Sub-word Level Lip Reading With Visual Attention

K R Prajwal   Triantafyllos Afouras   Andrew Zisserman

Visual Geometry Group, University of Oxford

{prajwal,afourast,az}@robots.ox.ac.uk

https://www.robots.ox.ac.uk/~vgg/research/vtp-for-lip-reading

Abstract

The goal of this paper is to learn strong lip reading models that can recognise speech in silent videos. Most prior works deal with the open-set visual speech recognition problem by adapting existing automatic speech recognition techniques on top of trivially pooled visual features. Instead, in this paper, we focus on the unique challenges encountered in lip reading and propose tailored solutions. To this end, we make the following contributions: (1) we propose an attention-based pooling mechanism to aggregate visual speech representations; (2) we use sub-word units for lip reading for the first time and show that this allows us to better model the ambiguities of the task; (3) we propose a model for Visual Speech Detection (VSD), trained on top of the lip reading network. Following the above, we obtain state-of-the-art results on the challenging LRS2 and LRS3 benchmarks when training on public datasets, and even surpass models trained on large-scale industrial datasets by using an order of magnitude less data. Our best model achieves 22.6% word error rate on the LRS2 dataset, a performance unprecedented for lip reading models, significantly reducing the performance gap between lip reading and automatic speech recognition. Moreover, on the AVA-ActiveSpeaker benchmark, our VSD model surpasses all visual-only baselines and even outperforms several recent audio-visual methods.

1. Introduction

Lip reading, or visual speech recognition, is the task of recognising speech from silent video. It has many practical applications which include improving speech recognition in noisy environments, enabling silent dictation, or dubbing and transcribing archival silent films [25]. It also has important medical applications, such as helping speech impaired individuals, e.g. people suffering from Lou Gehrig's disease speak [54], or enabling people with aphonia (loss of voice) to communicate just by using lip movements.

Lip reading and audio-based automatic speech recog-

nition (ASR) both have the common goal of transcribing speech, however, they differ regarding the input: while in ASR the input signal is an audio waveform, in essence, a one-dimensional time series, lip reading has to deal with high-dimensional video inputs that have both temporal and spatial complexity. This makes training large end-to-end models harder due to GPU memory and computation constraints. Furthermore, understanding speech from visual information alone is challenging due to the inherent ambiguities present in the visual stream, i.e. the existence of homophemes where different characters that are visually indistinguishable (e.g. 'pa', 'ba', and 'ma'). That lip reading is a much harder task is also supported by the fact that although humans can understand speech reasonably well even in the presence of noise and across a variety of accents, they perform relatively poorly on lip reading [7, 16].

Designing a lip reading model requires both a visual component – mouth movements need to be identified – as well as a temporal sequence modelling component, which typically involves learning a language model that can resolve ambiguities in individual lip shapes. Recent developments in deep learning models and the availability of large-scale annotated datasets has led to breakthroughs surpassing human performance [16]. However, most of these works have taken the approach of adapting techniques used for ASR and machine translation, without catering to the particularities of the vision problem.

The conjecture in this paper is that the performance of lip reading, in terms of both accuracy and data efficiency, can be improved if the model is designed from the start taking into account the peculiarities of the visual, rather than the audio domain. To this end, we consider both the visual encoding and the text tokenisation.

Visual encoding. Our first contribution is the design of a novel visual backbone for lip reading. The spatiotemporal complexity in lip reading requires dealing with problems such as tracking the mouth in moving talking heads. This is usually achieved with complicated pre-processing pipelines based on facial landmarks. However, those are sub-optimal in many cases. For example, landmarks don't work well in

profile views [26]. Moreover, it is unclear what is the optimal region-of-interest for lip reading: it has been shown that besides the lips, other parts of the face, e.g. the cheeks, may also contain useful discriminative information [68]. Also, this region-of-interest can vary drastically in terms of scale, aspect ratio across identities and utterances. Thus, in this work, we propose an end-to-end trainable attention-based pooling mechanism that learns to track and aggregate the lip movement representations, resulting in a significant performance boost.

Text tokenisation. Lip reading methods most commonly output character-level tokens. This output representation however is sub-optimal as characters are sometimes more fine-grained than the input, with multiple characters corresponding to a single video frame. Furthermore, characters do not encode any prior knowledge about the language, which leads to higher dependency on the decoder's language modeling capacity that must also 'learn to read'. In this work, we instead use sub-word tokens (word-pieces) which not only match with multiple adjacent frames but are also semantically meaningful for learning a language easily. Word-pieces result in much shorter (than character) output sequences which greatly reduces the run-time and memory requirements. They also provide a language prior, reducing the language modelling burden of the model. We experimentally compare character and word-piece tokenization to justify this choice.

Visual Speech Detection. One issue with performing lip reading inference on real-world silent videos is that, since there is no audio track, there is no automated procedure for cropping out the clips where the person is speaking. ASR models use Voice Activity Detection (VAD) as a key pre-processing step, but this is clearly not applicable for silent videos. Here, the parts of a video containing speech have to be determined using the video input alone; in other words, by performing Visual Speech Detection (VSD). This can be very useful *e.g.* for running inference on silent movies. Among other findings in this work, we show that it is possible to train a strong VSD model on top of our pre-trained lip reading encoder.

Other downstream tasks. Besides improving performance on the sentence-level lip reading task itself, obtaining improved lip movement representations can have a broader impact, as those are often used for other related downstream tasks – e.g. sound source separation [19], visual keyword spotting [42], and visual language identification [4].

In summary, we make the following three contributions: (i) a visual backbone architecture using attention-based pooling on the spatial feature map; (ii) the use of sub-word units, rather than characters for the language tokens; and (iii) a strong Visual Speech Detection model, directly trained on top of the lip reading encoder.

In the experiments we show the benefits of (i) and (ii)

on improving lip reading performance, and we also introduce a two-stage training protocol that simplifies the curriculum used in prior works. As will be seen, with these design choices and training methodology, the performance of our best models exceeds prior work on standard evaluation benchmarks, and even outperforms proprietary models that use an order of magnitude more data for training. Similarly, we show the benefit of (i) and the lip reading encoder to our visual speech detection model that is far superior to previous methods on a standard evaluation benchmark.

We discuss potential ethical concerns and limitations of our work in the arXiv version. Please check our project page for video examples, code, and pre-trained models.

