Computer Vision and Image Understanding 173 (2018) 76–85

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Learning to lip read words by watching videos

Joon Son Chung*, Andrew Zisserman

Visual Geometry Group, Department of Engineering Science, University of Oxford, United Kingdom

ARTICLE INFO

Keywords:
Lip reading
Lip synchronisation
Active speaker detection
Large vocabulary
Dataset

ABSTRACT

Our aim is to recognise the words being spoken by a talking face, given only the video but not the audio. Existing works in this area have focussed on trying to recognise a small number of utterances in controlled environments (*e.g.* digits and alphabets), partially due to the shortage of suitable datasets.

We make three novel contributions: first, we develop a pipeline for fully automated data collection from TV broadcasts. With this we have generated a dataset with over a million word instances, spoken by over a thousand different people; second, we develop a two-stream convolutional neural network that learns a joint embedding between the sound and the mouth motions from unlabelled data. We apply this network to the tasks of audio-to-video synchronisation and active speaker detection; third, we train convolutional and recurrent networks that are able to effectively learn and recognize hundreds of words from this large-scale dataset.

In lip reading and in speaker detection, we demonstrate results that exceed the current state-of-the-art on public benchmark datasets.

1. Introduction

Lip-reading, the ability to understand speech using only visual information, is a very attractive skill. It has clear applications in speech transcription for cases where audio is not available, such as for archival silent films or (less ethically) off-mike exchanges between politicians or celebrities (the visual equivalent of open-mike mistakes). It is also complementary to the audio understanding of speech, and indeed can adversely affect perception if audio and lip motion are not consistent (as evidenced by the McGurk and MacDonald, 1976 effect). For such reasons, lip-reading has been the subject of a vast research effort over the last few decades.

Our objective in this work is a scalable approach to large lexicon *speaker independent* lip-reading. Furthermore, we aim to recognize words from *continuous speech*, where words are not segmented, and there may be co-articulation of the lips from preceding and subsequent words. Achieving this goal enables a form of 'word spotting' in (no-audio) video streams.

In lip-reading there is a fundamental limitation on performance due to *homophones*. These are sets of words that sound different, but involve identical movements of the speaker's lips. Thus they cannot be distinguished using visual information alone. For example, in English the phonemes '*p*', '*b*' and '*m*' are visually identical, and consequently the words *mark, park* and *bark*, are homophones (as are *pat, bat* and *mat*) and so cannot be distinguished by lip-reading. This problem has been well studied and there are lists of ambiguous phonemes and words available (Goldschen et al., 1996; Lucey et al., 2004). It is worth noting that the converse problem also applies: for example '*m*' and '*n*' are easily confused in audio, but are visually distinct. We take account of such homophone ambiguity in assessing the performance of our methods.

Apart from this limitation, lip-reading is a challenging problem in any case due to intra-class variations (such as accents, speed of speaking, mumbling), and adversarial imaging conditions (such as poor lighting, strong shadows, motion, resolution, foreshortening, etc.).

In this paper we investigate using Convolutional Neural Networks (CNNs) for directly recognizing individual *words* from a sequence of lip movements. Our reason for considering CNNs, rather than the more usual Recurrent Neural Networks that are used for sequence modelling, is their ability to learn to classify images on their content given only image supervision at the class level, *i.e.* without having to provide stronger supervisory information such as bounding boxes or pixel-wise segmentation. This ability is evident from the results of the ImageNet classification challenge (Russakovsky et al., 2015).

We take advantage of this ability to recognize temporal signals in an image time series. In particular we consider one second sequences of lip movements of continuous speech and learn to recognize words within the sequence given only class level supervision, but do not require stronger *temporal* supervision such as specifying the start and end of the word. Clearly, *spatial* registration of the mouth is an important element to consider in the design of the networks. Typically, the imaged head will move in the video, either due to actual movement of the head or due to camera motion. One approach would be to tightly register the

* Corresponding author.
E-mail address: joon@robots.ox.ac.uk (J.S. Chung).

https://doi.org/10.1016/j.cviu.2018.02.001
Received 29 April 2017; Received in revised form 24 December 2017; Accepted 1 February 2018
Available online 06 February 2018
1077-3142/ © 2018 Elsevier Inc. All rights reserved.

mouth region (including lips, teeth and tongue, that all contribute to word recognition), but another is to develop networks that are tolerant to some degree of motion jitter. We take the latter approach, and do not enforce tight registration.

