mathematically, the GRU computes its hidden states following the steps as shown in Equations (4) to (7):

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \tag{4}$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \tag{5}$$

$$\tilde{h}_t = \tanh(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h) \tag{6}$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \tag{7}$$

where x_t denotes the input at time step t, and h_{t-1} and h_t represent the hidden states at the previous and current time steps, respectively. The variable z_t is the update gate, which controls how much of the previous state is carried forward, while r_t is the reset gate that determines how much of the previous state contributes to the candidate activation. The term $\tilde{h}_t$ represents the candidate hidden state, computed by applying a non-linear transformation to the combination of the input and the reset-modified previous state. The final hidden state h_t is then calculated as a convex combination of the previous hidden state h_{t-1} and the candidate state $\tilde{h}_t$, weighted by the update gate. The operator $\odot$ denotes element-wise (Hadamard) multiplication, and W_*, U_*, and b_* refer to learnable weight matrices and bias terms for each respective gate.

In a BiGRU, two such layers are instantiated—one processing the sequence forward in time and the other in reverse. The outputs from both directions are then concatenated at each time step, producing a rich representation that incorporates both historical and anticipatory information. This bidirectional formulation is especially important in lip-reading tasks, where the interpretation of a phoneme or viseme often depends on preceding and succeeding mouth movements.

The integration of BiGRUs thus offers a robust temporal modelling backbone, complementing the spatial representations learned by earlier CNN and transformer components in the architecture.

3.3.4 Proposed Architecture 1: Light Patch Embedding Transformer Network (LPETNet)

LPETNet is a hybrid architecture that integrates both convolutional and transformer-based modules to effectively capture spatial and temporal dependencies in visual speech data. The input to the model consists of RGB video frames cropped to the mouth region, which are uniformly resized from an initial spatial resolution of 80×160 to 64×128 prior to being processed by any network layer. This resizing reduces computational load while retaining sufficient spatial detail for learning discriminative features. The model comprises three primary processing stages: depthwise separable 3D CNN for local spatiotemporal feature extraction, lightweight transformer for global sequence modelling, and BiGRU module for temporal aggregation. A schematic of LPETNet is presented in Figure 3.

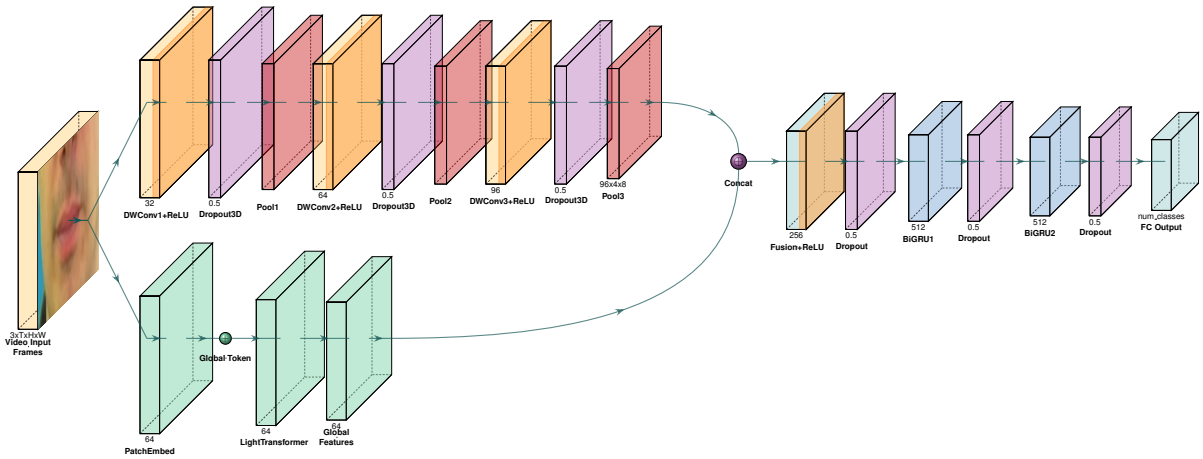

Figure 3: Schematic architecture of the proposed LPETNet model.

The input to the network is a sequence of RGB video frames cropped to the mouth region, each of dimension $3 \times 75 \times 64 \times 128$ (channel $\times$ time $\times$ height $\times$ width). The CNN pathway begins with three consecutive layers of depthwise separable 3D convolutions, which have been shown to provide efficient spatial encoding while reducing the number of parameters significantly, as supported by Guo et al. (2019). The first convolutional block projects the input to 32 channels, followed by a max pooling layer with kernel size $(1, 2, 2)$, reducing the spatial resolution while preserving temporal information. The second and third convolutional blocks increase the feature dimension to 64 and 96 channels respectively, each followed by 3D dropout and max pooling layers. The hyperparameter choices in this convolutional backbone, including kernel sizes, strides, and pooling configurations, are adopted from the baseline architecture proposed in the LipNet model by Assael et al. (2016), which this dissertation extends and builds upon. These layers contribute to a final CNN feature map of shape $[C, T \cdot H \cdot W] = [96, 75 \cdot 4 \cdot 8]$.

Parallel to the CNN pipeline, the transformer pathway processes the input via an efficient patch embedding mechanism. This module applies a 3D convolution with a kernel size of 12×12 and stride 12, projecting the input into non-overlapping localised patches. The choice of a 12×12 kernel is motivated by the work of Chen et al. (2021), who demonstrated that larger patch kernels can effectively capture broader spatial

contexts while reducing computational load in transformer-based vision architectures. Meanwhile, the stride value of 12 is selected in accordance with the findings of Yin et al. (2024), who highlighted that increased stride in patch embedding improves the receptive field and enhances the local continuity of extracted features, particularly in scenarios where texture-preserving representations are critical. Each patch is then normalised and transformed into a token sequence, and a learnable global token is prepended to each sequence, capturing global context across the video.

The sequence of patch tokens is then passed through a lightweight transformer block utilising linear attention, which scales linearly with sequence length instead of quadratically as in traditional attention mechanisms. This efficient design follows from the formulation in Ahn et al. (2023), allowing for scalable attention computation without compromising performance. A feedforward module with a reduced MLP ratio of 2 is used, incorporating GELU activation for smooth non-linearity, as recommended by Lee (2023) due to its favourable optimisation properties. To enhance regularisation and prevent overfitting, a dropout rate of 0.1 is applied within the lighweight transformer block, consistent with the configuration proposed by Vaswani et al. (2017). The transformer's output is reduced to a single vector by extracting the global token for each time step.

The CNN outputs undergo flattening across the spatial dimensions, converting each 3D activation map into a single feature vector per timestep. This ensures compatibility with the transformer-derived global token representations, which are temporally aligned. The resulting outputs of the CNN and transformer pathways are then concatenated along the feature dimension and projected via a fusion layer into a 256-dimensional latent space. The use of a 256-dimensional projection, along with the application of ReLU activation, is consistent with the architectural design introduced by Assael et al. (2016), on which this dissertation builds. ReLU serves as an effective non-linearity that preserves gradient flow while enabling sparse activations.

To mitigate overfitting, dropout with a probability of 0.5 is employed following the fusion and recurrent layers. This choice is informed by the findings of Srivastava et al. (2014), who demonstrated that a dropout rate of 0.5 is near-optimal across a wide range of deep learning architectures and domains, as it balances the trade-off between model robustness and training convergence.

Finally, the model outputs are passed through a linear classification layer projecting to 28 output classes, corresponding to the character set used in the CTC loss formulation. The output tensor is transposed to shape $[B, T, C]$ to conform to the requirements of CTC-based training. This architecture balances efficiency and expressivity, combining lightweight visual encoding, global attention, and recurrent sequence modelling to achieve robust lip-reading performance on sentence-level visual speech datasets. Table 2 shows all the transformations and hyperparameters for the proposed LPETNet model.

Table 2: LPETNet model hyperparameters

Layer	Size / Stride / Pad	Input Size	Dimension Order
Input (RGB Frames)	–	$3 \times 75 \times 64 \times 128$	$C \times T \times H \times W$
Depthwise Separable Conv3D-1	$3 \times 5 \times 5$ / $(1,2,2)$ / $(1,2,2)$	$3 \times 75 \times 64 \times 128$	$C \times T \times H \times W$
MaxPool3D-1	$1 \times 2 \times 2$ / $(1,2,2)$	$32 \times 75 \times 32 \times 64$	$C \times T \times H \times W$
Depthwise Separable Conv3D-2	$3 \times 5 \times 5$ / $(1,1,1)$ / $(1,2,2)$	$32 \times 75 \times 32 \times 64$	$C \times T \times H \times W$
MaxPool3D-2	$1 \times 2 \times 2$ / $(1,2,2)$	$64 \times 75 \times 16 \times 32$	$C \times T \times H \times W$
Depthwise Separable Conv3D-3	$3 \times 3 \times 3$ / $(1,1,1)$ / $(1,1,1)$	$64 \times 75 \times 16 \times 32$	$C \times T \times H \times W$
MaxPool3D-3	$1 \times 2 \times 2$ / $(1,2,2)$	$96 \times 75 \times 8 \times 16$	$C \times T \times H \times W$
Patch Embedding (Transformer)	12×12 / 12	$3 \times 75 \times 64 \times 128$	$C \times T \times H \times W$
Transformer (Linear Attention)	$p = 0.1$	$75 \times 50 \times 64$	$T \times P \times E$
Extracted Global Token	–	75×64	$T \times E$
CNN Feature Flattening	–	$96 \times 75 \times 4 \times 8$	$C \times T \times H \times W$
Concat (CNN + Transformer)	–	75×3072, 75×64	$T \times F$
Fusion FC + ReLU + Dropout	256 , $p = 0.5$	75×3136	$T \times F$
BiGRU Layer 1	256	75×256	$T \times F$
Dropout	$p = 0.5$	75×512	$T \times F$
BiGRU Layer 2	256	75×512	$T \times F$
Dropout	$p = 0.5$	75×512	$T \times F$
FC Output Layer	$27 + 1(blank)$ classes	75×512	$T \times F$
Softmax	–	75×28	$T \times C_n$

The symbolic notations used in Table 2 are defined as follows. The variable C denotes the number of input channels (typically 3 for RGB video frames), T represents the temporal sequence length (75 frames per sample), and H and W refer to the spatial height and width of each frame, which are progressively reduced through the convolutional and pooling layers. The symbol P indicates the number of non-overlapping spatial patches extracted per frame by the patch embedding module, and E denotes the embedding dimension of each patch token. The variable F denotes the feature dimension. The variable p denotes the dropout probability. Finally, $C_n = 28$ denotes the total number of output classes, comprising 27 valid character labels and 1 additional class reserved for the CTC blank token.

3.3.5 Proposed Architecture 2: Attention-reinforced Light Patch Embedding Transformer Network (ALPETNet)

ALPETNet builds upon the LPETNet architecture by incorporating additional attention-based components aimed at enhancing spatiotemporal feature integration. The key modifications include: (1) channel attention applied after each depthwise separable convolutional block, (2) a multi-scale patch embedding module for extracting diverse local patterns, (3) a gated transformer block with a learnable feature modulation gate, and (4) a cross-attention fusion mechanism where transformer-derived global context guides convolutional feature aggregation.

The channel attention mechanism, inspired by squeeze-and-excitation networks introduced by Hu et al. (2018), is applied after each convolutional block. It adaptively recalibrates feature responses by explicitly modelling inter-channel dependencies, allowing the network to emphasise informative features while suppressing redundant ones. This improves spatial feature encoding across the convolutional hierarchy.