2. Related Work

We present an overview of prior work on lip reading, including a discussion of how these methods select and track the visual regions of interest, as well as the output tokenizations they use, followed by a brief overview of the use of attention for visual feature aggregation in other domains.

Lip reading. Early works on lip reading relied on hand-crafted pipelines and statistical models for visual feature extraction and temporal modelling [21, 37, 43, 44, 48]; an extensive review of those methods is presented in [70]. The advent of deep learning and the availability of large-scale lip reading datasets such as LRS2 [15] and LRS3 [2], rejuvenated this area. Progress was initially on word-level recognition [16, 58], and then moved onto sentence-level recognition by adapting models developed for ASR using LSTM sequence-to-sequence [15] or CTC [7, 54] approaches. [47] take a hybrid approach, training an LSTM-based sequence-to-sequence model with an auxiliary CTC loss. One trend in recent work is moving to Transformer-based architectures [1], or variants using convolution blocks [67], and hybrid architectures like a Conformer [22]. Another trend is to investigate the benefits of training with larger datasets, either directly by training on proprietary data that is orders of magnitude larger than any public dataset [40], or indirectly by distilling ASR models into lip reading ones [3, 33, 65]. For visual feature extraction and short-term dynamic modelling, most modern pipelines rely on spatiotemporal CNNs consisting of multiple 3D convolutional layers [7, 54], or more lightweight alternatives that comprise a single 3D convolutional layer followed by 2D ones [1, 16, 58] applied frame-wise.

Mouth ROI selection, registration, and tracking. A thorough investigation on facial region of interest (ROI) selection for lip reading is provided by [68]. The videos included in datasets like LRS2 and LRS3 are commonly pre-processed with a face detection and tracking pipeline which outputs clips roughly centered around the speaker's face. Many previous works use a central crop on the provided videos as input to the feature extractors [1, 38, 58]. More

elaborate pipelines use facial landmarks to register the face to a canonical view and/or only extract the crops of the mouth area [7, 31, 40, 46, 54, 67]. [68] propose to input a large part of the face, combined with Cutout [17] to encourage the model to also use the extra-oral face regions. After selecting which input region to extract the low-level CNN features from, all above works apply Global Average Pooling (GAP) on the extracted visual features map; this obtains a compact representation, but discards spatial information. Recent works [67] have shown that replacing GAP with a spatio-temporal fusion module improves performance.

Text tokenization. Most prior works on lip reading output character-level predictions [1, 14, 15, 38, 40, 47, 67]. Those approaches usually use an external language model during inference to boost performance [27, 39]. Instead [54] chose to output phoneme sequences, using phonetic dictionaries. This approach has the advantage of a more accurate mapping of lip-movements to sounds but requires a complicated decoding pipeline involving a proprietary finite-state-transducer. [20, 30] use a hard-crafted heuristic to map words onto viseme sequences and vice versa, and use viseme tokens for representing the output and target text. In this work, we instead propose using sub-word level tokenisation, which greatly reduces the output sequence length, thus accelerating both training and inference, and neatly encodes prior language information that improves the overall performance.

Visual feature aggregation with attention. Our work is also related to methods that use attention for improving visual representations of images or videos. [24, 61] use attention-weighted-averages of visual features as building blocks for various classification and detection tasks, while OCNet [66] uses self-attention to model context between pixels for semantic segmentation. Several recent papers have replaced convolutions with Transformer [60] blocks in visual representation pipelines. DETR [11] and Efficient DETR [64] learn object detectors by applying spatial transformers on top of CNN feature extractors. Similarly, the Visual Transformer [62] tokenises low-level CNN features and then processes them using a Transformer to model relationships between tokens. ViT [18] completely replaces CNNs in the visual pipeline with Transformer layers applied on image patch sequences, while the Timesformer [10] has been suggested as a purely Transformer-based solution for video representation learning.

Speech Detection. An important pre-processing stage in ASR pipelines is Voice Activity Detection (VAD), which involves the detection of the presence of speech in audio [51]. The reliability of audio-based VAD systems deteriorates in the presence of noise or cocktail party scenarios [35]. In audio-visual pipelines, such as ones used for creating large-scale audio-visual speech datasets [2, 16], this step is commonly replaced by an Active Speaker Detection

(ASD) stage which determines face tracks that match the speech. Audio-visual ASD models have been effectively trained either using direct supervision [6, 13, 32, 52, 59] or in a self-supervised [5, 16] fashion by employing contrastive objectives. The visual counterpart to VAD is Visual Speech Detection (VAD), which operates only on the video input. Early work on VSD (also termed Visual VAD or V-VAD) was based on handcrafted visual features and statistical modelling using methods such as HMMs, GMMs and PCA [8, 34–36, 45, 49, 55–57]. More recent works proposed methods based on optical flow [9] or a combination of CNN and LSTMs [23, 53]. These methods are limited in having been trained or evaluated on constrained or non-public datasets. The train set of WildVVAD [23], a new annotated VSD dataset has been made public, however, at the time of publication of this paper, its test set was not available, we were, therefore, unable to use this dataset for benchmarking.

3. Method

In this section, we describe our proposed method. The architecture of the model is outlined in Figure 1. Next, we explain each stage of the pipeline and refer the reader to the arXiv version for further details.

3.1. Visual backbone

CNN. The input to the pipeline is a silent video clip of T frames, $\boldsymbol{x} \in \mathbb{R}^{T \times H \times W \times 3}$. A spatio-temporal residual CNN is applied on sub-clips of 5 frames (i.e. 0.2s) with a unit frame stride, to extract visual spatial feature maps $\boldsymbol{f} \in \mathbb{R}^{T \times h \times w \times c}$. For our best model, $H = W = 96$, $(h, w) = (H/4, W/4) = (24, 24)$, and $c = 128$.