We make contributions in three areas: first, we build a pipeline for automated large scale data collection, including visual and temporal alignment. With this we are able to obtain training data for hundreds of distinct words, thousands of instances for each word, and over a thousand speakers (Section 2); second, we develop a two-stream convolutional neural network *SyncNet* that learns a joint embedding between the sound and the mouth motions using cross-modal self-supervision. We apply this network to the tasks of audio-to-video synchronisation and active speaker detection (Section 3); third, we develop and compare a number of network architectures for classifying multi-frame time series of lips (Section 4). In speaker detection and lip reading, our results exceed the state-of-the-art on public datasets, Columbia (Chakravarty and Tuytelaars, 2016) and OuluVS2 (Anina et al., 2015).

As discussed in the related work below, in three aspects: (i) speaker independence, (ii) learning from continuous speech, and (iii) lexicon (vocabulary) size, we go far beyond the current state of the art. We also exceed the state of the art in terms of performance, as is also shown in Section 5 by comparisons on the standard OuluVS2 benchmark dataset (Anina et al., 2015).

1.1. Related work

Research on lip reading (*a.k.a.* visual speech recognition) has a long history. A thorough survey of shallow (*i.e.* not deep learning) methods is given in the recent review (Zhou et al., 2014b), and will not repeated in detail here. Many of the existing works in this field have followed similar pipelines which first extract spatio-temporal features around the lips (either motion-based, geometric-feature based or both), and then align these features with respect to a canonical template. For example, Pei et al. (2013), which holds state-of-the-art on many datasets, extracts the patch trajectory as a spatiao-temporal feature, and then aligns these features to reference motion patterns.

A number of recent papers have used deep learning methods to tackle problems related to lip reading. Koller et al. (2015) train an image classifier CNN to discriminate *visemes* (mouth shapes, visual equivalent of *phonemes*) on a sign language dataset where the signers mouth words. Similar CNN methods have been performed by Noda et al. (2014) to predict *phonemes* in spoken Japanese. In the context of word recognition, Tamura et al. (2015) has used deep bottleneck features (DBF) to encode *shallow* input features such as LDA and GIF (Ukai et al., 2012). Similarly Petridis and Pantic (2016) uses DBF to encode the image for every frame, and trains a LSTM classifier to generate a word-level classification.

One of the major obstacle to progress in this field has been the lack of suitable datasets (Zhou et al., 2014b). Table 1 gives a summary of existing datasets. The amount of available data is far from sufficient to train scalable and representative models that will be able to generalise beyond the controlled environments and the very limited domains (*e.g.* digits and the alphabet).

Word classification with large lexicons has not been attempted in lip reading, but Jaderberg et al. (2014) has tackled a similar problem in the context of text spotting. Their work shows that it is feasible to train a general and scalable word recognition model for a large pre-defined dictionary, as a multi-class classification problem. We take a similar approach.

Of relevance to the architectures and methods developed in this paper are CNNs for action recognition that learn from multiple-frame image sequences such as Ji et al. (2013); Karpathy et al. (2014) and Tran et al. (2015), particularly the ways in which they capture spatio-temporal information in the image sequence using temporal pooling layers and 3D convolutional filters.

2. Building the dataset

This section describes our multi-stage pipeline for automatically collecting and processing a very large-scale visual speech recognition dataset, starting from British television programmes. Using this pipeline we have been able to extract 1000s of hours of spoken text covering an extensive vocabulary of 1000s of different words, with over 1M word instances, and over 1000 different speakers.

The key ideas are to: (i) obtain a temporal alignment of the spoken audio with a text transcription (broadcast as subtitles with the programme). This in turn provides the time alignment between the visual face sequence and the words spoken; (ii) obtain a spatio-temporal alignment of the lower face for the frames corresponding to the word sequence; and, (iii) determine that the face is speaking the words (*i.e.* that the words are not being spoken by another person in the shot). The pipeline is summarised in Fig. 2 and the individual stages are discussed in detail in the following paragraphs.

Stage 1. Selecting programme types. We require programmes that have a changing set of talking heads, so choose news and current affairs, rather than dramas with a fixed cast. Table 2 lists the programmes. There is a significant variation of format across the programmes – from the regular news where a single speaker is talking directly at the camera, to panel debate where the speakers look at each other and often shifts their attention. There are a few people who appear repeatedly in the videos (*e.g.* news presenter in BBC News or the host in the others), but the large majority of participants change every episode (Fig. 1).