Inspired by the recent work of Liu et al. (2024), the proposed model incorporates a multi-scale patch embedding module that facilitates hierarchical feature extraction across varied spatial resolutions. Specifically, patches of sizes 4×4, 8×8, and 12×12 are extracted from each frame using separate convolutional projections, each with a corresponding stride to ensure non-overlapping coverage. The resulting patch tokens from each resolution are then concatenated along the token dimension and passed through a shared normalisation layer to form a unified sequence embedding. This architectural design enables the transformer to jointly process fine- and coarse-grained visual features within a single pass, thereby enhancing its capacity to generalise across diverse lip shapes, speaking styles, and articulatory dynamics. As highlighted by Liu et al. (2024), multi-scale token representations improve the model's adaptability to varying spatial contexts, which is particularly advantageous in lip reading where both subtle local movements and broader spatial cues contribute to accurate interpretation.

A gated transformer block, inspired by the design proposed in Wu et al. (2023), replaces the standard transformer in this proposed architecture. In addition to the efficiency of linear attention, the block introduces a learnable gating mechanism that modulates the output of the feedforward MLP branch. This gating function, defined as a sigmoid-controlled interpolation between the transformed features and their residual connection, enables dynamic control over feature propagation. To mitigate overfitting and encourage regularisation, a dropout rate of 0.1 is applied within the gated transformer block, consistent with the configuration proposed by Vaswani et al. (2017). The result is improved gradient flow, enhanced representation expressivity, and robustness to noisy inputs, as detailed in their ablation studies. By selectively filtering relevant information while suppressing redundant or irrelevant features, the gated design helps stabilise the training process and contributes to more discriminative representations in sequential modelling tasks.

The outputs of the CNN pathway are first flattened across their spatial dimensions to produce a sequence of feature vectors, each corresponding to a single timestep. This transformation ensures structural compatibility with the temporally aligned transformer tokens. To facilitate hybrid feature fusion integration, a cross-attention fusion mechanism is then employed. This module projects CNN and transformer features into a shared latent space and uses self-attention weights derived from transformer tokens to modulate CNN outputs. This guides the fusion process, ensuring that temporal patterns identified by the transformer influence the spatiotemporal information encoded by CNNs. The design of this fusion block draws inspiration from the cross-attention strategy introduced by Chen et al. (2021), where the class token from one transformer branch attends to patch tokens of another branch. This design enables efficient exchange of multi-scale information while maintaining computational efficiency, and in this case, it enhances the selective integration of global transformer context into local CNN encodings.

The complete model architecture, integrating these components, is illustrated in Figure 4. As in LPETNet, the fused features are passed through stacked BiGRU layers followed by a linear classification layer with 28 output classes (27 characters and 1 CTC blank token).

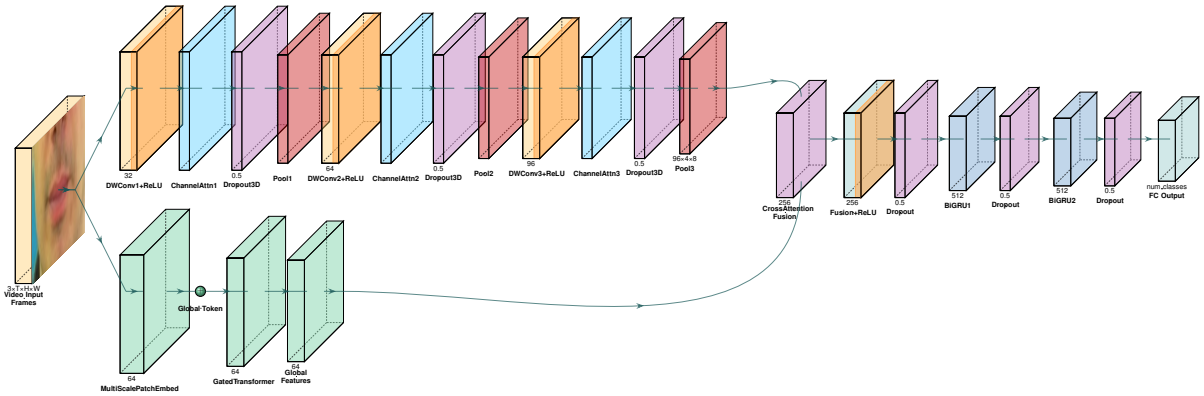

Figure 4: Schematic architecture of the proposed ALPETNet model.

All transformations and model hyperparameters are summarised in Table 3.

Table 3: ALPETNet model hyperparameters

Layer	Size / Stride / Pad	Input Size	Dimension Order
Input (RGB Frames)	–	$3 \times 75 \times 64 \times 128$	$C \times T \times H \times W$
Depthwise Separable Conv3D-1 + Channel Attention	$3 \times 5 \times 5$ / $(1, 2, 2)$ / $(1, 2, 2)$	$3 \times 75 \times 64 \times 128$	$C \times T \times H \times W$
MaxPool3D-1	$1 \times 2 \times 2$ / $(1, 2, 2)$	$32 \times 75 \times 32 \times 64$	$C \times T \times H \times W$
Depthwise Separable Conv3D-2 + Channel Attention	$3 \times 5 \times 5$ / $(1, 1, 1)$ / $(1, 2, 2)$	$32 \times 75 \times 32 \times 64$	$C \times T \times H \times W$
MaxPool3D-2	$1 \times 2 \times 2$ / $(1, 2, 2)$	$64 \times 75 \times 16 \times 32$	$C \times T \times H \times W$
Depthwise Separable Conv3D-3 + Channel Attention	$3 \times 3 \times 3$ / $(1, 1, 1)$ / $(1, 1, 1)$	$64 \times 75 \times 16 \times 32$	$C \times T \times H \times W$
MaxPool3D-3	$1 \times 2 \times 2$ / $(1, 2, 2)$	$96 \times 75 \times 8 \times 16$	$C \times T \times H \times W$
Multi-Scale Patch Embedding	$\{4, 8, 12\} \times \{4, 8, 12\}$ / $\{4, 8, 12\}$	$3 \times 75 \times 64 \times 128$	$C \times T \times H \times W$
Gated Transformer Block	$p = 0.1$	$75 \times \{512, 128, 50\} \times 64$	$T \times P \times E$
Extracted Global Token	–	75×64	$T \times E$
CNN Feature Flattening	–	$96 \times 75 \times 4 \times 8$	$C \times T \times H \times W$
Cross-Attention Fusion	–	75×3072, 75×64	$T \times F$
Fusion FC + ReLU + Dropout	256, $p = 0.5$	75×256	$T \times F$
BiGRU Layer 1	256	75×256	$T \times F$
Dropout	$p = 0.5$	75×512	$T \times F$
BiGRU Layer 2	256	75×512	$T \times F$
Dropout	$p = 0.5$	75×512	$T \times F$
FC Output Layer	$27 + 1$(blank) classes	75×512	$T \times F$
Softmax	–	75×28	$T \times C_n$

The symbolic notations used in Table 3 are defined as follows. C denotes the number of input channels (3 for RGB), T is the temporal sequence length, and H and W are the frame height and width. P represents the number of patches extracted from each frame by the multi-scale patch embedding module, while E refers to the embedding dimension of each token. The variable F denotes the feature dimension. The variable p denotes the dropout probability. $C_n = 28$ is the number of output classes, including 27 characters and one blank token for the CTC loss.

3.4 Training Process

All training experiments in this dissertation are conducted on a high-performance desktop workstation equipped with a 13th Gen Intel® Core™ i5-13600KF processor operating at 3.50 GHz, 64 GB of installed RAM, and an NVIDIA GeForce RTX 4060 Ti graphics card with 16 GB of dedicated VRAM. This configuration provides the neces-

sary computational resources for training transformer-augmented deep learning models while maintaining high throughput and memory efficiency. All models are implemented using the PyTorch framework, which is chosen over Keras due to its greater flexibility, dynamic computation graph capabilities, and stronger community support for research-grade custom architectures such as gated transformers and multi-scale patch embeddings.

3.4.1 Data Augmentation

To enhance generalisation and reduce the risk of overfitting, a horizontal flipping strategy is applied as a form of data augmentation during training. With a probability of 0.5, the spatial orientation of the video frames is horizontally mirrored while preserving the temporal order of the sequence. This transformation introduces spatial variability, enabling the model to become more robust to differences in lip orientation and speaker pose. Although relatively simple, this augmentation has proved effective in encouraging the network to learn position-invariant visual features. When combined with regularisation techniques such as dropout, this approach contributes to more stable training and improved performance on validation data.

3.4.2 Connectionist Temporal Classification

CTC, introduced by Graves (2012), is an alignment-free, non-autoregressive method widely used for sequence transduction tasks such as speech recognition, handwriting recognition, and VSR. It is particularly effective in scenarios where the alignment between input and output sequences is unknown or variable in length, such as mapping sequences of video frames to spoken words.

The CTC loss operates independently of the model's internal structure and is applied at the output level to guide the mapping of input sequences to label sequences. At each timestep, the model produces a probability distribution over a set of output labels $\mathcal{A}$ augmented with a special blank token ϕ, forming the extended label set $\mathcal{A}' = \mathcal{A} \cup \{\phi\}$. These predictions are typically the output of a softmax layer. The presence of the blank token allows the model to handle cases where multiple input frames correspond to a single target character or where no meaningful output is present at a given frame. In this dissertation, the alphabet $\mathcal{A}$ consists of the 26 lowercase English letters along with a word boundary symbol (typically represented as a space or separator), forming a compact label vocabulary for sentence-level VSR.

The probability of the target sequence z given the input sequence x is computed in CTC as the sum over all valid alignment paths π that collapse to z under a mapping function F, as shown in Equation (8):

$$\mathcal{L}(x, z) = -\log p(z|x), \qquad \text{where} \quad p(z|x) = \sum_{\pi \in F^{-1}(z)} \prod_{t=1}^{T} y_t^{\pi_t} \tag{8}$$

where x denotes the input sequence of length T, z represents the target sequence of length U, and $\pi \in \mathcal{A}'^T$ is an alignment path over the extended alphabet $\mathcal{A}' = \mathcal{A} \cup \{\phi\}$, where $\mathcal{A}$ consists of the 26 lowercase English letters together with a word boundary symbol, and ϕ is the blank token.

The function $F(\pi)$ refers to the collapsing operation that maps an alignment path π to a final label sequence z by first removing consecutive repeated labels and then eliminating any blank symbols. In Equation (8), the summation $\sum_{\pi \in F^{-1}(z)}$ aggregates over all alignment paths π such that collapsing via $F(\pi)$ yields the target sequence z.

In practice, the collapsing operation is exemplified by the transcription of the word `one`, as shown in Equation 9:

$$\pi = (o, o, \phi, \phi, \phi, \phi, \phi, \phi, \phi, n, n, \phi, \phi, e) \xrightarrow{F} z = (o, n, e) \tag{9}$$

where the mapping function F removes all blank symbols ϕ and collapses consecutive repeated labels, resulting in the correct output sequence `one`.

The term $y_t^{\pi_t}$ corresponds to the model's predicted probability of emitting symbol π_t at time step t. The product $\prod_{t=1}^{T} y_t^{\pi_t}$ computes the probability of a particular alignment path, and the outer summation marginalises over all such paths that are consistent with the final target sequence.

Each π is a path of length T drawn from the set $\mathcal{A}'^T$, and since the true alignment between input and output is unknown, this formulation allows the model to consider all valid alignments during training. This probabilistic flexibility enables the network to learn correct label transitions over time without needing explicit frame-level supervision.