Visual Transformer Pooling (VTP). The CNN feature map $\boldsymbol{f}_t \in \mathbb{R}^{hw \times c}$ corresponding to every input frame $t \in \{1, \ldots, T\}$ is processed individually by a shared Visual Transformer Pooling (VTP) block. The feature map is first flattened to $\boldsymbol{f}_t \in \mathbb{R}^{h \times w \times c}$ and projected to a desired Transformer feature dimension d to get $\boldsymbol{f}_t \in \mathbb{R}^{hw \times d}$. Then, spatial positional encodings (SPE) are added to it; the result is passed through an encoder consisting of N_{VTP} Transformer layers, to get an enhanced self-attended feature map

$$\boldsymbol{z}_t = encoder_v(\boldsymbol{f}_t + SPE_{1:hw}) \in \mathbb{R}^{hw \times d}.$$

A learnable query vector $\boldsymbol{Q_{att}} \in \mathbb{R}^{d \times 1}$ is then used to extract a visual attention mask

$$\boldsymbol{a}_t = softmax(\boldsymbol{Q_{att}}^\top \boldsymbol{z}_t) \in \mathbb{R}^{hw \times 1}.$$

The attention mask is used to compute a weighted average over the self-attended feature map

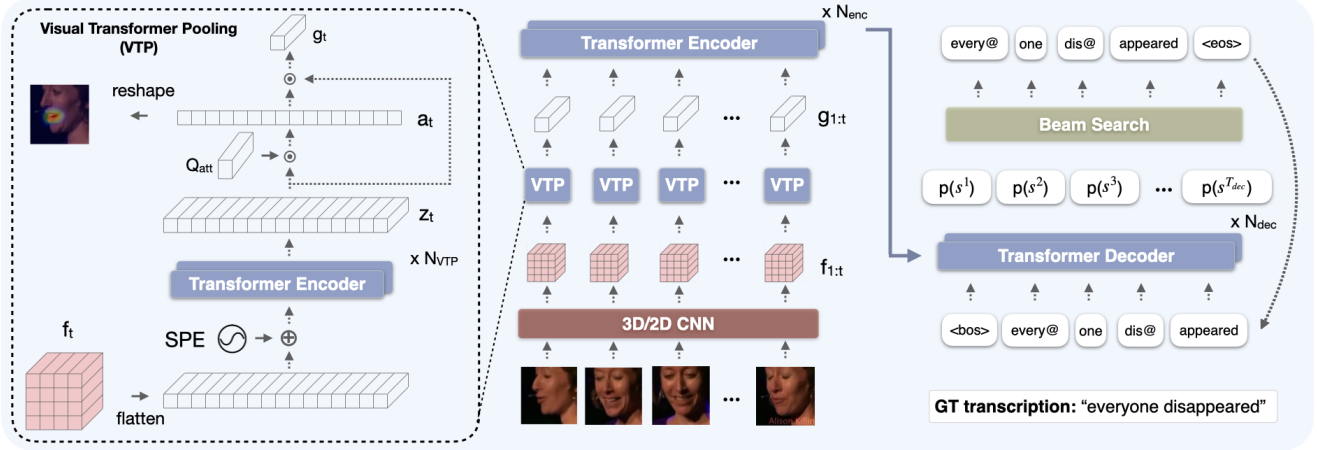$$\boldsymbol{g}_t = \frac{1}{hw} \sum_{u=1}^{hw} a_t^u z_t^u \in \mathbb{R}^d$$

Figure 1. **Proposed lip reading architecture.** *Left:* The input video frames are passed through a spatio-temporal CNN to extract low-level visual features $\boldsymbol{f}$. The feature map corresponding to every input frame is then separately processed by a Visual Transformer Pooling module (VTP). The VTP block adds spatial positional encodings (SPE) to the input features and passes the result through a Transformer encoder to produce a self-attended feature map $\boldsymbol{z}_t$. A query vector $\boldsymbol{Q}_{att}$ is used to compute an attention mask which is in turn used to obtain a spatially weighted average of $\boldsymbol{z}_t$. This produces a compact visual representation of the lip appearance and movement around each input video frame. Concatenating the frame-wise features forms a temporal feature sequence $\boldsymbol{g}$. This is passed as input to an encoder-decoder Transformer (*right*) that auto-regressively predicts sub-word probabilities for one token at a time. An output sentence is eventually inferred from these distributions using a beam search.

where a_t^u and z_t^u denote the feature and attention weight respectively, associated with frame t and location $u \in \{1, \ldots, hw\}$. By stacking the resulting vectors $\boldsymbol{g}_t$ in time, we obtain an embedding sequence $\boldsymbol{g} = (\boldsymbol{g}_1, \boldsymbol{g}_2 \cdots, \boldsymbol{g}_T) \in \mathbb{R}^{T \times d}$ which contains a compact spatio-temporal representation for every input frame.

Transformer encoder-decoder. An encoder-decoder Transformer model is used to predict a text token sequence $s = (s_1, s_2 \cdots, s_{T_{dec}})$ from the source video embedding sequence $\boldsymbol{g}$, one token at a time: temporal positional encodings (PE) are added to $\boldsymbol{g}$, and the result is input to an encoder, which consists of N_{enc} multi-head Transformer layers, to produce a self-attended embedding sequence

$$\boldsymbol{g_{enc}} = \text{ENCODER}(\boldsymbol{g} + PE_{1:T}) \in \mathbb{R}^{T \times d}.$$

The decoder, which consists of N_{dec} Transformer layers, then attends on this sequence and predicts the output text token sequence s in an auto-regressive manner, by factorising its joint probability:

$$\log p(s|\boldsymbol{x}) = \sum_{t=1}^{T_{dec}} \log p(s_t | \boldsymbol{g_{enc}}(\boldsymbol{x}), s_{1:t-1}) \qquad (1)$$

where positional encodings have also been added to the auto-regressive decoder inputs as in [60].

The text sentences are encoded into token sequences (and vice versa tokens are decoded into text) using a sub-word level tokeniser, in particular WordPiece [63]. We tried other sub-word tokenizations such as Byte-Pair-Encoding (BPE) that is used in GPT2 [50], but it performed worse compared to using WordPiece.

Beam search decoding and rescoring. Decoding is performed with a left-to-right beam search of width B. We also decode a second time after flipping all the input video frames horizontally. Additional language knowledge is incorporated by using an external language model (LM) to re-score [12] the $2 \times B$-best hypotheses $S = \{s_1 \cdots s_B; s_1^h \cdots s_B^h\}$ from the beam searches, and obtain the highest scoring one as the final sentence prediction:

$$s_{best} = \underset{s \in S}{\arg\max} \left[\alpha \log p(s|\boldsymbol{x}) + (1 - \alpha) \log p_{LM}(s) \right]$$

Here, $s_1^h \cdots s_B^h$ denotes the beam sequences after horizontally flipping the input. We found that additional test-time augmentations such as small degrees of rotation and/or color jitters did not improve the results.