Stage 2. Subtitle processing and alignment. We require the alignment between the audio and the subtitle in order to get a time-stamp for every word that is being spoken in the videos. The BBC transmits subtitles as bitmaps rather than text, therefore subtitle text is extracted from the broadcast video using standard OCR methods (Buehler et al., 2009; Everingham et al., 2006). The subtitles are not time-aligned, and also not verbatim as they are generated live. The Penn Phonetics Lab Forced Aligner (Hermansky, 1990; Yuan and Liberman, 2008) (based on the open-source HTK toolbox Woodland et al., 1995) is used to force-align the subtitle to the audio signal. The

Table 1
Existing lip reading datasets. **I** for **I**solated (one word, letter or digit per recording); **C** for **C**ontinuous recording. The reported performance is on speaker-independent experiments. (* For GRID Cooke et al., 2006, there are 51 classes in total, but the first word in a phrase is restricted to 4, the second word 4, etc. 8.5 is the average number of possible classes at each position in the phrase.)

Name	Env.	Output	I/C	# class	# subj.	Best perf.
AVICAR (Lee et al., 2004)	In-car	Digits	C	10	100	37.9% (Fu et al., 2008)
AVLetter (Matthews et al., 2002)	Lab	Alphabet	I	26	10	43.5% (Zhao et al., 2009)
CUAVE (Patterson et al., 2002)	Lab	Digits	I	10	36	83.0% (Papandreou et al., 2009)
GRID (Cooke et al., 2006)	Lab	Words	C	8.5*	34	79.6% (Wand and Koutn, 2016)
OuluVS1 (Zhao et al., 2009)	Lab	Phrases	I	10	20	89.7% (Pei et al., 2013)
OuluVS2 (Anina et al., 2015)	Lab	Phrases	I	10	52	73.5% (Zhou et al., 2014a)
LRW	TV	Words	C	500	1000+	–

Table 2
Video statistics. The yield is the proportion of useful face appearance relative to the total length of video. A useful face appearance is one that appears continuously for at least 5 s, with the face being that of the speaker.

Channel	Series name	Description	# vid.	Length	Yield
BBC 1 HD	News at 1	Regular news	1242	30 min	39.9%
BBC 1 HD	News at 6	Regular news	1254	30 min	33.9%
BBC 1 HD	News at 10	Regular news	1301	30 min	32.9%
BBC 1 HD	Breakfast	Regular news	395	Varied	39.2%
BBC 1 HD	Newsnight	Current affairs debate	734	35 min	40.0%
BBC 2 HD	World News	Regular news	376	30 min	31.9%
BBC 2 HD	Question Time	Current affairs debate	353	60 min	48.8%

aligner uses the Viterbi algorithm to compute the maximum likelihood alignment between the audio (modelled by PLP features Rubin et al., 2013) and the text. This method of obtaining the alignment has significant performance benefits over regular speech recognition methods that do not use prior knowledge of what is being said. The alignment result, however, is not perfect due to: (1) the method often misses words that are spoken too quickly; (2) the subtitles are not verbatim; (3) the acoustic model is only trained to recognise American English. The noisy labels are filtered by double-checking against the commercial IBM Watson Speech to Text service. In this case, the only remaining label noise is where an interview is dubbed in the news, which is rare.

Stage 3. Shot boundary detection, face detection, and tracking. The shot boundaries are determined to find the within-shot frames for which face tracking is to be run. This is done by comparing color histograms across consecutive frames (Lienhart, 2001). The HOG-based face detection method of King (2009) is performed on every frame of the video (Fig. 4 left). As with most face detection methods, this results in many false positives and some missed detections. In a similar manner to Everingham et al. (2006), all face detections of the same person are grouped across frames using a KLT tracker (Tomasi and Kanade, 1992) (Fig. 4 middle). If the track overlaps with face detections on the majority of frames, it is assumed to be correctly tracking the face.

Stage 4. Facial landmark detection and speaker identification. Facial landmarks are needed to determine the mouth position for speaker/ non-speaker classification. They are determined in every frame of the face track using the method of Kazemi and Sullivan (2014) (Fig. 4 right). The landmarks are used to determine the mouth region, and to map it to a canonical position as input to the two-stream network described in Section 3 that is used to determine who is speaking in the video, and reject the clip if the face is not-speaking in sync. It is important to determine whether the face shown is actually speaking or not. For example, there may be a reaction shot or voice-over.

Stage 5. Compiling the training and test data. The training, validation and test sets are disjoint in time. The dates of videos corresponding to each set is shown in Table 3. Note that we leave a week's gap between the test set and the rest in case any news footage is repeated. The lexicon is obtained by selecting the 500 most frequently occurring words between 5 and 10 characters in length (Fig. 6 gives the word duration statistics). This word length is chosen such that the speech duration does not exceed the fixed one-second bracket that is used in the recognition architecture, whilst shorter words are not included because there are too many ambiguities due to homophones (*e.g.* 'bad', 'bat', 'pat', 'mat', etc. are all visually identical), and sentence-level context would be needed to disambiguate these.

