Blog Post 2: Blake Buckner

In my first post, I explored the world of lip reading, outlining its importance, the different levels at which it can be applied, and the potential for CNNs to provide accurate predictions. I also shared my initial thoughts on datasets and model architecture, emphasizing the ambitious but promising nature of this project. Since then, I have refined my approach by evaluating available datasets more critically and honing in on a practical method tailored to the resources accessible to me. In this post, I'll walk through the concrete methods I am using to build and train my lip reading model.

The most important step I've taken towards completing this project is testing and settling on a dataset. My primary hope was to get access to the Lip Reading in the Wild (LRW) dataset, and although I made several attempts to gain access, I unfortunately did not receive a response (LRW 2016). For the AVSpeech dataset but decided against using it because it requires intensive processing of YouTube videos and features multiple languages, which fall outside the scope of my current focus (Ephrat et al., 2018). This process led me to settle on the GRID dataset, which I believe is the most suitable for my needs (Cooke et al., 2006).

As with any dataset, there are both advantages and limitations to this approach. The GRID dataset contains over 30,000 short video clips, each accompanied by alignment ground truth labels. These videos feature 34 speakers, with about 1,000 clips per person, and the entire dataset totals around 14GB. I have already downloaded and tested it locally, confirming that it functions well for my purposes.

Each video is in front of a uniform blue background, with participants positioned roughly in the same spot within the frame. This consistency reduces distractions from background details and allows the model to focus primarily on the speaker's mouthes. While I initially aimed to develop a system capable of handling random, real-world videos, I knew such an ambitious goal would be challenging given my current resources. Therefore, narrowing the scope to this controlled dataset is a more practical and feasible starting point.

Although the words spoken in the videos are somewhat unusual, they are carefully constructed to cover a wide range of sounds, which should provide a solid foundation for training an effective lip reading model.

command	color*	preposition	letter*	digit*	adverb
bin	blue	at	A-Z	1-9	again
lay	green	by	Excluding W	Zero	now
place	red	white			please
set	white	with			soon

Table 1: Sentence structure (Cookie et al. 2006)

Table 1 above showcases the main design of the utterances within the GRID dataset. The starred items represent a selection of words mixed in every permutation. These were carefully chosen to maximize phonetic diversity while minimizing linguistic complexity. The letter ‘W’ is excluded since it is multisyllabic, and ‘Zero’ is specified because of dialectal pronunciation differences. As a result, I am shifting my approach away from focusing solely on individual words to instead concentrate on more sound-based analyses. This adjustment requires me to find a balance between evaluating whole words and analyzing individual sounds or phonemes.

In exploring this, further, I came across the concept of visemes. Visemes are similar to phonemes but are defined by the shape of the mouth during articulation (Urban 2025). Because multiple sounds often correspond to a single viseme due to identical mouth shapes, thus distinguishing between these sounds visually can be challenging. This means multiple different sounds could have one viseme because they look the same and are thus what is specifically hard to judge. Incorporating this knowledge could help address the inherent ambiguity in lip reading, especially by identifying which sounds overlap visually. This insight makes it easier to narrow down the possible interpretations of visually similar sounds and thereby improve prediction accuracy. Accounting for visemes is something I hope to integrate into my model to better handle these visual ambiguities, but that may remain a stretch goal.