3.2. Training

Optimisation objective. Given a training dataset $\mathcal{D}$ consisting of pairs (x, s^*) of video clips and their ground truth transcriptions, the model is trained to maximise the log likelihoods of the transcriptions by optimising the following objective

$$\mathcal{L} = -\mathbb{E}_{(x,s^*) \in \mathcal{D}} \log p(s^*|\boldsymbol{x}) \qquad (2)$$

Teacher forcing. To accelerate training, we follow common practice for sequence-to-sequence training with Transformers, and feed in the previous ground-truth token as the decoder input at every step, instead of using auto-regression. The tokens are fed into the decoder via a learnable embedding layer.

Training protocol. Training is performed in two stages. First, the whole network is trained end-to-end on short phrases of 2 words. Following [1, 16], we use frame word-boundaries to crop out training samples from all the possible combinations of 2 consecutive words in the dataset, which provides natural augmentation. Upon convergence, we freeze the visual backbone, then pre-extract and dump the visual features for all the samples. In the second training stage that follows, we train the encoder-decoder sub-network on all possible sub-sequences (n-grams) of length 2 or larger that can be generated by combining consecutive word utterances in the dataset.

Discussion. We note that our training protocol is much simpler than the ones commonly used in prior works [1,16,47]. By using the same network and loss during the backbone pre-training stage, we obtain a good initialization of the entire network and enable a smooth transfer. This is in contrast to other works that pre-train with a different proxy loss and require a separate word classification head that is subsequently discarded. The second stage is significantly simpler to implement and requires a single run, unlike curriculums that gradually increase the length of the training sentences and usually require a complicated tuning process with multiple manual restarts to achieve the best results. We observed that our proposed second stage training setup matches the performance of the complicated curriculum strategy used in previous works, while being more efficient in terms of training time and manual efforts.

4. Experiments

4.1. Data

LRS2 & LRS3. For training and evaluation, we use two publicly available sentence-level lip reading datasets: LRS2 [16] and LRS3 [2]. LRS2 contains video clips from a variety of shows from British television, such as Countryfile and Top Gear; the transcribed content sums up to approximately 224 hours in total. LRS3 has been collected from over $5,000$ TED and TEDx talks in English, available on YouTube, totalling 475 hours. Both datasets have been created using a detection and tracking pipeline that produces face-cropped clips roughly centered around the speaker's talking head. All videos are available at a 224×224 pixel resolution and 25 fps. The datasets contain a "pretrain" partition that includes extensive head tracks including word boundaries that have been produced by force-aligning subtitles to the audio. Those word alignments enable training at any granularity. The test sets contain only full sentences.

Additional dataset: TEDx_ext. To obtain more training data, we create a new dataset from TEDx talks downloaded from YouTube, by using a pipeline similar to [2]. We collect $13,211$ TEDx talks in English that are not included in LRS3. Unlike the videos used for the creation

of LRS3 which contain manually annotated transcripts, the new videos only have the closed captions automatically produced by the YouTube ASR system. As these captions are only approximately aligned to the audio, we use the Montreal Force Aligner [41] to obtain accurate alignments for the word boundaries needed by our training pipeline. For the rest of the processing (face detection, tracking, and cropping) we used the same pipeline as in [2]. The resulting training dataset contains $1,204$ hours in total over $318,459$ visual speech tracks, including text transcriptions with word boundary alignment. We call this new training set TEDx_ext. We note that since this pipeline does not require any manual transcriptions, the supervision comes for free, therefore it is easily scalable. However, the supervision is not as strong due to the noise in the training data introduced by the imperfect ASR transcriptions. But, as we will see, our model achieves a huge performance boost after training on this noisy data.

4.2. Implementation details

During the first training stage, we apply random visual augmentations on the input frames to reduce overfitting: the input videos are first resized to a square 160 pixels resolution, from which a central square 96-pixel crop is extracted. Random horizontal flipping and rotation (up to $10°$) are also applied before inputting to the lip reading pipeline. During inference we use the central 96-pixel crop, and only apply the horizontal flipping augmentation.

For our best model, *i.e.*, VTP on $(H/4, W/4)$, we set $N_{VTP} = 6$ layers with 8 heads each for the encoder of the VTP module. For computational efficiency, VTP uses the recently proposed Linear Transformer [28] instead of the original Transformer [60]. We found that this change did not lead to a drop in the recognition performance, while being much more computationally efficient. Another design choice we had to make is deciding after which CNN layer the VTP should be applied. Transformer layers are computationally expensive at higher resolution feature maps (*i.e.* earlier layer activations), but can capture more detailed information. Given this trade-off, we experiment with three different feature map resolutions, at $(h, w) = (H/4, W/4), (H/8, W/8), (H/16, W/16)$. For the latter two variants, we set the feature dimension $d = 512$. When pooling on $(h, w) = (H/4, W/4)$, we keep the compute and memory needs in check by performing two small changes: using $d = 256$ for the first 3 VTP layers, and then setting $d = 512$ but down-sampling the feature map to $(H/8, W/8)$ for the remaining 3 layers.

The encoder-decoder Transformer contains $N_{enc} = 6$ and $N_{dec} = 6$ layers, each with 8 attention heads. We use sinusoidal positional encodings [60] for PE and learnable positional encodings for SPE. We use the WordPiece tok-

| Method | Training | | Evaluation | |
	Datasets used	Total # hours	LRS2	LRS3
LIBS [69]	LRS2, LRS3	698	65.3	-
Hyb. CTC/Att. [47]	LRS2, LRW	389	63.5	-
TDNN [65]	LRS2	224	48.9	-
Conv-seq2seq [67]	LRS2, LRS3	698	51.7	60.1
CTC + KD [3]	LRS2, LRS3, VoxCeleb2‡	1,032	51.3	59.8
Hyb. + Conformer [38]	LRS2, LRW	389	37.9	-
Hyb. + Conformer [38]	LRS3, LRW	639	-	43.3
Ours	LRS2, LRS3	698	**28.9**	**40.6**
TM-seq2seq [1]	LRS2, LRS3, LRW, MV-LRS†	1,637	48.3	58.9
CTC-V2P [54]	LSVSR†	3,886	-	55.1
RNN-T [40]	YT31k†	31,000	-	33.6
Ours	LRS2, LRS3, MV-LRS†, TEDx$_{ext}$	2,676	**22.6**	**30.7**