These 500 words occur at least 800 times in the training set, and at least 40 times in each of the validation and test sets. For each of the occurrences, the one-second clip is taken, and the face is cropped with the mouth centered using the registration found in Stage 4. The words are *not* isolated, as is the case in other lip-reading datasets; as a result, there may be co-articulation of the lips from preceding and subsequent words. The *test* set is manually checked for errors.

3. Learning a synchronization network for lip motion and audio

The ability to identify who is speaking is crucial in building the dataset described in Section 2, and has many applications beyond this task.

This section describes the representations and network architectures for a synchronization network (SyncNet), which ingests 0.2 s audio and video clips, and generates a joint embedding between the inputs. The audio-to-video embedding is then used to identify the active speaker and to correct the lip-sync error.

No explicit annotations (*e.g.* word labels, or the precise time offset) are used to train this network – we only assume that in the majority of television videos, the audio and the video are *usually* synced, and we use cross-modal self-supervision to learn the embedding.

The model consists of two asymmetric streams for audio and video, each of which is described below.

3.1. Loss function

The training objective is that the output of the audio and the video

Fig. 1. A sample of speakers in our dataset.

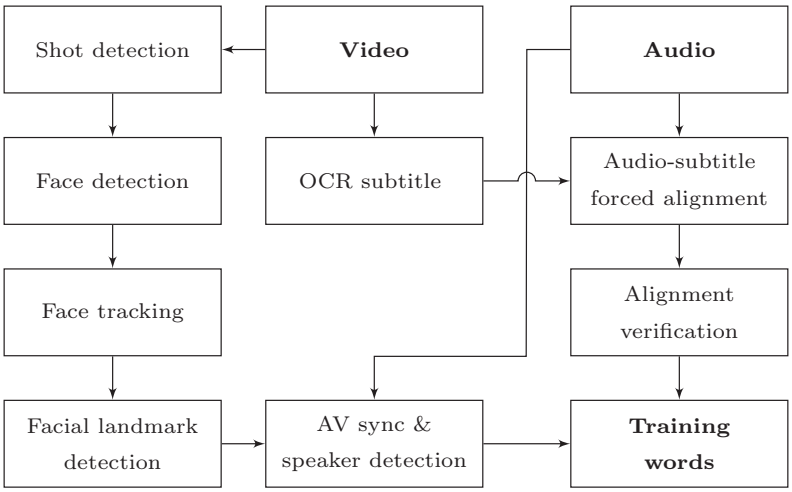

Fig. 2. Pipeline to generate the text and visually aligned dataset.

Table 3
Dataset statistics.

Set	Dates	# class	#/class
Train	01/01/2010–28/02/2015	500	800+
Val	01/03/2015–25/07/2015	500	50
Test	01/08/2015–31/03/2016	500	50

networks are similar for *genuine* pairs, and different for *false* pairs. Specifically, the Euclidean distance between the network outputs is minimised or maximised. We propose to use the contrastive loss (Eq. (1)), originally proposed for training Siamese networks (Chopra et al., 2005). v and a are fc_7 vectors for the video and the audio streams, respectively. $y \in [0, 1]$ is the binary similarity between the audio and the video inputs.

$$E = \frac{1}{2N}\sum_{n=1}^{N}(y_n)d_n^2 + (1-y_n)\max(margin - d_n, 0)^2 \quad (1)$$

$$d_n = \|v_n - a_n\|_2 \quad (2)$$

An alternative to this would be to approach the problem as one of classification (binary classification of on-sync and off-sync, *or* multi-class between the different offset bins using synthetic data), however we were unable to achieve convergence using this method.

3.2. Training

The training procedure is inspired by the Siamese network (Chopra et al., 2005), however our network is different in that it consists of non-identical streams, two independent sets of parameters and inputs from two different domains. The network weights are learnt using stochastic gradient descent with momentum. The parameters for both streams of the network are learnt simultaneously.

3.3. Audio stream

The input audio data is MFCC values. This is a representation of the short-term power spectrum of a sound on a non-linear mel scale of frequency. 13 mel frequency bands are used at each time step. The features are computed at a sampling rate of 100 Hz, giving 20 time steps for a 0.2 s input signal.

3.3.1. Representation

The audio is encoded as a heatmap image representing MFCC values for each time step and each mel frequency band (see Fig. 8). The top and bottom three rows of the image are reflected to reduce boundary effects. Previous work Geras et al. has also attempted to train image-style CNN for similar inputs.