The CTC layer outputs a continuous distribution over the extended alphabet at each timestep and learns to produce coherent character sequences. Gradients are efficiently computed using the forward-backward algorithm, and model parameters are updated through backpropagation, making CTC a scalable and effective alternative to autoregressive decoders, especially in tasks where the input sequence is longer than the output.

In this dissertation, the PyTorch-native `CTCLoss` is used during training, and the output is decoded using the `ctc_decoder` module from `torchaudio.models.decoder`, which supports both beam search and lexicon-based decoding strategies for more robust inference. A comparative evaluation of different decoding strategies—including greedy decoding, pure beam search, and character-based beam search—is presented in Section 4.3, where their performance is systematically analysed within the context of lipreading.

3.4.3 Adam Optimiser

The model is trained using the Adam optimiser, a first-order stochastic optimisation method introduced by Kingma and Ba (2014). Adam, short for Adaptive Moment Estimation, extends stochastic gradient descent by adapting individual learning rates for

each parameter based on estimates of the first and second moments of the gradients. This makes it particularly well-suited for problems involving sparse gradients or complex model architectures.

At each training step t, given the gradient of the loss function with respect to model parameters, $g_t = \nabla_{\theta_t} \mathcal{L}_t$, Adam updates two exponentially decaying moving averages, as shown in Equations (10) and (11):

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{10}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \tag{11}$$

where m_t denotes the first moment estimate, namely the exponentially weighted moving average of past gradients, and v_t denotes the second moment estimate, representing the exponentially weighted moving average of the squares of past gradients. The parameters β_1 and β_2 are hyperparameters controlling the decay rates of these moving averages, typically set close to 1 such as $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The quantity g_t is the gradient of the loss $\mathcal{L}_t$ with respect to the model parameters θ_t at time step t.

Because m_t and v_t are initially biased towards zero, Adam applies a bias correction to compensate for their underestimation during early iterations, as shown in Equations (12) and (13):

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \tag{12}$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \tag{13}$$

where $\hat{m}_t$ and $\hat{v}_t$ denote the bias-corrected estimates of the first and second moments, respectively, and t is the current iteration number used to normalise the moving averages appropriately.

The model parameters are subsequently updated according to the rule shown in Equation (14):

$$\theta_{t+1} = \theta_t - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} \tag{14}$$

where θ_t and θ_{t+1} refer to the model parameters at time steps t and $t + 1$, respectively. The learning rate α controls the magnitude of the update step, while ϵ is a small positive constant introduced to ensure numerical stability by preventing division by zero during the update.

Thus, the operations described in Equations (10) to (14) collectively define the complete Adam optimisation algorithm, enabling efficient and adaptive learning even in the presence of noisy or sparse gradients.

In this dissertation, Adam is used with its default parameters: $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$, as recommended by Kingma and Ba (2014). These values are known to offer a good balance between convergence speed and stability, making them especially effective for deep learning models with multiple components, such as convolutional encoders, transformer-based attention blocks, and recurrent sequence models.

3.4.4 Training Stability and Loss Convergence

The training dynamics of the two proposed models—LPETNet and ALPETNet—are visualised through their training and validation loss curves as shown in Figure 5 and Figure 6, respectively. Both models are trained over 120 epochs using a batch size of 64.

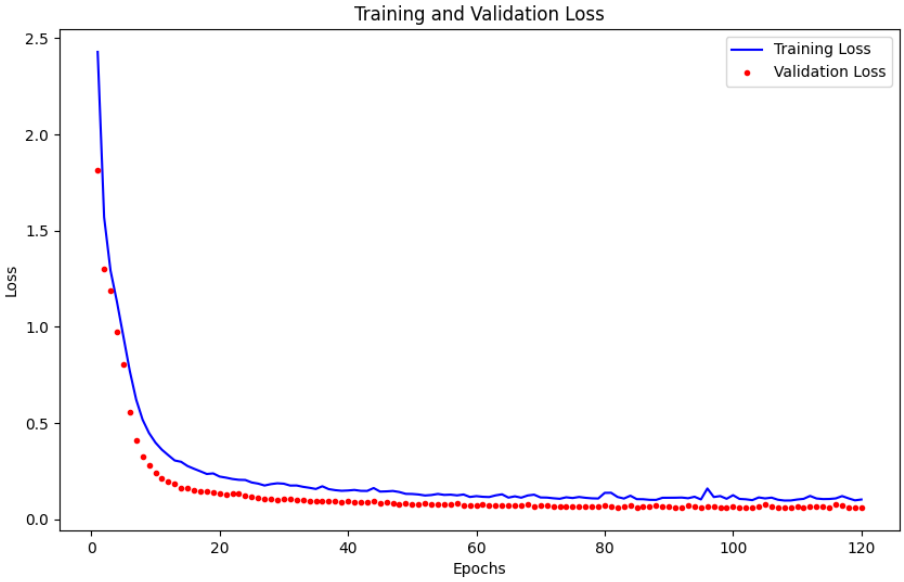

Figure 5: Training and validation loss curves for LPETNet trained over 120 epochs.

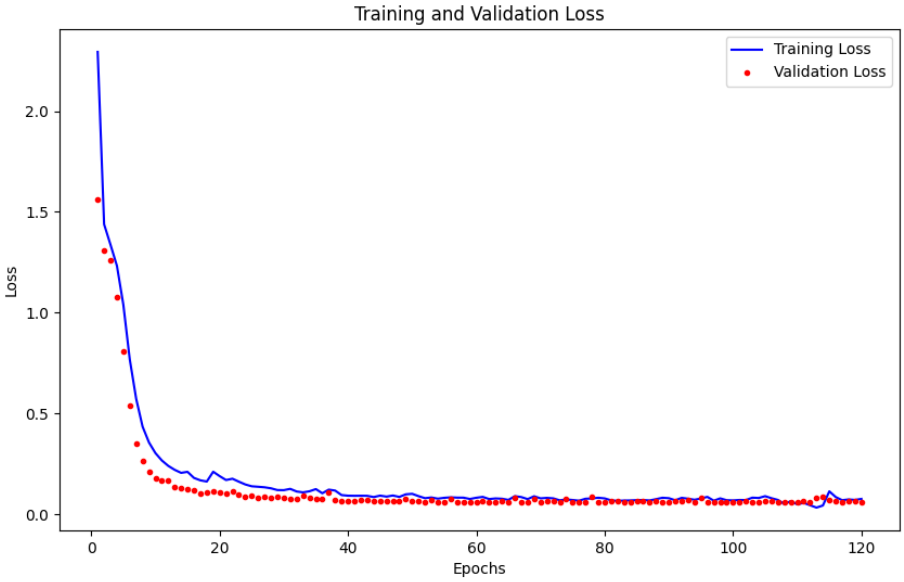

Figure 6: Training and validation loss curves for ALPETNet trained over 120 epochs.

Both loss curves demonstrate smooth and stable convergence, with the validation loss closely following the training loss throughout the training process. This behaviour suggests that neither model is prone to overfitting, as the validation performance continues to improve alongside training without sharp divergence or plateauing.

For LPETNet, the validation loss stabilises early and exhibits only minor fluctuations, indicating that the model generalises well despite its relatively lightweight architecture. In contrast, ALPETNet shows an even closer alignment between training and validation curves, highlighting the benefits of the added architectural components—such as channel attention, cross-attention fusion, and gated transformer blocks. These enhancements appear to contribute to more effective gradient flow and better regularisation, helping the model avoid overfitting even as it captures more complex dependencies.

The observed loss trajectories reinforce the overall stability of the training process and confirm the suitability of the optimisation strategies employed. Notably, the consistent downward trend in validation loss across epochs for both models suggests that the applied dropout mechanisms and data augmentation are successful in promoting generalisation under the CTC loss framework.

3.5 Chapter Summary

This chapter has presented the dataset, preprocessing pipeline, and the design of two novel architectures—LPETNet and ALPETNet—for VSR. Building on previous models from the literature, these architectures introduce lightweight yet expressive components such as depthwise convolutions, patch transformers, and gated attention. The models are trained using CTC loss and optimised with Adam, demonstrating stable convergence and effective generalisation.

The next chapter, Chapter 4, builds upon this foundation by comparing LPETNet and ALPETNet against existing baselines and evaluating their decoding performance across varying beam widths and strategies. Metrics such as WER, Character Error Rate (CER), BLEU(Bilingual Evaluation Understudy), and decoding time are used to assess model accuracy and efficiency.

4 Results and Discussions

This chapter presents a comprehensive evaluation of the proposed VSR models—LPETNet and ALPETNet—across multiple decoding strategies, performance metrics, and interpretability analyses. It begins with an outline of the evaluation framework, introducing WER, CER, and BLEU score as key metrics in Section 4.1. A comparative study with existing lip-reading models is presented in Section 4.2, highlighting how the proposed architectures align with or improve upon established baselines. The decoding performance of LPETNet and ALPETNet is then systematically examined under varying beam widths and inference strategies in Section 4.3, revealing key trade-offs between accuracy and computational cost. Section 4.4 analyses character-level prediction errors via confusion matrices, while Section 4.5 extends this analysis to the phoneme level using viseme-aware clustering. Finally, Section 4.6 employs saliency map visualisation to interpret the spatiotemporal regions most influential to the model's predictions.

4.1 Evaluation Metrics

To assess the performance of the proposed models, three widely adopted evaluation metrics are used: CER, WER, and the BLEU score. These metrics, presented in Equations (15), (16), and (17), provide a comprehensive evaluation framework that captures transcription accuracy at both character and word levels, as well as the semantic similarity of predicted sequences to the ground truth.

Character Error Rate. The CER quantifies the model's accuracy at the character level, offering fine-grained insight into transcription performance. It is calculated as shown in Equation (15):

$$CER = \frac{S + D + I}{C}, \tag{15}$$

where S is the number of character substitutions, D is the number of deletions, I is the number of insertions, and C is the total number of characters in the reference transcription.

Word Error Rate. WER measures sentence-level transcription accuracy by comparing the predicted and reference word sequences. It is calculated as shown in Equation (16):

$$WER = \frac{S + D + I}{W}, \tag{16}$$

where S, D, and I represent the number of word-level substitutions, deletions, and insertions respectively, and W is the number of words in the reference sentence.

BLEU Score. The BLEU score, introduced by Papineni et al. (2002), evaluates the quality of text generation using modified n-gram precision and a brevity penalty to penalise overly short predictions. In this dissertation, sentence-level BLEU is computed using the implementation provided by the Natural Language Toolkit (NLTK) from Bird

(2006). Given that the smallest units in the GRID corpus vocabulary include single-character words such as letters (“h”, “k”), this dissertation employs unigram BLEU ($n = 1$) to ensure fair evaluation. The sentence-level BLEU score is calculated as shown in Equation (17):

$$\text{BLEU} = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right), \tag{17}$$

where p_n denotes the modified n-gram precision, w_n is the weight assigned to n-grams (uniform in this case), and BP is the brevity penalty, defined as shown in Equation (18):

$$BP = \begin{cases} 1 & \text{if } c > r \\ \exp\left(1 - \frac{r}{c}\right) & \text{if } c \leq r \end{cases}, \tag{18}$$

where c represents the length of the candidate (predicted) sentence and r the length of the reference sentence. This formulation ensures that the score reflects both lexical precision and output length adequacy, making it suitable for evaluating sentence-level semantic fidelity.