Table 1. Comparison of different lip reading models on the test sets of the LRS2 and LRS3 datasets in terms of Word Error Rate % (WER, lower is better), along with the datasets and the aggregate number of hours used for training each model. Our model achieves state-of-the-art results, outperforming all previous baselines when trained on publicly available data (i.e. LRS2 and LRS3). If we additionally use MV-LRS and TEDx$_{ext}$ for training, then our best model obtains results comparable with that of [40], even though we are only using an order of magnitude less data. This is indicative of the data efficiency of our proposed pipeline. †Large non-public labelled datasets: MV-LRS [1] contains 730 hours, LSVSR [54] 3.9k hours, and YT31k [40] 31k hours of transcribed video. ‡unlabelled dataset. Results shown in blue have been obtained by training (partly or entirely) on non-public data.

enizer of the BERT model in HuggingFace[1], with a vocabulary of 30522 tokens. We also use an off-the-shelf pretrained GPT2 language model for beam rescoring. For the beam rescoring, we set hyperparameter $\alpha = 0.7$ for LRS2 and $\alpha = 0.6$ for LRS3 respectively. We train all models with the Adam optimiser [29] with $\beta_1 = 0.9$, $\beta_2 = 0.98$ and $\epsilon = 10^{-9}$. In the first stage of the training, we follow a Noam learning rate schedule [60] for the first 50 epochs and then reduce the learning rate by a factor of 5 every time the validation loss plateaus, until reaching 10^{-6}. For the second stage, the learning rate is initially set to $5e^{-5}$ and reduced by a factor of 5 on plateau down to 10^{-6}. For our best reported models on public data, the first stage of training takes approximately 14 days on 4 Tesla v100s GPUs. The second stage takes 1.5 days on 1 Tesla v100 GPU.

4.3. State-of-the-art lipreading

We compare the results of our method to existing works in Table 1. It is clear that our best model outperforms all prior work trained on public data, on both the LRS2 and LRS3 benchmarks. In particular, compared to the strongest baseline of [38], our best model performs 9% better on LRS2 and 2.7% better on LRS3. When also using MV-LRS and TEDx$_{ext}$ for training, we obtain a significant boost, achieving 22.6% and 30.7% WER for LRS2 and LRS3 respectively. We even outperform [40] by a considerable mar-

gin while using $10\times$ lesser training data. This clearly suggests that our pipeline is highly data-efficient.

4.4. Ablations

We perform various ablations to better understand the different aspects of our pipeline. For all the ablation experiments the models are trained and evaluated on public data only, *i.e.* the LRS2 and LRS3 datasets.

Importance of each module. We show the impact of each proposed module in the final scores, starting from a variation of the TM-seq2seq model [1][2], and building up to our full model. We summarize the results of this study in Table 2. It is clear that all the proposed improvements give significant performance boosts and are largely orthogonal. In particular, the use of WordPiece tokens contributes a 3.8% absolute improvement on LRS2, while introducing the VTP module decreases the WER by 6.3%. Using LM to rescore the beams and applying test-time horizontal flipping leads to another 1.1% and 0.9% improvement respectively.

VTP resolution. The VTP module is capable of aggregating the spatial features at arbitrary feature map resolutions. But, we show that it is more effective when operating on finer high-resolution feature maps rather than coarser low-resolution feature maps. This is evident in Table 3, where pooling after $conv_{2,3}$ at a spatial resolution of 24×24 is much more effective than pooling on lower-resolution feature maps of 12×12 or 6×6.

---

[1]https://huggingface.co/transformers/pretrained_models.html

[2]Using the same CNN extractor as our model for fair comparison

Figure 2. Visualization of the visual attention masks $\boldsymbol{a}$ from the VTP module superimposed on the input frames that produce them. The video clips used here are random samples from the LRS3 dataset. It is evident that the model follows the more discriminative mouth region.

Method	WER	Δ
TM-seq2seq[†] baseline	41.0	-
+ WordPiece	37.2	−3.8
+ VTP	30.9	**−6.3**
+ Beam LM rescoring	29.8	−1.1
+ Test-time augmentation	28.9	−0.9

Table 2. Ablation on the design improvements proposed in this work. The results reported are for the test set of the LRS2 dataset. It is clear that all the proposed components contribute independently to the performance boost. [†]The baseline is an improved version of TM-seq2seq [1] (details are in the arXiv version).

Method	# transformer layers	WER
Without VTP	0	37.2
VTP @ $(H/16, W/16)$	2	35.7
VTP @ $(H/8, W/8)$	3	33.8
VTP @ $(H/4, W/4)$	6	30.9

Table 3. Ablation on the input spatial resolution for the VTP module. The number of Transformer layers for each stage is chosen such that the total number of parameters in the visual front-end is approximately the same. We see that pooling from higher resolution feature maps clearly leads to better results.

Training protocol. Previous works [1] follow a curriculum strategy during training: the sequence length is gradually increased over the course of the training. While this protocol, indeed works better, we argue that the boost in performance does not come from the curriculum learning but

from something else: data augmentation. Over the course of the training process, the model gets to train on all subsequences (n-grams) of various lengths, and this is an effective data augmentation that reduces over-fitting. Indeed, we observe that if we simply train on all n-gram sub-sequences at once (as opposed to a slow length increase), we achieve a WER of 30.92, which is comparable to the WER of 30.91 following a curriculum protocol. Not only does this experiment shed new light on the current understanding of the training pipeline of lip reading, it also achieves similar results while following a much simpler training process that requires significantly less manual tuning.

4.5. Visual attention visualization

In Figure 2, we visualize the visual attention maps that the VTP module produces. Note that the lips region is tracked very accurately while the speakers turn their heads around, even for extreme profile views.