3.3.2. Architecture

We use a convolutional neural network inspired by those designed for image recognition. Our layer architecture (Fig. 7) is based on VGG-M (Chatfield et al., 2014), but with modified filter sizes to ingest the inputs of unusual dimensions of 13 × 20 (13 in the frequency domain, and 20 in the time domain).

3.4. Visual stream

3.4.1. Representation

The input format to the visual network is a sequence of mouth regions as grayscale images, as shown in Fig. 8. The input dimensions are 111 × 111 × 5 (W × H × T) for 5 frames, which corresponds to 0.2 s at 25 Hz.

3.4.2. Architecture

The visual stream is based on the VGG-M network, but the *conv1* filter size has been modified to ingest the 5-channel input instead of the usual 3.

3.5. Applications

The problems of AV synchronisation and active speaker detection are closely related in that the correspondence between the video and the accompanying audio must be established.

3.5.1. Audio-to-video synchronisation

To find the time offset between the audio and the video, we use a sliding-window approach. For each sample, the distance is computed between one 5-frame video feature and every audio feature in the ± 1 s range. The correct offset is found where this distance is at a minimum. Since not every 0.2 s sample contains discriminative information (*e.g.*, the person might be taking a breath), the distance for every offset value is averaged across the video clip. Typical distances against offset plots are shown in Fig. 9.

3.5.2. Active speaker detection

We test our method using the dataset (Fig. 10) and the evaluation protocol of (Chakravarty and Tuytelaars, 2016). The objective is to determine who the speaker is in a multi-subject scene.

The dataset contains 6 speakers, of which 1 is used for development and 5 (Bell, Bollinger, Lieberman, Long, Sick) for testing. A score

Table 4
F_1-scores on the Columbia speaker detection dataset. The results of (Chakravarty and Tuytelaars, 2016) have been digitised from Fig. 3(b) of their paper, and are accurate to around ± 0.5%.

Method	Chakravarty and Tuytelaars (2016)		Ours	
Window	10	100	10	100
Bell	82.9%	90.3%	93.7%	**100%**
Bollinger	65.8%	69.0%	83.4%	**100%**
Lieberman	73.6%	82.4%	86.8%	**100%**
Long	86.9%	96.0%	97.7%	**99.8%**
Sick	81.8%	89.3%	86.1%	**99.8%**

threshold is set using the annotations on the remaining speaker (Abbas), at the point where the ROC curve intersects the diagonal (the equal error rate).

We report the F_1-scores in Table 4. The scores for each test sample are averaged over a 10-frame or 100-frame window. The performance is almost perfect for the 100-frame window. The disadvantage of increasing the size of the averaging window is that the method cannot detect examples in which the person speaks for a very short period; though this is not a problem for this dataset.

4. Models for lip reading

The task for the network is to predict which words are being spoken, given a video of a talking face. The input format to the network is a sequence of mouth regions, as shown in Fig. 5. Previous attempts at visual speech recognition have relied on very precise localisation of the facial landmarks (the mouth in particular); our aim is learn from more noisy data, and tolerate some localisation irregularities both in position and in time.

4.1. Architectures

We cast the problem as one of multi-way classification, and so base our architecture on ones designed for image classification (Chatfield et al., 2014; Krizhevsky et al., 2012; Simonyan and Zisserman, 2015). In particular, we build on the VGG-M model (Chatfield et al., 2014) since this has a good classification performance, but is much faster to train and experiment on than deeper models, such as VGG-16 (Simonyan and Zisserman, 2015). We develop and compare models that differ principally in how they 'ingest' the T input frames (where here T = 25 for a 1 s interval). These variations take inspiration from previous work on human action classification (Ji et al., 2013; Karpathy et al., 2014; Ng et al., 2015; Tran et al., 2015). Apart from these differences, the architectures share the configuration of VGG-M, and this allows us to directly compare the performance across different input designs. These configurations are closely related to the visual stream of SyncNet (Section 3).

We next describe the five architectures, summarised in Fig. 11, followed by a discussion of their differences. Their performance is compared in Section 5. The numbers in the names refer to the number of temporal frames ingested by each tower, and the EF, MT, LF and LSTM indicates where the fusion occurs.

4.1.1. Early Fusion (EF-25)

The network ingests a 25-channel image, where each of the channels encode an individual frame in *greyscale*. The layer structure for the subsequent layers is identical to that of the regular VGG-M network. This method is related to the Early Fusion model in Karpathy et al. (2014), which takes *colour* images and uses a T × 3-channel convolutional filter at *conv1*. We did experiment with 25 × 3-channel colour input, but found that the increased number of parameters at *conv1* made training difficult due to overfitting (resulting in validation performance that is around 5% weaker; not quoted in Section 5).