4.2 Comparative Analysis with Existing Models

To benchmark the performance of the proposed architectures, this section presents a comparative evaluation against several well-established lip-reading models in the literature, all tested on the GRID corpus. Among these, LipNet (Assael et al., 2016) set the foundation for end-to-end sentence-level VSR using spatiotemporal convolutions and BiGRUs. HLRNet (Sarhan et al., 2021) enhanced this baseline by introducing hierarchical receptive fields and inception-style blocks to capture richer spatiotemporal patterns. LCANet (Xu et al., 2018) advanced the architecture further by proposing two variants—A-CTC and AH-CTC—integrating attention and highway connections to improve sequence modelling. More recently, Resformer (Xue et al., 2023) pushed the state of the art by combining a ResNet backbone with Transformer encoders and self-distillation mechanisms, achieving highly competitive performance.

Table 4 summarises the CER and WER achieved by each of these models, alongside the results of the proposed LPETNet and ALPETNet architectures. Due to the absence of BLEU score reporting in several prior studies, BLEU evaluations have been omitted from this table to ensure consistent and fair comparison across all models.

Table 4: Comparison of CER and WER between existing lip-reading models and the proposed LPETNet and ALPETNet architectures.

Model	CER (%)	WER (%)
LipNet	1.9	4.8
HLRNet	1.4	3.3
A-CTC	1.7	4.1
AH-CTC (LCANet)	1.3	2.9
Resformer	**0.4**	**1.7**
LPETNet (Proposed Architecture 1)	2.1	5.4
ALPETNet (Proposed Architecture 2)	1.7	4.1

The results indicate that while Resformer achieves the lowest error rates—0.4% CER and 1.7% WER—it does so with a much more complex architecture that includes deep residual blocks and transformer stacks, leading to increased computational cost and memory demands. In contrast, the proposed ALPETNet model matches the WER of A-CTC at 4.1% and achieves a competitive CER of 1.7%, all while maintaining a significantly more lightweight architecture.

This performance is particularly noteworthy given ALPETNet's use of efficient components such as depthwise separable convolutions, channel attention, and gated transformer blocks with linear attention. These design choices allow it to capture both local and global dependencies without incurring the resource overhead typically associated with large-scale transformer models. Meanwhile, LPETNet—lacking channel attention and cross-attention mechanisms records a higher CER of 2.1% and WER of 5.4%, highlighting the added value of attention-based enhancements in ALPETNet.

Overall, the results validate the effectiveness of the proposed architecture. Despite being computationally efficient, ALPETNet maintains strong transcription accuracy, positioning it as a practical candidate for real-time or resource-constrained VSR deployments.

4.3 Comparative Analysis between Decoding Strategies

To further evaluate model performance during inference, a comparative analysis of different CTC decoding strategies is conducted. These included greedy decoding, pure beam search, and character-based language model (char-LM) beam search. The experiments are run separately for LPETNet and ALPETNet architectures across beam widths, $\lambda \in \{1, 2, 4, 8, 16\}$.

The character-based language model is implemented using the KenLM toolkit, which is only available on Unix-based platforms such as Linux and macOS. This integration allows the decoder to incorporate external language models by adjusting the scoring of CTC paths based on linguistic fluency. Two key hyperparameters regulate the decoding process: α, which controls the weight of the language model, and β, which

penalises longer sequences. In all evaluations, CTC decoding parameters, α and β are set to 1.0 and 1.5 respectively to balance language bias and output length.

In CTC decoding, beam width, λ determines the number of top candidate paths retained at each timestep. A higher λ enables the decoder to explore a wider hypothesis space, often improving accuracy at the expense of computational efficiency. Greedy decoding corresponds to the limiting case of $\lambda = 1$ with no language model applied.

LPETNet Decoding Performance

Decoding performance for the LPETNet model is presented in Table 5, which reports WER, CER, and BLEU scores across both pure beam search and char-LM decoding strategies over various beam widths (λ). Table 6 complements this by comparing the corresponding decoding times. The trends in error rates are visualised in Figures 7 and 8, while decoding runtimes are shown in Figure 9.

Table 5: LPETNet: Decoding accuracy across strategies and beam widths, (λ).

Strategy	beam widths (λ)	WER (%)	CER (%)	BLEU
Greedy (Pure Beam)	**1**	**5.40**	**2.10**	**0.9460**
Pure Beam	2	5.40	2.10	0.9460
Pure Beam	4	5.40	2.10	0.9460
Pure Beam	8	5.40	2.10	0.9460
Pure Beam	16	5.40	2.10	0.9460
Char-LM	1	5.96	2.23	0.9404
Char-LM	2	5.58	2.13	0.9442
Char-LM	4	5.56	2.12	0.9445
Char-LM	8	5.54	2.12	0.9446
Char-LM	16	5.54	2.12	0.9446

Table 6: LPETNet: Decoding time (in seconds) across strategies and beam widths (λ).

Strategy	beam widths (λ)	Time (s)
Greedy (Pure Beam)	**1**	**130.62**
Pure Beam	2	158.80
Pure Beam	4	158.77
Pure Beam	8	158.84
Pure Beam	16	159.74
Char-LM	1	158.95
Char-LM	2	158.35
Char-LM	4	159.54
Char-LM	8	159.44
Char-LM	16	160.87

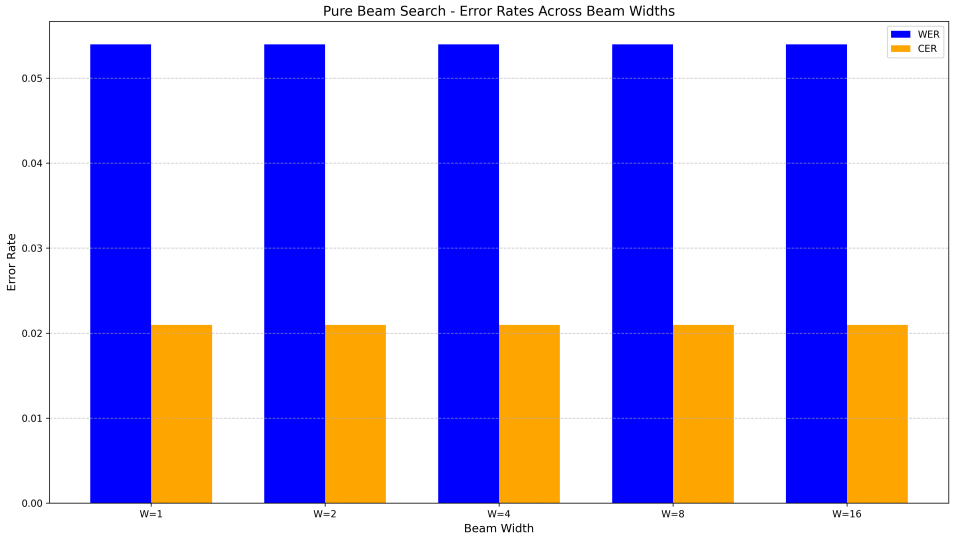

Figure 7: LPETNet: Pure beam search error rates across beam widths (λ).

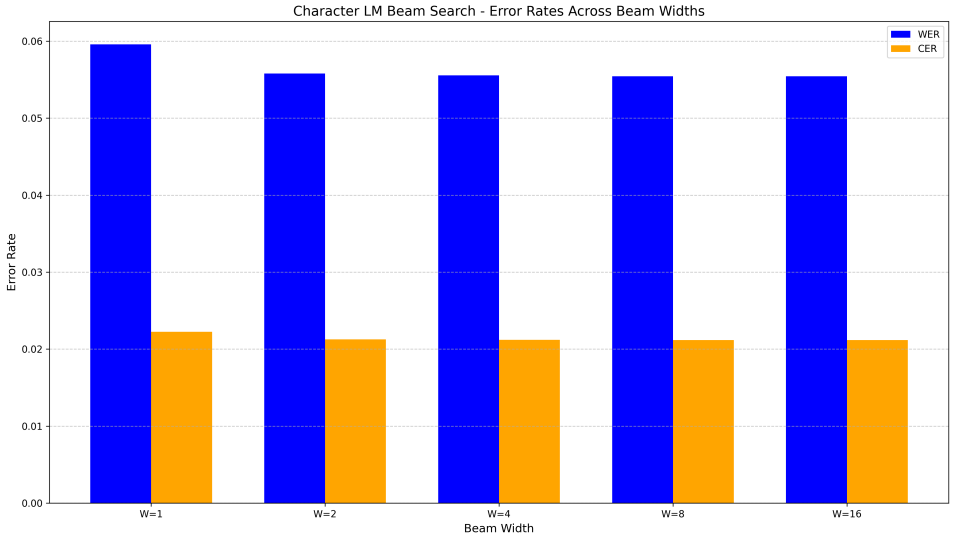

Figure 8: LPETNet: Character LM beam search error rates across beam widths (λ).

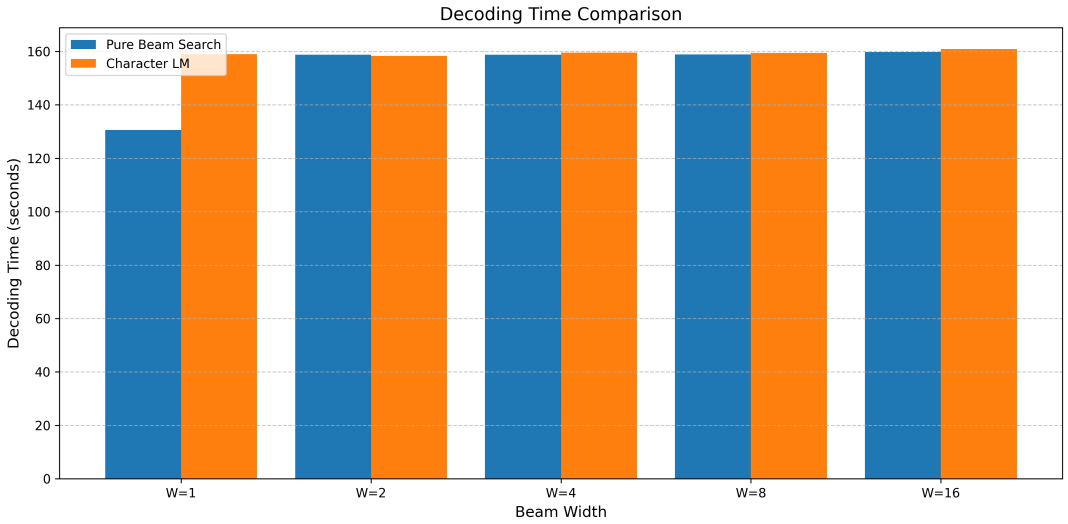

Figure 9: LPETNet: Decoding time comparison across decoding strategies and beam widths (λ).

As empirically observed from the results, pure beam search demonstrates consistent performance across all beam widths, achieving a stable WER of 5.40% and CER of 2.10%. The BLEU score remains unchanged at 0.9460, suggesting that increasing λ provides no added benefit in terms of output quality. In contrast, char-LM decoding initially results in slightly degraded performance, with a WER of 5.96% and CER of 2.23% at $\lambda = 1$. As λ increases, there is a marginal improvement in both metrics, but the gains plateau beyond $\lambda = 4$. Despite these small gains, char-LM decoding incurs a significant time penalty—exceeding 158 seconds in all cases compared to just 130.62 seconds for greedy decoding.

ALPETNet Decoding Performance

For ALPETNet, the decoding performance metrics are detailed in Table 7, while decoding runtimes are summarised in Table 8. Figures 10 and 11 illustrate WER and CER trends across increasing λ for each strategy, and Figure 12 depicts the associated decoding time.

Table 7: ALPETNet: Decoding accuracy across strategies and beam widths (λ).