5. Visual Speech Detection Application

We build a VSD model on top of our lip reading transformer encoder by simply adding a fully connected (FC) layer and a sigmoid activation on top of the frame-level encoder outputs to classify whether the person is speaking in that frame or not:

$$\boldsymbol{y} = \sigma(\,\mathrm{FC}\,(\boldsymbol{g_{enc}})\,) \in \mathbb{R}^T.$$

The architecture is illustrated in Figure 3. We train the VSD head on top of the pre-trained lip reading encoder using a
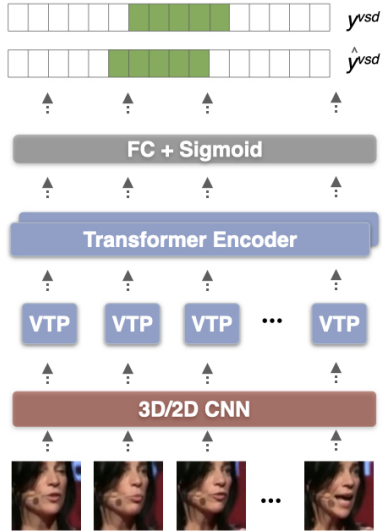
Figure 3. Visual Speech Detection pipeline. To predict the presence of speech at each video frame, we add a sigmoid prediction head on top of the lip reading model's visual encoder.

binary cross-entropy loss between the predicted and ground truth labels, y_t and y_t^*:

$$\mathcal{L}^v = \frac{1}{T} \sum_{t=1}^{T} y_t^* \log y_t + (1 - y_t^*) \log(1 - y_t) \quad (3)$$

Dataset and evaluation. We train our VSD model on the train split of the popular AVA ActiveSpeaker dataset [52]. This dataset is created from movies and contains 120 videos ($2.6M$ frames) for training, 33 videos ($768K$ frames) for validation and 109 videos ($2M$ frames) in the test set. Each frame contains bounding box annotations for the face, along with a label indicating if the person is (i) speaking and audible, (ii) speaking but not audible, (iii) inaudible. The second class covers cases where the person is visually speaking in the background, but his voice/speech is not audible. Since we operate only on the visual frames, we combine the samples of the first two classes and train the model to perform binary classification. We initialize the weights from our pretrained lip-reading models and fine-tune all the layers with the Adam optimizer by using a small learning rate of 10^{-6}. To evaluate the performance of our model and baselines we use the mean Average Precision (mAP) metric as defined by the dataset authors [52]; we compute the metric using the evaluation script[3] that the authors supply. We also report our scores on the held-out non-public test set with the assistance of the Ava-ActiveSpeaker challenge organizers.

Results. We show quantitative results in Table 4, where we report the VSD performance of our best VTP-based model (corresponding to the last row of Table 1), alongside results from previous works. It is evident that our model

---

greatly outperforms the video-only baseline of [52], and even outperforms several of the recently proposed audio-visual methods ([6, 13]). Moreover, to once again showcase the benefits of the VTP module, we also compare with an ablation of our model that uses Global Average Pooling (GAP) on top of the CNN features instead of VTP, with identical train-val-test settings. VTP outperforms this model on this task as well, by 8 mAP points. We refer to the project page for qualitative video examples and examples of inference on silent videos using the whole pipeline, including VSD and lip reading.

Method	A	V	mAP (val)	mAP (test)
Roth *et al.* [52]	✓	✓	79.2	82.1
Alcazar *et al.* [6]	✓	✓	87.1	86.7
Chung *et al.* [13]	✓	✓	87.8	87.8
MAAS-TAN [32]	✓	✓	88.8	88.3
TalkNet [59]	✓	✓	**92.3**	**90.8**
Roth *et al.* [52]	✗	✓	73.5	71.1
Ours (CNN + GAP)	✗	✓	81.4	80.2
Ours (VTP)	✗	✓	**89.2**	**88.2**

Table 4. Visual Speech Detection performance on the validation (*val*) and test sets of the AVA ActiveSpeaker benchmark dataset. The *A* and *V* columns denote which modalities the corresponding method uses as input. Our VTP model outperforms the baseline video-only model of [52] by a large margin (over 17 mAP improvement). In fact, we even outperform several of the recently proposed audio-visual methods ([6, 13]), obtaining results close to the current state-of-the-art without using any audio.

Limitations and Ethical considerations. We explore the limitations and failure cases of the lip reading and the VSD models in the arXiv version. We also discuss the ethical issues and the positive real-world applications of our work.

6. Conclusion

We have presented an improved architecture for lip reading based on attention-based aggregation of visual representations as well as several enhancements to the training protocol, including the use of sub-word tokenisation. Our best models achieve state-of-the-art results, outperforming prior work trained on public data by a significant margin, and even industrial models trained on orders of magnitude more data. We have also designed a Visual Speech Detection model on top of our lip reading system that obtains state-of-the-art results on this task and even outperforms several audio-visual baselines.

References

[1] Triantafyllos Afouras, Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. Deep audio-visual speech recognition. *IEEE PAMI*, 2019. 2, 3, 5, 6, 7

[2] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. LRS3-TED: a large-scale dataset for visual speech recognition. In *arXiv preprint arXiv:1809.00496*, 2018. 2, 3, 5

[3] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. ASR is all you need: Cross-modal distillation for lip reading. In *International Conference on Acoustics, Speech, and Signal Processing*, 2020. 2, 6

[4] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. Now you're speaking my language: Visual language identification. In *INTERSPEECH*, 2020. 2

[5] Triantafyllos Afouras, Andrew Owens, Joon Son Chung, and Andrew Zisserman. Self-supervised learning of audio-visual objects from video. In *Proc. ECCV*, 2020. 3

[6] Juan León Alcázar, Fabian Caba, Long Mai, Federico Perazzi, Joon-Young Lee, Pablo Arbeláez, and Bernard Ghanem. Active speakers in context. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12465–12474, 2020. 3, 8

[7] Yannis M. Assael, Brendan Shillingford, Shimon Whiteson, and Nando de Freitas. Lipnet: Sentence-level lipreading. *arXiv:1611.01599*, 2016. 1, 2, 3

[8] Andrew Aubrey, Bertrand Rivet, Yulia Hicks, Laurent Girin, Jonathon Chambers, and Christian Jutten. Two novel visual voice activity detectors based on appearance models and retinal filtering. In *2007 15th European Signal Processing Conference*, pages 2409–2413, 2007. 3

[9] Andrew J. Aubrey, Yulia A. Hicks, and Jonathon A. Chambers. Visual voice activity detection with optical flow. *Iet Image Processing*, 4:463–472, 2010. 3

[10] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *Proc. ICML*, 2021. 3

[11] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *arXiv preprint arXiv:2005.12872*, 2020. 3