4.1.2. Multiple Towers (MT-1)

There are T= 25 towers with common *conv1* layers (with shared weights), each of which takes one input frame. The activations from the towers are concatenated channel-wise after *pool1*, producing an output activation with 1200 channels. The subsequent 1 × 1 convolution is performed to reduce this dimension, to keep the number of parameters at *conv2* at a managable level. The rest of the network is the same as the regular VGG-M.

4.1.3. Multiple Towers (MT-5)

There are 21 towers with common *conv1* layers, each of which takes a 5-frame window, moving one 1-frame at a time. The subsequent layers are configured in the same way as **MT-1**.

4.1.4. Late Fusion (LF-5)

Like **MT-5**, the 21 towers each take 5-frame windows, with a stride of 1. However, each tower in this variant has common *conv1* to *fc6* layers with shared weights, after which the activations are concatenated. The subsequent layer structure is the same as **EF-25, MT-1** and **MT-5**.

4.1.5. Long Short-Term Memory (LSTM-5)

Each convolutional tower shares the layer configuration of the **LF-5** model. The two-layer LSTM ingests the visual features (*fc6* activations) of the 5-frame sliding window, moving 1-frame at a time, and returns the classification result at the end of the sequence.

4.1.6. Discussion

The early fusion architecture **EF-25** shares similarities with previous work on human action recognition using CNNs (Ji et al., 2013; Karpathy et al., 2014; Ng et al., 2015) in that registration between frames is assumed. The models perform time-domain operations beginning from the first layer to precisely capture local motion direction and speed (Karpathy et al., 2014). For these methods to capture useful information, good registration of details between frames is critical. However, we are not imposing strict registration, and in any case it goes slightly against the signal (lip motion and mouth region deformation) that we are trying to capture.

In contrast, the **MT-1** model delays all time-domain registrations (and operations) until after the first set of convolutional and pooling layers. This gives tolerance against minor registration errors (the receptive field size at *conv2* is 11 pixels). Note, the common *conv1* layers of the multiple towers ensures that the same filter weights are used for all frames, whereas in the early fusion architecture **EF-25** it is possible to learn different weights for each frame.

The **MT-5** model shares similarities with both **EF-25** and **MT-1** models – the 5-frame input to each tower allows the network to learn some local motion information, but are more tolerant to movements than the **EF-25** model over the whole time period.

The **LF-5** model also shares many characteristics of **MT-5**, but delays time-domain operations until after all of the convolutional layers, except within the 5 neighbouring frames between which the movement would be negligible.

Likewise, the **LSTM-5** delays time-domains operations, and in addition, this model benefits from the ability to accept sequences of variable lengths, unlike the other models.

One other design choice is the size of the input images. This was chosen as 111 × 111 pixels, which is smaller than that typically used in image classification networks. The reason is that the size of the cropped mouth images are rarely larger than 111 × 111 pixels, and this smaller choice means that smaller filters can be used at *conv1* than those used in VGG-M without sacrificing receptive fields, but at a gain in avoiding unnecessary parameters being learnt.

4.2. Training

4.2.1. Data augmentation

Data augmentation often helps to improve validation performance by reducing overfitting in CNN image classification tasks (Krizhevsky et al., 2012). We apply the augmentation techniques used on the ImageNet classification task by Simonyan and Zisserman (2015) and Krizhevsky et al. (2012) (*e.g.* random cropping, flipping, colour shift), with a consistent transformation applied to all frames of a single clip. To further augment the training data, we make random shifts in time by up to 0.2 s, which improves the *top-1* validation error by 3.5% compared to the standard ImageNet augmentation methods. It was not feasible to scale in the time-domain as this results in artifacts being shown due to the relatively low video refresh rate of 25 fps.

4.2.2. Details

Our implementation is based on the MATLAB toolbox MatConvNet (Vedaldi et al., 2014) and Caffe (Jia, 2013). The network is trained using SGD with momentum 0.9 and batch normalisation (Ioffe and Szegedy, 2015), but without dropout. The training was stopped after 20 epochs, or when the validation error did not improve for 3 epochs, whichever is sooner. The learning rate of 10^{-2}–10^{-4} was used, decreasing on log scale.

5. Experiments

In this section we evaluate and compare the several proposed architectures, and discuss the challenges arising from the visual ambiguities between words. We then compare to the state of the art on a public benchmark.