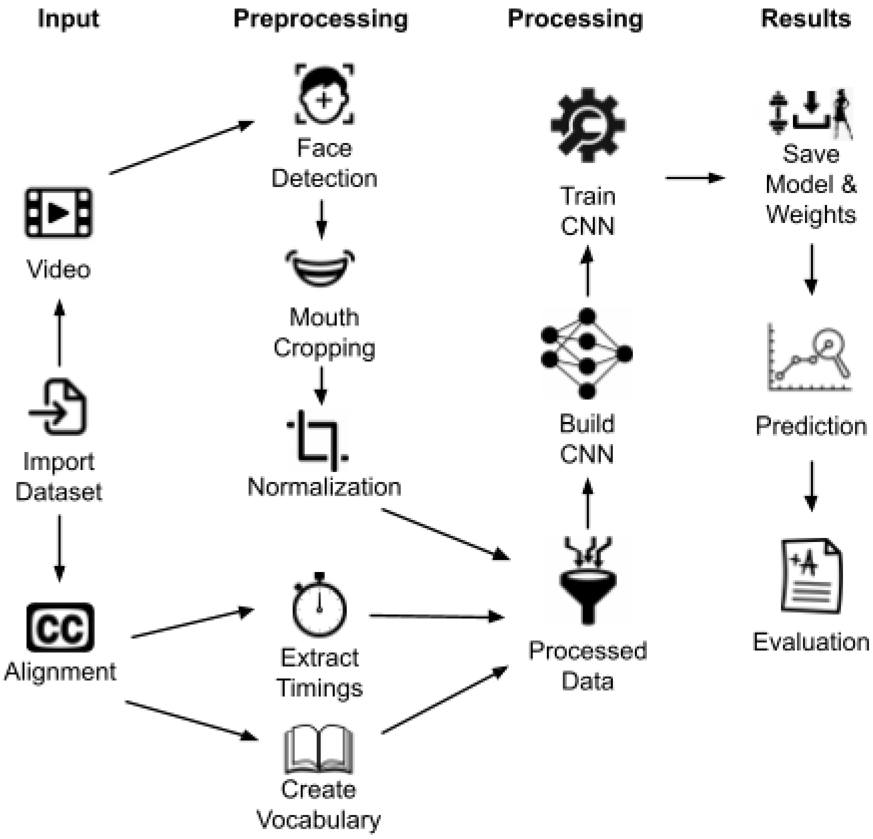

Figure 1: Full Project Pipeline

My code pipeline remains similar to my initial work in progress pipeline, but I now have a clearer understanding of the specifics involved. This pipeline is organized into four key stages: **Input**, **Preprocessing**, **Processing**, and **Results**. In the input stage, raw video and alignment data is collected, either downloaded directly or imported from Kaggle. The preprocessing stage follows, where the video undergoes critical transformations to isolate the face, crop the mouth, and other normalization that standardizes the data. Simultaneously, timing information is extracted and a vocabulary is created based on the ground truth labels provided in the alignment files. These preprocessed components feed into the processing stage, where a Convolutional Neural Network is constructed and trained on the prepared data to learn the mapping between lip movements and speech. Finally, in the results stage, the trained CNN model and its weights are saved for future deployment. Its predictions are generated based on the model's interpretation of the video data and this performance is then evaluated. This structured methodology ensures an efficient and systematic approach to video analysis, with each stage seamlessly building upon the previous.

I have a few stretch goals that I hope to achieve, depending on the level of success and speed of progression through the rest of my work. One of these goals is to revisit my context-based ideas and explore whether they can still be implemented in a useful way. I am holding onto this concept because it remains one of the more novel aspects of my approach, and has the potential to contribute to the field and improve overall accuracy, even if provided less accurate results directly from the model. Another stretch goal is to extend this project to handle more complex, real-world examples. Specifically, I aim to develop a more generalizable method capable of processing wild videos. Achieving this would require building a system that can process diverse videos such that the mouth regions are structured similarly to those in the training data. This would demand more advanced facial recognition techniques and improved image manipulation methods but I believe this would be a significant improvement for my project.

All in all, I believe these steps represent solid progress toward completing this project. While there are undeniable challenges like dataset limitations and the difficulties of processing real-world videos I feel confident that I have identified these obstacles and developed a method to work around them. Looking ahead, I’m optimistic about refining the model, broadening its range of applications, and ultimately creating an effective and practical final program.

Works Cited

Assael, Y. M., Shillingford, B., Whiteson, S., & Freitas, N. de. (2016, November 4). *LipNet: End-to-end sentence-level lipreading*. OpenReview. https://openreview.net/forum?id=BkjLkSqxg

Cooke, M., Barker, J., Cunningham, S., & Shao, X. (2006). The Grid Audio-Visual Speech Corpus (1.0) [Data set]. Zenodo. https://doi.org/10.5281/zenodo.3625687

Chung, J. S., Senior, A., Vinyals, O., & Zisserman, A. (2017). Lip reading sentences in the wild. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. https://doi.org/10.1109/cvpr.2017.367

Ephrat, A., Mosseri, I., Lang, O., Dekel, T., Wilson, K., Hassidim, A., Freeman, W. T., & Rubinstein, M. (2018). Looking to listen at the Cocktail Party. *ACM Transactions on Graphics*, *37*(4), 1–11. https://doi.org/10.1145/3197517.3201357

H, Ramya, Sundararajan G, & Kumaran M. (2023). LipNet: Bridging Communication Gaps through Real-time Lip Reading and Speech Recognition. *International Journal of Advanced Research in Computer and Communication Engineering, 12*(9), 100-104. https://ijarcce.com/wp-content/uploads/2023/10/IJARCCE.2023.12917.pdf

Mathew, A., Saldanha, A. & Babu, C.N. Audio–video syncing with lip movements using generative deep neural networks. *Multimed Tools Appl* 83, 82019–82033 (2024). https://doi.org/10.1007/s11042-024-18695-x

Urban, E. (n.d.). *Get facial position with Viseme - Azure AI services*. Azure AI services | Microsoft Learn.https://learn.microsoft.com/en-us/azure/ai-services/speech-service/how-to-speech-synthesis-viseme?tabs=visemeid&pivots=programming-language-python