[12] William Chan, Navdeep Jaitly, Quoc V. Le, and Oriol Vinyals. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *ICASSP*, 2016. 4

[13] Joon Son Chung. Naver at activitynet challenge 2019–task b active speaker detection (ava). *arXiv preprint arXiv:1906.10555*, 2019. 3, 8

[14] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. VoxCeleb2: Deep speaker recognition. In *INTERSPEECH*, 2018. 3

[15] Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. Lip reading sentences in the wild. In *Proc. CVPR*, 2017. 2, 3

[16] Joon Son Chung and Andrew Zisserman. Lip reading in the wild. In *Proc. ACCV*, 2016. 1, 2, 3, 5

[17] Terrance DeVries and Graham W. Taylor. Improved regularization of convolutional neural networks with cutout. In *arXiv preprint arXiv:1708.04552*, 2017. 3

[18] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 3

[19] Ariel Ephrat, Inbar Mosseri, Oran Lang, Tali Dekel, Kevin Wilson, Avinatan Hassidim, William T Freeman, and Michael Rubinstein. Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation. *ACM Transactions on Graphics*, 37(4):1–11, 2018. 2

[20] Souheil Fenghour, Daqing Chen, Kun Guo, and Perry Xiao. Lip reading sentences using deep learning with only visual cues. *IEEE Access*, 8:215516–215530, 2020. 3

[21] John N Gowdy, Amarnag Subramanya, Chris Bartels, and Jeff Bilmes. Dbn based multi-stream models for audio-visual speech recognition. In *2004 IEEE International conference on acoustics, speech, and signal processing*, volume 1, pages I–993. IEEE, 2004. 2

[22] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang. Conformer: Convolution-augmented transformer for speech recognition. In *INTERSPEECH*, 2020. 2

[23] Sylvain Guy, Stéphane Lathuilière, Pablo Mesejo, and Radu Horaud. Learning visual voice activity detection with an automatically annotated dataset. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 4851–4856. IEEE, 2021. 3

[24] Saumya Jetley, Nicholas A Lord, Namhoon Lee, and Philip HS Torr. Learn to pay attention. In *Proc. ICLR*, 2018. 3

[25] Abhishek Jha, Vinay Namboodiri, and C. Jawahar. Spotting words in silent speech videos: a retrieval-based approach. *Machine Vision and Applications*, 30, 03 2019. 1

[26] Benjamin Johnston and Philip Chazal. A review of image-based automatic facial landmark identification techniques. *EURASIP Journal on Image and Video Processing*, 2018:86, 09 2018. 2

[27] Anjuli Kannan, Yonghui Wu, Patrick Nguyen, Tara N. Sainath, Zhifeng Chen, and Rohit Prabhavalkar. An analysis of incorporating an external language model into a sequence-to-sequence model. In *Proc. ICASSP*, 2018. 3

[28] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning*, pages 5156–5165. PMLR, 2020. 5

[29] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *Proc. ICLR*, 2014. 6

[30] Oscar Koller, Hermann Ney, and Richard Bowden. Deep learning of mouth shapes for sign language. In *2015 IEEE International Conference on Computer Vision Workshop (IC-CVW)*, pages 477–483, 2015. 3

[31] Alexandros Koumparoulis, Gerasimos Potamianos, Youssef Mroueh, and Steven J. Rennie. Exploring roi size in deep learning based lipreading. In *Proc. The 14th International Conference on Auditory-Visual Speech Processing*, pages 64–69, 2017. 3

[32] Juan León-Alcázar, Fabian Caba Heilbron, Ali Thabet, and Bernard Ghanem. Maas: Multi-modal assignation for active speaker detection. In *Proc. ICCV*, 2021. 3, 8

[33] Wei Li, Sicheng Wang, Ming Lei, Sabato Marco Siniscalchi, and Chin-Hui Lee. Improving audio-visual speech recognition performance with cross-modal student-teacher training. In *Proc. ICASSP*, pages 6560–6564. IEEE, 2019. 2

[34] Peng Liu and Zuoying Wang. Voice activity detection using visual information. In *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages I–609, 2004. 3

[35] Qingju Liu, Andrew J. Aubrey, and Wenwu Wang. Interference reduction in reverberant speech separation with visual voice activity detection. *IEEE Transactions on Multimedia*, 16(6):1610–1623, 2014. 3

[36] Qingju Liu, Wenwu Wang, and Philip Jackson. A visual voice activity detection method with adaboosting. In *Sensor Signal Processing for Defence (SSPD 2011)*, pages 1–5, 2011. 3

[37] Karen Livescu, Ozgur Cetin, Mark Hasegawa-Johnson, Simon King, Chris Bartels, Nash Borges, Arthur Kantor, Partha Lal, Lisa Yung, Ari Bezman, et al. Articulatory feature-based methods for acoustic and audio-visual speech recognition: Summary from the 2006 jhu summer workshop. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, volume 4, pages IV–621. IEEE, 2007. 2

[38] Pingchuan Ma, Stavros Petridis, and Maja Pantic. End-to-end audio-visual speech recognition with conformers. In *ICASSP*, 2021. 2, 3, 6

[39] Andrew L. Maas, Ziang Xie, Dan Jurafsky, and Andrew Y. Ng. Lexicon-free conversational speech recognition with neural networks. In *Proceedings the North American Chapter of the Association for Computational Linguistics*, 2015. 3

[40] Takaki Makino, Hank Liao, Yannis Assael, Brendan Shillingford, Basilio Garcia, Otavio Braga, and Olivier Siohan. Recurrent neural network transducer for audio-visual speech recognition. In *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2019. 2, 3, 6

[41] Michael McAuliffe, Michaela Socolof, Sarah Mihuc, M. Wagner, and Morgan Sonderegger. Montreal forced aligner: Trainable text-speech alignment using kaldi. In *INTERSPEECH*, 2017. 5

[42] Liliane Momeni, Triantafyllos Afouras, Themos Stafylakis, Samuel Albanie, and Andrew Zisserman. Seeing wake words: Audio-visual keyword spotting. In *Proc. BMVC*, 2020. 2

[43] Eng-Jon Ong and Richard Bowden. Learning temporal signatures for lip reading. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 958–965, 2011. 2