Strategy	beam widths (λ)	WER (%)	CER (%)	BLEU
Greedy (Pure Beam)	**1**	**4.16**	**1.71**	**0.9584**
Pure Beam	2	4.16	1.71	0.9584
Pure Beam	4	4.16	1.71	0.9584
Pure Beam	8	4.16	1.71	0.9584
Pure Beam	16	4.16	1.71	0.9584
Char-LM	1	4.55	1.82	0.9545
Char-LM	2	4.34	1.78	0.9565
Char-LM	4	4.33	1.77	0.9567
Char-LM	8	4.32	1.77	0.9567
Char-LM	16	4.32	1.77	0.9567

Table 8: ALPETNet: Decoding time (in seconds) across strategies and beam widths (λ).

Strategy	beam widths (λ)	Time (s)
Greedy (Pure Beam)	**1**	**125.37**
Pure Beam	2	164.00
Pure Beam	4	164.21
Pure Beam	8	163.85
Pure Beam	16	164.22
Char-LM	1	163.20
Char-LM	2	163.62
Char-LM	4	163.95
Char-LM	8	163.72
Char-LM	16	164.61

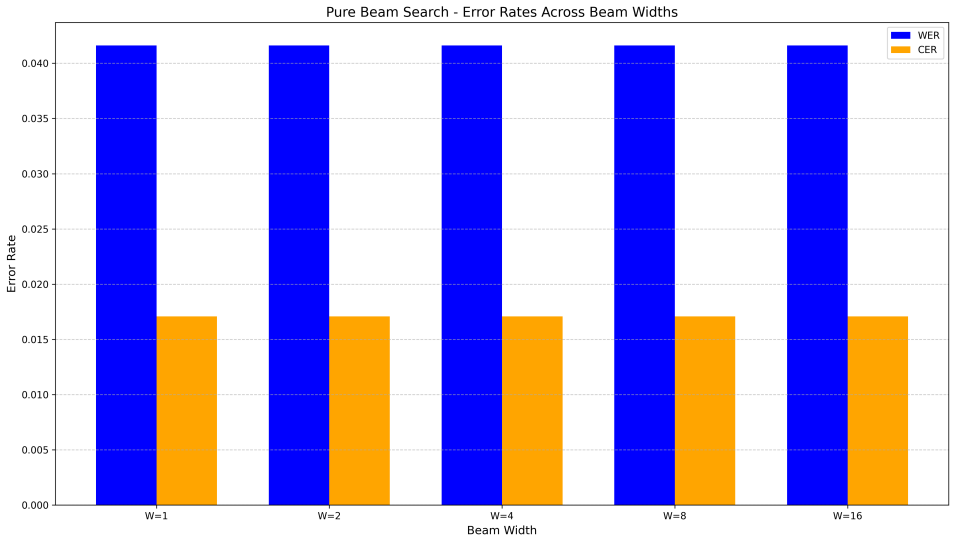

Figure 10: ALPETNet: Pure beam search error rates across beam widths (λ).

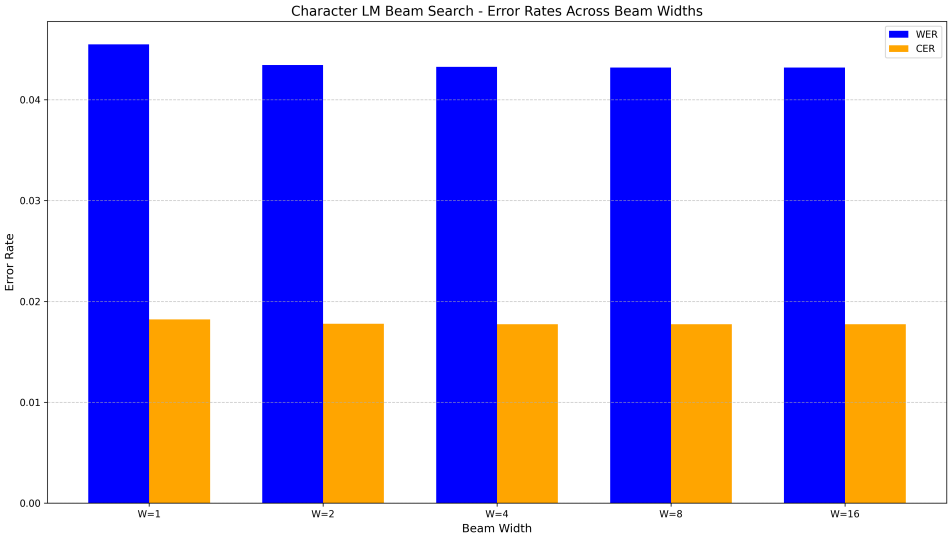

Figure 11: ALPETNet: Character LM beam search error rates across beam widths (λ).

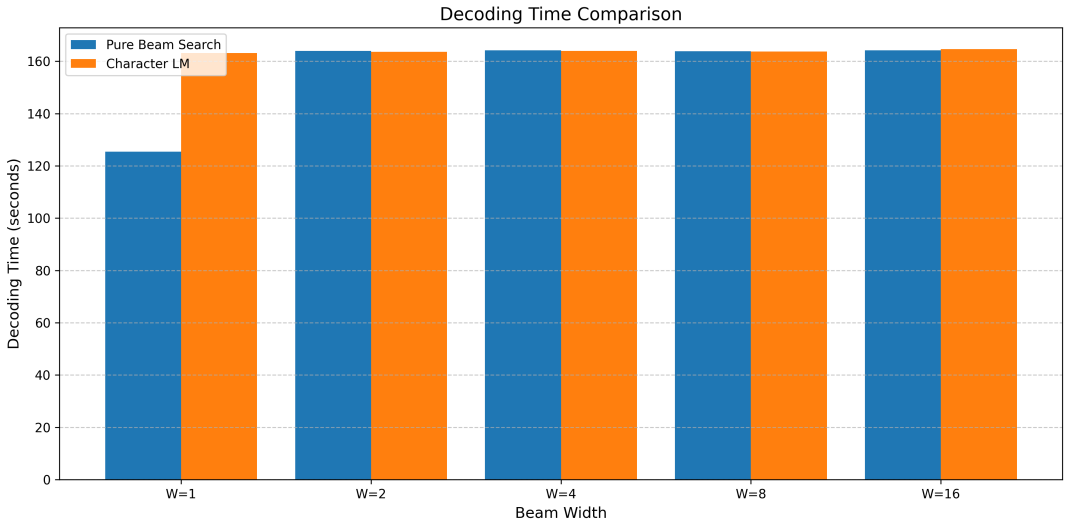

Figure 12: ALPETNet: Decoding time comparison across decoding strategies and beam widths (λ).

From these results, ALPETNet consistently outperforms LPETNet across all decoding settings. Pure beam decoding maintains the lowest error rates, with WER at 4.16% and CER at 1.71%, unaffected by beam width. BLEU scores remain high at 0.9584. In contrast, char-LM decoding performs slightly worse overall, starting with a WER of 4.55% and CER of 1.82% at $\lambda = 1$, and gradually improving to 4.32% and 1.77% respectively by $\lambda = 16$. However, the associated decoding time rises noticeably, exceeding 164 seconds at the highest beam widths.

Overall Analysis and Comparison

The results across both architectures reinforce a few key insights. Firstly, pure beam decoding consistently outperforms char-LM decoding in terms of both accuracy and runtime. This is particularly evident in LPETNet, where the char-LM adds computational overhead without delivering substantial performance gains. Even in ALPETNet, where base predictions are more accurate, the incremental improvement from char-LM decoding does not justify the additional decoding cost.

Secondly, the impact of increasing beam width appears negligible beyond $\lambda = 4$, suggesting limited benefits from wider search in both strategies. This plateauing effect implies that both LPETNet and ALPETNet generate sufficiently confident predictions that do not require large beam sizes or external char-LM to achieve accurate results.

Finally, ALPETNet clearly offers superior performance across all metrics while maintaining comparable decoding times, demonstrating the effectiveness of its attention-guided and fusion-enhanced design. The results validate that when the architecture is sufficiently expressive and temporally aware, the reliance on language models diminishes, reducing both inference time and system complexity.

4.4 Character-wise Confusion Matrix Analysis

To further assess the fine-grained predictive capability of the proposed system, a character-wise confusion matrix is generated using the outputs of the ALPETNet model. Given that ALPETNet has demonstrated superior overall performance in terms of WER and CER, this model is selected for detailed per-character analysis. As shown in Figure 13, the confusion matrix provides a visual summary of how frequently each character in the target vocabulary is correctly classified or confused with others during inference.

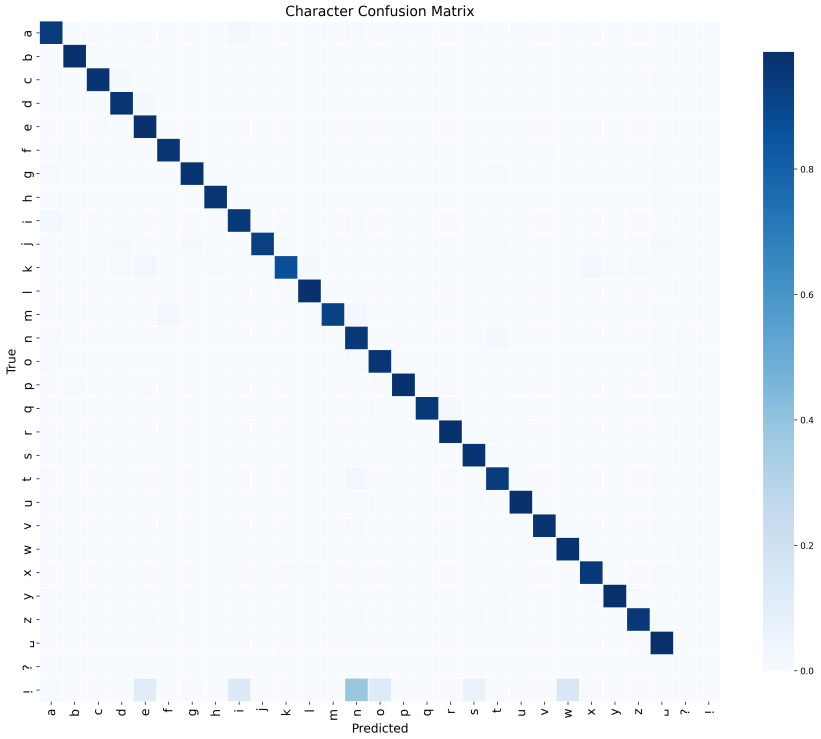

Figure 13: Character-level normalised confusion matrix for ALPETNet.

The confusion matrix demonstrates a dominant diagonal structure, indicating high character-level recognition accuracy by the ALPETNet model. Darker shades along the diagonal reflect a greater number of correct classifications, affirming the model's ability to learn discriminative spatiotemporal patterns for VSR.

Despite overall strong performance, certain character-level misclassifications remain. For example, the character `a` is occasionally misclassified as `i`, a trend that recurs in over-predicted sequences. This is attributable to their similar vertical mouth opening and brief articulation, particularly when viewed without accompanying audio. Likewise, the character `m` is often confused with both `n` and `f`, due to overlapping visual articulations. The confusion with `n` is expected, given their shared nasal production and similar lip closure during articulation. Misclassification of `m` as `f`, by contrast, arises from partial visual resemblance—especially under occlusion or when subtle lip motions are not distinctly captured. These errors highlight the inherent difficulty of differentiating bilabial nasals from labiodental fricatives in visual-only recognition tasks.