5.1. Comparison of architectures

5.1.1. Evaluation protocol

The models are evaluated on the independent test set (Section 2). We report *top-1* and *top-10* accuracies , as well as recall against rank curves. Here, the *'Recall@K' is the proportion of times that the correct class is found in the top-K predictions for the word. The experiments were performed under two different conditions: 'continuous*' where the input sequences also contain co-articulation from the neighbouring words within the one-second window and 'isolated' where the words are segmented according to the forced alignment output, and thus can last less than one-second.

5.1.2. Results

The results are shown in Table 5. The experimental results show that the registration-tolerant models gives a modest improvement over **EF-25**, and the performance improvement is likely to be more significant where the tracking quality is less ideal. Having 5 frames as input seems to achieve a good balance for registration, in that it is able to compute useful temporal derivatives (**MT-5** is better than **MT-1** where no temporal derivatives are computed) which requires local (in time) registration, but does not require the global registration of **EF-25** (which is inferior to both MT models). The **LSTM-5** shows stronger performance compared to the CNN-based models. For all models, the performance is slightly better under the 'isolated' conditions since there are fewer ambiguities due to co-articulation.

Table 5
Word classification accuracy. **Left:** On the LRW dataset for the different architectures. **Right:** On OuluVS2 (short phrases, frontal view). **Con. (continuous):** the input sequences also contain co-articulation from the neighbouring words within the one-second window; **Iso. (isolated):** the words are segmented according to the forced alignment output, and thus can last less than one-second.

Net	LRW (Con.)		LRW (Iso.)			OuluVS2
	R@1	R@10	R@1	R@10		R@1
EF-25	57.0%	88.8%	62.5%	92.6%	Zhou et al. (2014a)	73.5%
MT-1	61.1%	90.4%	64.2%	94.2%	Saitoh et al. (2016)	85.6%
MT-5	66.8%	94.6%	69.0%	95.6%	MT-1	93.2%
LF-5	65.4%	93.3%	68.2%	94.8%	MT-5	93.2%
LSTM-5	66.0%	94.3%	71.5%	96.4%	LSTM-5	94.1%

Table 6
Most frequently confused word pairs for the 'continuous' experiment. The numbers refer to class confusions.

0.32	BENEFITS	BENEFIT	0.24	HAPPEN	HAPPENED
0.31	QUESTIONS	QUESTION	0.24	FORCE	FORCES
0.31	REPORT	REPORTS	0.23	HAPPENED	HAPPEN
0.31	BORDER	IMPORTANT	0.23	SERIOUS	SERIES
0.31	AMERICA	AMERICAN	0.23	TROOPS	GROUPS
0.29	GROUND	AROUND	0.22	QUESTION	QUESTIONS
0.28	RUSSIAN	RUSSIA	0.21	PROBLEM	PROBABLY
0.28	FIGHT	FIGHTING	0.21	WANTED	WANTS
0.26	FAMILY	FAMILIES	0.21	RUSSIA	RUSSIAN
0.26	AMERICAN	AMERICA	0.20	TAKEN	TAKING
0.26	BENEFIT	BENEFITS	0.20	PROBLEM	PROBLEMS
0.25	ELECTIONS	ELECTION	0.20	MISSING	MEETING
0.24	WANTS	WANTED	0.20	PARTIES	PARTY

The *top-10* accuracy for the best models are over 95%, despite the relatively modest *top-1* figure of around 70%. This is a result of ambiguities in lip reading, which we will discuss next.

5.2. Analysis of confusions

Here, we examine the classification results, in particular, the scenarios in which the network fails to correctly classify the spoken word. Table 6 shows the most common confusions between words in the test set for the 'continuous' experiment. This is generated by taking the largest off-diagonal values in the word confusion matrix. This result confirms our prior knowledge about the challenges in visual speech recognition – almost all of the top confusions are either (i) a plural of the original word (*e.g.* 'report' and 'reports') which is ambiguous because one word is a subset of the other, and the words are not isolated so this can be due to co-articulation; or (ii) a known homophone visual ambiguity (explained in Section 1) where the words cannot be distinguished using visual information alone (*e.g.* 'billion' and 'million', 'worse' and 'worst'). Such errors are phonetically understandable. For example, some of the most common confusions, *e.g.* 'groups' which is phonetically (G R UW P S) and 'troops' (T R UW P S) , 'ground' (G R AW N D) and 'around' (ER AW N D), actually share most of the phonemes.

Apart from these difficulties, the failure cases are typically for extreme samples. For example, due to strong international accents, or poor quality/low bandwidth location reports and Skype interviews, where there are motion compression artifacts or frames dropped from the transmission.