[44] George Papandreou, Athanassios Katsamanis, Vassilis Pitsikalis, and Petros Maragos. Adaptive multimodal fusion by uncertainty compensation with application to audiovisual speech recognition. *Audio, Speech, and Language Processing, IEEE Transactions on*, 17(3):423–435, 2009. 2

[45] Foteini Patrona, Alexandros Iosifidis, Anastasios Tefas, Nikolaos Nikolaidis, and Ioannis Pitas. Visual voice activity detection in the wild. *IEEE Transactions on Multimedia*, 18(6):967–977, 2016. 3

[46] Stavros Petridis, Themos Stafylakis, Pingchuan Ma, Feipeng Cai, Georgios Tzimiropoulos, and Maja Pantic. End-to-end audiovisual speech recognition. *CoRR*, abs/1802.06424, 2018. 3

[47] Stavros Petridis, Themos Stafylakis, Pingchuan Ma, Georgios Tzimiropoulos, and Maja Pantic. Audio-Visual Speech Recognition with a Hybrid CTC/Attention Architecture. In *IEEE Spoken Language Technology Workshop*, pages 513–520, 2018. 2, 3, 5, 6

[48] G. Potamianos, C. Neti, G. Gravier, A. Garg, and A.W. Senior. Recent advances in the automatic recognition of audiovisual speech. *Proceedings of the IEEE*, 91(9):1306–1326, 2003. 2

[49] Gerasimos Potamianos, C. Neti, Juergen Luettin, and Iain Matthews. Audio-visual automatic speech recognition: An overview. *Issues in audio-visual speech processing*, 2004. 3

[50] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 4

[51] Javier Ramirez, Juan Manuel Górriz, and José Carlos Segura. Voice activity detection. fundamentals and speech recognition system robustness. *Robust speech recognition and understanding*, 6(9):1–22, 2007. 3

[52] Joseph Roth, Sourish Chaudhuri, Ondrej Klejch, Radhika Marvin, Andrew C. Gallagher, Liat Kaver, Sharadh Ramaswamy, Arkadiusz Stopczynski, Cordelia Schmid, Zhonghua Xi, and Caroline Pantofaru. Ava-activespeaker: An audio-visual dataset for active speaker detection. In *ICASSP*, 2020. 3, 8

[53] Rahul Sharma, Krishna Somandepalli, and Shrikanth Narayanan. Toward visual voice activity detection for unconstrained videos. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 2991–2995, 2019. 3

[54] Brendan Shillingford, Yannis Assael, Matthew W. Hoffman, Thomas Paine, Cían Hughes, Utsav Prabhu, Hank Liao, Hasim Sak, Kanishka Rao, Lorrayne Bennett, Marie Mulville, Ben Coppin, Ben Laurie, Andrew Senior, and Nando de Freitas. Large-Scale Visual Speech Recognition. In *INTERSPEECH*, 2019. 1, 2, 3, 6

[55] Spyridon Siatras, Nikos Nikolaidis, Michail Krinidis, and Ioannis Pitas. Visual lip activity detection and speaker detection using mouth region intensities. *IEEE Transactions on Circuits and Systems for Video Technology*, 19(1):133–137, 2009. 3

[56] David Sodoyer, Bertrand Rivet, Laurent Girin, Christophe Savariaux, jean-luc Schwartz, and Christian Jutten. A study of lip movements during spontaneous dialog and its application to voice activity detection. *The Journal of the Acoustical Society of America*, 125:1184–96, 03 2009. 3

[57] D. Sodoyer, B. Rivet, L. Girin, J.-L. Schwartz, and C. Jutten. An analysis of visual speech information applied to voice activity detection. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, volume 1, pages I–I, 2006. 3

[58] Themos Stafylakis and Georgios Tzimiropoulos. Combining residual networks with lstms for lipreading. In *Interspeech*, 2017. 2

[59] Ruijie Tao, Zexu Pan, Rohan Kumar Das, Xinyuan Qian, Mike Zheng Shou, and Haizhou Li. Is someone speaking? exploring long-term temporal features for audio-visual active speaker detection. In *Proceedings of the 29th ACM International Conference on Multimedia*, page 3927–3935, 2021. 3, 8

[60] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. In *NeurIPS*, 2017. 3, 4, 5, 6

[61] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proc. CVPR*, pages 7794–7803, 2018. 3

[62] Bichen Wu, Chenfeng Xu, Xiaoliang Dai, Alvin Wan, Peizhao Zhang, Masayoshi Tomizuka, Kurt Keutzer, and Peter Vajda. Visual transformers: Token-based image representation and processing for computer vision. *CoRR*, abs/2006.03677, 2020. 3

[63] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144, 2016. 4

[64] Zhuyu Yao, Jiangbo Ai, Boxun Li, and Chi Zhang. Efficient DETR: improving end-to-end object detector with dense prior. *CoRR*, abs/2104.01318, 2021. 3

[65] Jianwei Yu, Shi-Xiong Zhang, Jian Wu, Shahram Ghorbani, Bo Wu, Shiyin Kang, Shansong Liu, Xunying Liu, Helen Meng, and Dong Yu. Audio-visual recognition of overlapped speech for the lrs2 dataset. In *ICASSP*, pages 6984–6988, 05 2020. 2, 6

[66] Yuhui Yuan and Jingdong Wang. Ocnet: Object context network for scene parsing. In *arXiv preprint arXiv:1809.00916*, 2018. 3

[67] Xingxuan Zhang, Feng Cheng, and Shilin Wang. Spatio-temporal fusion based convolutional sequence learning for lip reading. In *Proc. ICCV*, pages 713–722, 2019. 2, 3, 6

[68] Yuanhang Zhang, Shuang Yang, Jingyun Xiao, Shiguang Shan, and Xilin Chen. Can We Read Speech Beyond the Lips? Rethinking RoI Selection for Deep Visual Speech Recognition. In *arXiv preprint arXiv:2003.03206*, 2020. 2, 3

[69] Ya Zhao, Rui Xu, Xinchao Wang, Peng Hou, Haihong Tang, and Mingli Song. Hearing lips: Improving lip reading by distilling speech recognizers. In *arXiv preprint arXiv:1911.11502*, 2019. 6

[70] Ziheng Zhou, Guoying Zhao, Xiaopeng Hong, and Matti Pietikäinen. A review of recent advances in visual speech decoding. *Image and vision computing*, 32(9):590–605, 2014. 2