Another prominent pattern is observed in overpredictions, captured in the final row of the matrix via the `!` token. These hallucinated characters, which appear in predictions without a corresponding ground-truth match, predominantly involve `n`, `o`, and `w`. Their recurrence suggests these characters are visually salient or temporally ambiguous, leading the decoder to overproduce them in uncertain contexts.

Additionally, sparse activations involving special symbols like `?` and `␣` are observed. The `?` symbol is not a model output but is introduced post hoc to account for predictions that fall outside the valid character set or cannot be matched temporally with the ground truth. The space character `␣`, which marks inter-word boundaries, also exhibits minor

confusion, especially near transitions between word segments.

Taken together, these findings confirm that while ALPETNet performs robustly at the character level, certain systematic errors arise from ambiguous articulatory motions or sequence misalignments. These are further examined in Section 4.5, particularly through the lens of phonemic and visemic overlaps.

4.5 Phoneme-wise Confusion Matrix Analysis

To complement the character-level evaluation, a phoneme-level confusion analysis is conducted using ARPAbet representations. Of the 39 phonemes defined in the ARPAbet set, 31 are present in the GRID corpus, allowing for a comprehensive assessment of phoneme-specific recognition performance within the dataset's vocabulary constraints. This analysis provides fine-grained insights into which phonetic distinctions are captured well by the model and which remain visually ambiguous under silent conditions. The confusion matrix is computed by aligning predicted and ground-truth phoneme sequences and is shown in Figure 14, where darker shades along the diagonal represent higher phoneme classification accuracy.

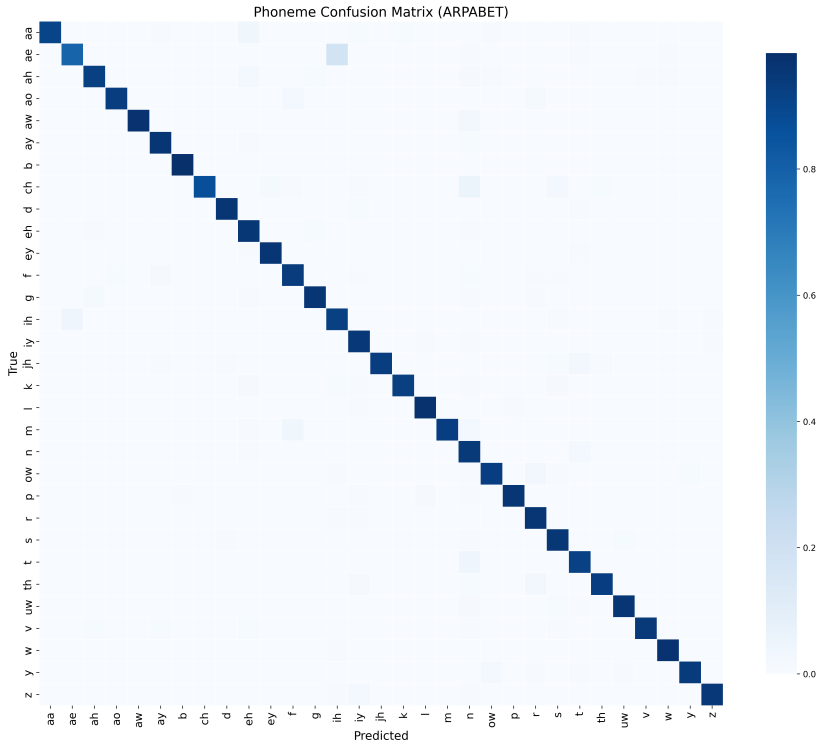

Figure 14: Phoneme confusion matrix for ALPETNet using ARPAbet transcriptions.

To facilitate interpretability, phoneme-to-viseme clustering is adopted following the taxonomy proposed by Neti et al. (2000), presented in Table 9. This classification groups phonemes into perceptually similar categories (visemes) based on shared articulatory features, enabling meaningful interpretation of phoneme-level errors in VSR.

Table 9: Phoneme-to-viseme clustering based on Neti et al. (2000).

Code	Viseme Class	Phonemes in Cluster
V1	Lip-rounded monophthongs	/ao/ /ah/ /aa/ /er/ /oy/ /aw/ /hh/
V2	Rounded back vowels	/uw/ /uh/ /ow/
V3	Front vowels and diphthongs	/ae/ /eh/ /ey/ /ay/
V4	High front vowels	/ih/ /iy/ /ax/
A	Alveolar-semivowels	/l/ /el/ /r/ /y/
B	Alveolar-fricatives	/s/ /z/
C	Alveolar stops and nasals	/t/ /d/ /n/ /en/
D	Palato-alveolar affricates	/sh/ /zh/ /ch/ /jh/
E	Bilabials	/p/ /b/ /m/
F	Dentals	/th/ /dh/
G	Labio-dentals	/f/ /v/
H	Velars and glides	/ng/ /k/ /g/ /w/
S	Silence	/sil/ /sp/

Vowel and Diphthong Confusions. Significant confusion is observed between the front vowel /ae/ and the high front vowel /ih/, both belonging to visually overlapping viseme classes (V3 and V4). Notably, the back vowel /aa/ exhibits frequent misclassification as the mid-front vowel /eh/, suggesting that the visual model associates similar open-jaw articulations between these phonemes. These errors reinforce the challenge of separating monophthongs and diphthongs based purely on lip and jaw movements, particularly when mid-to-low vowel transitions are visually ambiguous.

Alveolar and Nasal Interference. The alveolar nasal /n/ is repeatedly confused with the alveolar stop /t/. Although distinct acoustically, they share a place of articulation, which limits visual separability. Lesser confusion also appears between /d/ and /n/, further supporting this visual ambiguity.

Cross-Class Misclassifications. A clear confusion arises between /ch/ (an affricate in D) and /n/ (a nasal in C), despite their different articulatory classes. This pattern indicates alignment mismatches or visual similarity in lip closure during rapid articulation. Additional off-diagonal entries reveal confusion between /r/ (A) and /th/ (F), both of which involve tongue-front gestures that are visually similar when viewed from the front.

Labiodental–Nasal Overlap. Confusion is evident between /m/ (E) and /f/ (G). Although these phonemes involve different articulation types, the lip shapes during production can appear similar under occlusion or poor lighting conditions.

Bilabial Clarity. The model demonstrates generally strong separability among bilabial phonemes /p/, /b/, and /m/, with dominant diagonal entries indicating accurate classification. While there is a minor confusion between /p/ and /b/ at a rate of approximately 1%, this remains relatively low and does not significantly detract from the model's overall ability to distinguish bilabial articulations. These results suggest that ALPETNet effectively captures visually salient cues such as full lip closure and release

dynamics, even in the presence of subtle intra-class ambiguity.

Interpretation and Implications. These findings are consistent with prior literature, which reports the inherent difficulty of resolving phonemes within the same viseme class. Visemic clusters such as V3 and V4 (front and high vowels) present the greatest challenge. Introducing viseme-aware regularisation or explicitly encoding viseme boundaries within the loss function would increase robustness against such intra-class confusions.

GRID Corpus Lexicon. The complete vocabulary of the GRID corpus can be formally expressed as a union of distinct lexical categories, where:

$$\mathcal{V}_{\text{GRID}} = \mathcal{C}_{\text{cmd}} \cup \mathcal{C}_{\text{col}} \cup \mathcal{C}_{\text{prep}} \cup \mathcal{C}_{\text{let}} \cup \mathcal{C}_{\text{dig}} \cup \mathcal{C}_{\text{adv}}$$

with each component defined as:

$$\begin{aligned}
\mathcal{C}_{\text{cmd}} &= \{\texttt{SET}, \texttt{LAY}, \texttt{PLACE}\} \\
\mathcal{C}_{\text{col}} &= \{\texttt{BLUE}, \texttt{GREEN}, \texttt{RED}, \texttt{WHITE}\} \\
\mathcal{C}_{\text{prep}} &= \{\texttt{AT}, \texttt{BY}, \texttt{IN}, \texttt{WITH}\} \\
\mathcal{C}_{\text{let}} &= \{\texttt{A}, \texttt{B}, \texttt{C}, \texttt{D}, \texttt{E}, \texttt{F}, \texttt{G}, \texttt{H}, \texttt{I}, \texttt{J}, \texttt{K}, \texttt{L}, \texttt{M}, \texttt{N}, \texttt{O}, \texttt{P}, \texttt{Q}, \texttt{R}, \texttt{S}, \texttt{T}, \texttt{U}, \texttt{V}, \texttt{X}, \texttt{Y}, \texttt{Z}\} \\
\mathcal{C}_{\text{dig}} &= \{\texttt{ZERO}, \texttt{ONE}, \texttt{TWO}, \texttt{THREE}, \texttt{FOUR}, \texttt{FIVE}, \texttt{SIX}, \texttt{SEVEN}, \texttt{EIGHT}, \texttt{NINE}\} \\
\mathcal{C}_{\text{adv}} &= \{\texttt{AGAIN}, \texttt{NOW}, \texttt{PLEASE}, \texttt{SOON}\}
\end{aligned}$$

Each utterance in the GRID corpus is a syntactically structured sentence of the form:

$$\texttt{[Command]} + \texttt{[Colour]} + \texttt{[Preposition]} + \texttt{[Letter]} + \texttt{[Digit]} + \texttt{[Adverb]}$$

This compositional structure enables systematic analysis of recognition performance at both the character and phoneme levels, as conducted in Sections 4.4 and 4.5. To further enhance interpretability, Section 4.6 presents a saliency map analysis that visualises the spatial and temporal input regions most influential to the model's predictions.

4.6 Saliency Visualisation

To better understand the spatial regions that contribute to ALPETNet's predictions at the frame level, saliency visualisation is employed using guided backpropagation, introduced by Springenberg et al. (2014). This method computes saliency maps for each timestep in the decoded sequence by backpropagating gradients from the model's output to the input video frames, while suppressing negative gradients to enhance visual clarity. The resulting maps highlight spatial regions the model relies on most when predicting individual characters, offering insight into both temporal dynamics and spatial focus across the lip region.

The evolution of attention across input frames for two representative words, `BLUE` and `PLEASE`, is illustrated through saliency visualisations that reveal the specific spatial regions the network attends to at each timestep when predicting individual characters, as shown in Figure 15 and Figure 16.

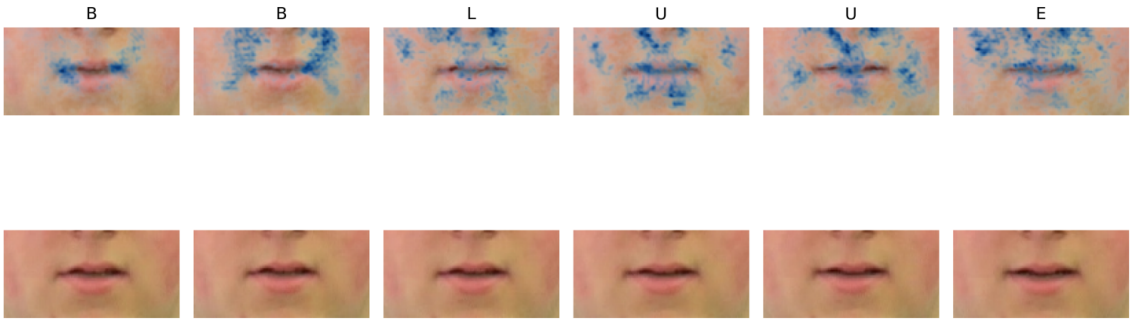

Figure 15: Saliency visualisation for the word BLUE. Top: saliency maps overlaid on input frames. Bottom: corresponding raw frames.

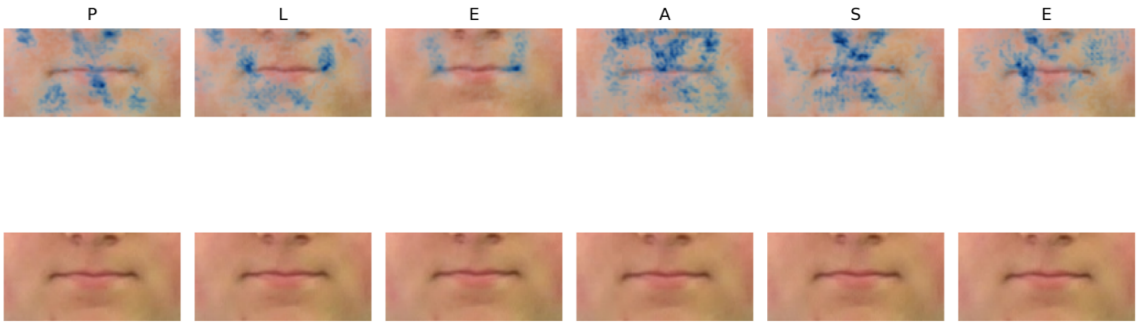

Figure 16: Saliency visualisation for the word PLEASE. Top: saliency maps overlaid on input frames. Bottom: corresponding raw frames.

In the case of the first word, the saliency maps show strong, symmetric activations around the lower and mid-lip contours, particularly during the bilabial closure for the character B. This corresponds to the visual signature of full lip closure and burst release necessary for bilabial articulation. For the character L, attention is slightly elevated along the lateral lip boundaries, reflecting the tongue's contact with the alveolar ridge and the resulting subtle cheek and lip movement during articulation of this lateral consonant. During the prediction of the character U, the network attends more broadly to the overall lip contour, capturing the rounded lip posture characteristic of back rounded vowels. Finally, the saliency associated with the character E, a high-front vowel, becomes more diffuse and lighter in intensity, suggesting the model relies on subtler cues such as lip spreading and vertical lip tension—features that are visually less pronounced yet essential for discriminating front vowels.

The saliency patterns for the second word reveal more complex articulatory transitions. At the onset, the bilabial plosive P triggers strong and symmetric activations around the lips and philtrum, indicating the network's reliance on visual cues related to lip compression and burst release—key characteristics for detecting bilabial closures. During production of the lateral consonant L, the model's attention shifts laterally, focusing on the cheeks and edges of the mouth, where lateral tongue contact subtly influences

facial motion. For the high-front vowel E, the saliency becomes more vertically extended and diffuse, with the network attending to lip spreading and the upper cheek region. This reflects the visual pattern of increased oral aperture and lip tension typical of front unrounded vowels. The open front vowel A elicits heightened saliency around the philtrum and central upper lip, suggesting the model is responding to the raised lip posture and vertical tension common during its articulation. The fricative S produces sharp, localised saliency near the lip's centre and corners, capturing the constriction associated with airflow modulation. Finally, the terminal E shows a renewed spread of attention along the mid-to-upper lip contours, closely resembling the first E activation but with a slight temporal lag, indicating the model's sensitivity to repeated vowels and phrase-final elongation.

These visualisations confirm that the model does not rely on static templates but rather adapts its attention dynamically according to character-specific articulatory patterns. The clear activation boundaries also suggest that ALPETNet possesses fine spatial resolution in its visual encoder. Such behaviour is in line with prior VSR studies such as LipNet by Assael et al. (2016), where similar spatiotemporal attention patterns are observed.

This spatial interpretability supports the phoneme and character-level findings discussed in Sections 4.4 and 4.5, further reinforcing the model's capability to track visual articulation at fine granularity.

4.7 Chapter Summary

This chapter has presented a detailed evaluation of the proposed architectures through quantitative metrics, decoding analysis, and interpretability studies. ALPETNet has demonstrated superior performance to LPETNet in terms of WER, CER, and BLEU, while maintaining a lightweight and modular design. Comparative benchmarks have confirmed its competitiveness against state-of-the-art lip-reading models. Further insights from character- and phoneme-level confusion matrices have highlighted persisting visemic ambiguities, while saliency visualisation has validated the model's capacity to capture spatially and temporally discriminative features.

These findings affirm the effectiveness of the architectural enhancements introduced in ALPETNet—particularly gated attention, multi-scale patch embedding, and cross-attention fusion—in achieving robust and interpretable VSR. At the same time, they expose challenges such as diminishing returns from external language models and difficulties in disambiguating visually similar phonemes.

The final chapter, Chapter 5 consolidates these insights by outlining the core contributions of this dissertation, proposing future directions—including multimodal extensions and viseme-aware training—and offering reflections on the technical, ethical, and personal challenges encountered throughout the dissertation.

5 Contribution and Future Works

This dissertation presents a comprehensive study into the development and evaluation of a lightweight yet expressive architecture for VSR, termed ALPETNet. The proposed model combines efficient spatiotemporal encoding with phoneme- and character-level decoding strategies, including a detailed analysis of decoding methodologies and saliency-based interpretability.

One of the principal contributions lies in the design of ALPETNet itself, which integrates convolutional and attention mechanisms to capture fine-grained lip dynamics while maintaining computational efficiency. The model is rigorously compared against the first proposed architecture, LPETNet using both character and phoneme-level evaluation metrics. The performance gains demonstrated by ALPETNet, particularly in error reduction and decoding robustness, underline its effectiveness in low-resource and real-time scenarios.

Additionally, this dissertation implements and benchmarks several CTC decoding strategies, namely greedy decoding, pure beam search, and character-based language modelling using the KenLM toolkit. An extensive comparison of these strategies is conducted across multiple beam widths, highlighting the trade-offs between decoding accuracy and computational time. The incorporation of a character-level language model and its nuanced impact on transcription performance are quantitatively explored.

Beyond model evaluation, this dissertation advances interpretability in lip-reading systems through visual saliency analysis. Character-level saliency maps are extracted to localise discriminative regions associated with each predicted letter, and these are further analysed across phonemes and visemes. The insights derived from these visualisations provide empirical evidence for the model's attention to articulatory cues, particularly in the presence of bilabial closures, vowel spreading, and fricative constriction. Moreover, confusion matrices at both the character and phoneme level offer clarity into the types of errors still prevalent in visual-only systems and motivate future improvements in viseme-aware training.

Finally, the GRID corpus is carefully deconstructed into its constituent lexical categories to support systematic recognition, and phoneme-to-viseme mappings are adopted from established linguistic taxonomies. This allows for structured evaluation and alignment with prior literature in audiovisual speech processing.

All aims and objectives outlined in the Section 1.2 and 1.3 have been successfully met. While the proposed model does not surpass all existing baselines in raw accuracy, it achieves performance that is competitive with state-of-the-art systems such as LipNet by Assael et al. (2016) and Resformer by Xue et al. (2023), while significantly reducing architectural complexity. This balance between recognition performance and computational efficiency makes the model particularly suitable for deployment in resource-constrained or security-critical settings, where interpretability and low latency are essential. As such, the dissertation fulfils the overarching goal of developing a lightweight yet linguistically robust VSR system.

5.1 Future Work

While this dissertation successfully demonstrates the efficacy of a lightweight transformer-based architecture for VSR, several avenues remain open for further exploration.

First, a promising extension would involve integrating audio features alongside visual input to construct a multimodal or ensemble model. This approach could enhance temporal alignment and phoneme disambiguation, particularly for visually indistinct phonemes. Combining auditory cues with visual information has been shown in prior works to significantly improve speech recognition accuracy in noisy or occluded settings.

Second, expanding the evaluation to larger and more diverse datasets could improve generalisability. As discussed in Section 2.6, corpora such as LRS by Son Chung et al. (2017), LRS2-BBC by Afouras et al. (2018a), and LRS3-TED by Afouras et al. (2018b) offer increased vocabulary size and variability in speaker identity, accent, and setting. Similarly, MV-LRS by Son and Zisserman (2017) introduces robustness to head pose variation through multi-view recordings. These datasets are not utilised in this dissertation due to VRAM and hardware constraints, but they provide an excellent foundation for future work in domain adaptation and scalability.

Third, the deployment of more sophisticated backbone architectures such as CvT and Swin Transformers could further enhance spatial and hierarchical representation learning. These models have demonstrated superior performance in various vision tasks but are not included in this dissertation due to GPU memory limitations.

In addition, investigating viseme-aware training mechanisms or viseme-augmented loss functions can help mitigate phoneme-level confusions as discussed in Section 4.5. Explicit modelling of viseme transitions could encourage the model to disambiguate phonemes with similar visual cues more reliably.

Finally, exploring real-time deployment in constrained environments, such as embedded systems and security surveillance settings, represents a promising direction for future work. Reducing inference latency while maintaining high recognition accuracy under limited computational budgets would bring the model closer to practical application. Such optimisations are particularly pertinent for security-critical deployments, including silent speech interfaces, surveillance systems operating in audio-restricted zones, and emergency communication technologies where robust VSR can offer critical advantages over traditional audio-based solutions.

Collectively, these directions aim to build upon the current system's strengths while addressing the limitations posed by dataset scale, architectural complexity, and multimodal integration.

5.2 Reflection

The development of this dissertation has involved navigating a range of technical and practical challenges, each of which contributed to a deeper understanding of applied machine learning in the domain of VSR. One notable hurdle is the need to dual boot into a Unix-based operating system (Linux) to facilitate the use of the KenLM library for decoding. Since KenLM is not natively supported on Windows, this has added a layer of complexity to the experimental pipeline, particularly for conducting the comparative evaluation between decoding strategies involving character-based language modelling.

Another key challenge is in the early stages of model implementation. The initial architectural choices lean towards highly complex transformer-based networks inspired by state-of-the-art literature. These designs, while theoretically appealing, has proved incompatible with the memory limitations of the available graphics card (RTX 4060 Ti, 16 GB). This mismatch has led to repeated out-of-memory errors and protracted debugging cycles, resulting in weeks of stagnated progress. It is only after extensive exploration of relevant papers and re-evaluation of architectural priorities that a more feasible modular approach is adopted. Thereafter, the models are trained sequentially from scratch, each converging within a matter of hours — as originally expected.

Beyond the technical hurdles, the dissertation has also demanded considerable effort in maintaining a disciplined experimental protocol. From constructing reproducible training pipelines to performing consistent metric evaluation across decoding strategies and saliency visualisation, each stage has reinforced the importance of scientific rigour and documentation. This iterative refinement process has not only enhanced the reliability of the results but has also improved the clarity of model interpretability — particularly through the character- and phoneme-level confusion analyses and saliency map examinations.

Throughout the dissertation, relevant legal, ethical, and societal aspects under LSEPI principles are considered. The source code and documentation constitute original intellectual property, are developed independently and in line with University policies. While there are no commercial plans, open-source release can be explored for academic benefit. As the dissertation does not involve personal data or non-public human input, data protection and ethics review are not required. Broader impacts, including accessibility and the risk of misuse, are addressed through transparent model design and strong documentation, ensuring alignment with responsible innovation standards.

In retrospect, the experience has highlighted the importance of adaptability, ethical awareness, and targeted troubleshooting. Balancing technical goals with societal considerations is key to success. Overall, the dissertation has strengthened my technical and research skills, laying a strong foundation for future work in multimodal, linguistically-aware machine learning.

References

Afouras, T., Chung, J. S., Senior, A., Vinyals, O., and Zisserman, A. (2018a). Deep audio-visual speech recognition. *IEEE transactions on pattern analysis and machine intelligence*, 44(12):8717–8727.

Afouras, T., Chung, J. S., and Zisserman, A. (2018b). Lrs3-ted: a large-scale dataset for visual speech recognition. *arXiv preprint arXiv:1809.00496*.

Ahn, K., Cheng, X., Song, M., Yun, C., Jadbabaie, A., and Sra, S. (2023). Linear attention is (maybe) all you need (to understand transformer optimization). *arXiv preprint arXiv:2310.01082*.

Anina, I., Zhou, Z., Zhao, G., and Pietikäinen, M. (2015). Ouluvs2: A multi-view audio-visual database for non-rigid mouth motion analysis. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, volume 1, pages 1–5. IEEE.

Arakane, T. and Saitoh, T. (2023). Efficient dnn model for word lip-reading. *Algorithms*, 16(6):269.

Assael, Y. M., Shillingford, B., Whiteson, S., and De Freitas, N. (2016). Lipnet: End-to-end sentence-level lipreading. *arXiv preprint arXiv:1611.01599*.

Bear, H. L. and Harvey, R. (2017). Phoneme-to-viseme mappings: the good, the bad, and the ugly. *Speech Communication*, 95:40–67.

Bird, S. (2006). Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL 2006 interactive presentation sessions*, pages 69–72.

Bulzomi, H., Schweiker, M., Gruel, A., and Martinet, J. (2023). End-to-end neuromorphic lip-reading. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4101–4108.

Chen, C.-F. R., Fan, Q., and Panda, R. (2021). Crossvit: Cross-attention multi-scale vision transformer for image classification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 357–366.

Cho, K., Van Merriënboer, B., Bahdanau, D., and Bengio, Y. (2014). On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.

Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

Chung, J. S. and Zisserman, A. (2017). Lip reading in the wild. In *Computer Vision–ACCV 2016: 13th Asian Conference on Computer Vision, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part II 13*, pages 87–103. Springer.

Cooke, M., Barker, J., Cunningham, S., and Shao, X. (2006a). An audio-visual corpus for speech perception and automatic speech recognition. *The Journal of the Acoustical Society of America*, 120(5):2421–2424.

Cooke, M., Barker, J., Cunningham, S., and Shao, X. (2006b). The grid audio-visual speech corpus.

Devi, T. M., Keerthana, S., Santhi, P., Pravallika, P., and Rajeshwari, S. (2023). Silent speech recognition: Automatic lip reading model using 3d cnn and gru. In *International Conference on Data Science, Machine Learning and Applications*, pages 827–832. Springer.

Fernandez-Lopez, A. and Sukno, F. M. (2018). Survey on automatic lip-reading in the era of deep learning. *Image and Vision Computing*, 78:53–72.

Fisher, C. G. (1968). Confusions among visually perceived consonants. *Journal of speech and hearing research*, 11(4):796–804.

Fu, Y., Lu, Y., and Ni, R. (2023). Chinese lip-reading research based on shufflenet and cbam. *Applied Sciences*, 13(2):1106.

Fung, I. and Mak, B. (2018). End-to-end low-resource lip-reading with maxout cnn and lstm. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2511–2515. IEEE.

Garg, A., Noyola, J., and Bagadia, S. (2016). Lip reading using cnn and lstm. *Technical report, Stanford University, CS231 n project report*.

Gergen, S., Zeiler, S., Abdelaziz, A. H., Nickel, R. M., and Kolossa, D. (2016). Dynamic stream weighting for turbo-decoding-based audiovisual asr. In *INTERSPEECH*, pages 2135–2139.

Goldschen, A. J., Garcia, O. N., and Petajan, E. D. (1997). Continuous automatic speech recognition by lipreading. In *Motion-Based recognition*, pages 321–343. Springer.

Graves, A. (2012). Connectionist temporal classification. In *Supervised sequence labelling with recurrent neural networks*, pages 61–93. Springer.

Guo, Y., Li, Y., Wang, L., and Rosing, T. (2019). Depthwise convolution is all you need for learning multiple visual domains. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8368–8375.

Gutierrez, A. and Robert, Z. (2017). Lip reading word classification. *Comput Vision-ACCV*, pages 1–9.

Hong, J., Kim, M., Choi, J., and Ro, Y. M. (2023). Watch or listen: Robust audio-visual speech recognition with visual corruption modeling and reliability scoring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18783–18794.

Hu, J., Shen, L., and Sun, G. (2018). Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Jeon, S., Elsharkawy, A., and Kim, M. S. (2021). Lipreading architecture based on multiple convolutional neural networks for sentence-level visual speech recognition. *Sensors*, 22(1):72.

Jitaru, A. C., Abdulamit, Ş., and Ionescu, B. (2020). Lrro: a lip reading data set for the under-resourced romanian language. In *Proceedings of the 11th ACM Multimedia Systems Conference*, pages 267–272.

Kastaniotis, D., Tsourounis, D., Koureleas, A., Peev, B., Theoharatos, C., and Fotopoulos, S. (2019). Lip reading in greek words at unconstrained driving scenario. In *2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA)*, pages 1–6. IEEE.

Katsamanis, A., Papandreou, G., and Maragos, P. (2009). Face active appearance modeling and speech acoustic information to recover articulation. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(3):411–422.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kumar, Y., Aggarwal, M., Nawal, P., Satoh, S., Shah, R. R., and Zimmermann, R. (2018). Harnessing ai for speech reconstruction using multi-view silent video feed. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 1976–1983.

Kumar, Y., Sahrawat, D., Maheshwari, S., Mahata, D., Stent, A., Yin, Y., Shah, R. R., and Zimmermann, R. (2020). Harnessing gans for zero-shot learning of new classes in visual speech recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 2645–2652.

Lee, M. (2023). Mathematical analysis and performance evaluation of the gelu activation function in deep learning. *Journal of Mathematics*, 2023(1):4229924.

Liu, W., Zhu, F., Ma, S., and Liu, C.-L. (2024). Mspe: Multi-scale patch embedding prompts vision transformers to any resolution. *arXiv preprint arXiv:2405.18240*.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. (2021). Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022.

Luettin, J., Thacker, N. A., and Beet, S. W. (1996). Visual speech recognition using active shape models and hidden markov models. In *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, volume 2, pages 817–820. IEEE.

Makhlouf, A., Lazli, L., and Bensaker, B. (2013). Hybrid hidden markov models and genetic algorithm for robust automatic visual speech recognition. *Journal of Information Technology Review (JITR)*, 4(3):105–114.

Matthews, I., Cootes, T. F., Bangham, J. A., Cox, S., and Harvey, R. (2002). Extraction of visual features for lipreading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2):198–213.

McGurk, H. and MacDonald, J. (1976). Hearing lips and seeing voices. *Nature*, 264(5588):746–748.

NadeemHashmi, S., Gupta, H., Mittal, D., Kumar, K., Nanda, A., and Gupta, S. (2018). A lip reading model using cnn with batch normalization. In *2018 eleventh international conference on contemporary computing (IC3)*, pages 1–6. IEEE.

Neti, C., Potamianos, G., Luettin, J., Matthews, I., Glotin, H., Vergyri, D., Sison, J., and Mashari, A. (2000). Audio visual speech recognition.

Oghbaie, M., Sabaghi, A., Hashemifard, K., and Akbari, M. (2025). When deep learning deciphers silent video: a survey on automatic deep lip reading. *Multimedia Tools and Applications*, pages 1–43.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Park, Y.-H., Park, R.-H., and Park, H.-M. (2024). Swinlip: An efficient visual speech encoder for lip reading using swin transformer.

Patterson, E. K., Gurbuz, S., Tufekci, Z., and Gowdy, J. N. (2002). Cuave: A new audio-visual database for multimodal human-computer interface research. In *2002 IEEE International conference on acoustics, speech, and signal processing*, volume 2, pages II–2017. IEEE.

Phan, L., Nguyen, H. T. H., Warrier, H., and Gupta, Y. (2022). Patch embedding as local features: Unifying deep local and global features via vision transformer for image retrieval. In *Proceedings of the Asian Conference on Computer Vision*, pages 2527–2544.

Potamianos, G., Neti, C., Gravier, G., Garg, A., and Senior, A. W. (2003). Recent advances in the automatic recognition of audiovisual speech. *Proceedings of the IEEE*, 91(9):1306–1326.

Rekik, A., Ben-Hamadou, A., and Mahdi, W. (2014). A new visual speech recognition approach for rgb-d cameras. In *Image Analysis and Recognition: 11th International Conference, ICIAR 2014, Vilamoura, Portugal, October 22-24, 2014, Proceedings, Part II 11*, pages 21–28. Springer.

Sarhan, A. M., Elshennawy, N. M., and Ibrahim, D. M. (2021). Hlr-net: a hybrid lip-reading model based on deep convolutional neural networks. *Computers, Materials and Continua*, 68(2):1531–49.

Shirakata, T. and Saitoh, T. (2020). Lip reading using facial expression features. *Int. J. Comput. Vis. Signal Process*, 1(1):9–15.

Son, J. S. and Zisserman, A. (2017). Lip reading in profile. In *Proceedings of the British Machine Vision Conference*, volume 2017.

Son Chung, J., Senior, A., Vinyals, O., and Zisserman, A. (2017). Lip reading sentences in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6447–6456.

Springenberg, J. T., Dosovitskiy, A., Brox, T., and Riedmiller, M. (2014). Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, H., Guo, P., Zhou, P., and Xie, L. (2024a). Mlca-avsr: Multi-layer cross attention fusion based audio-visual speech recognition. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8150–8154. IEEE.

Wang, H., Pu, G., and Chen, T. (2022). A lip reading method based on 3d convolutional vision transformer. *IEEE Access*, 10:77205–77212.

Wang, J., Pan, Z., Zhang, M., Tan, R. T., and Li, H. (2024b). Restoring speaking lips from occlusion for audio-visual speech recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19144–19152.

Wu, H., Xiao, B., Codella, N., Liu, M., Dai, X., Yuan, L., and Zhang, L. (2021). Cvt: Introducing convolutions to vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 22–31.

Wu, Q., Li, M., Shen, J., Lü, L., Du, B., and Zhang, K. (2023). Transformerlight: A novel sequence modeling based traffic signaling mechanism via gated transformer. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, pages 2639–2647.

Xiao, J. (2018). 3d feature pyramid attention module for robust visual speech recognition. *arXiv preprint arXiv:1810.06178*.

Xu, K., Li, D., Cassimatis, N., and Wang, X. (2018). Lcanet: End-to-end lipreading with cascaded attention-ctc. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 548–555. IEEE.

Xue, J., Huang, S., Song, H., and Shi, L. (2023). Fine-grained sequence-to-sequence lip reading based on self-attention and self-distillation. *Frontiers of Computer Science*, 17(6):176344.

Yang, S., Zhang, Y., Feng, D., Yang, M., Wang, C., Xiao, J., Long, K., Shan, S., and Chen, X. (2019). Lrw-1000: A naturally-distributed large-scale benchmark for lip reading in the wild. In *2019 14th IEEE international conference on automatic face & gesture recognition (FG 2019)*, pages 1–8. IEEE.

Yin, J., Wu, J., Gao, C., Yu, H., Liu, L., and Guo, S. (2024). A novel fish individual recognition method for precision farming based on knowledge distillation strategy and the range of the receptive field. *Journal of Fish Biology*, 105(3):721–734.

Zhou, Z., Zhao, G., Hong, X., and Pietikäinen, M. (2014). A review of recent advances in visual speech decoding. *Image and Vision Computing*, 32(9):590–605.