5.3. Comparison to state of the art

It is worth noting that the *top-1* classification accuracy of over 70%, shown in Table 5, is comparable to that of many of the recent works (Fu et al., 2008; Ngiam et al., 2011; Petridis and Pantic, 2016) performed on lexicon sizes that are orders of magnitude smaller (Table 1).

5.3.1. OuluVS2

We evaluate our method on the OuluVS2 dataset (Anina et al., 2015). The dataset consists of 52 subjects uttering 10 phrases (*e.g.* 'thank you', 'hello', etc.), and has been widely used in previous works. Here, we assess on a speaker-independent experiment, where some of the subjects are reserved for testing.

To apply our method on this dataset, we pre-train the convolutional layers on the BBC data, and re-train the fully-connected layers from scratch. Training from scratch on OuluVS2 underperforms as the size of

Fig. 3. Subtitles on BBC TV. **Left**: 'Question Time', **Right**: 'BBC News at One'.

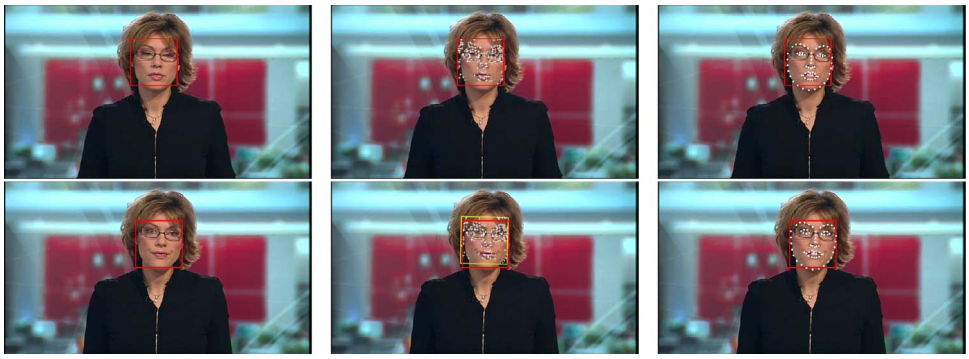

Fig. 4. Left: Face detections; **Middle:** KLT features and the tracked bounding box (in yellow); **Right:** Facial landmarks. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Fig. 5. One-second clips that contain the word '*about*'. Top: male speaker, bottom: female speaker.

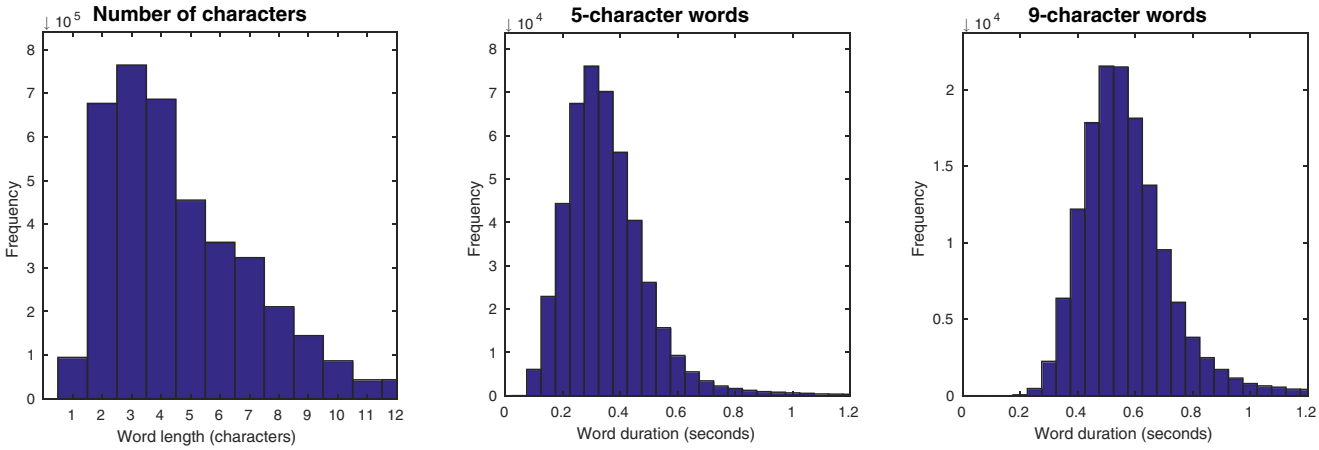

Fig. 6. Word statistics. Regardless of the actual duration of the word, we take a 1 s clip for training and test.

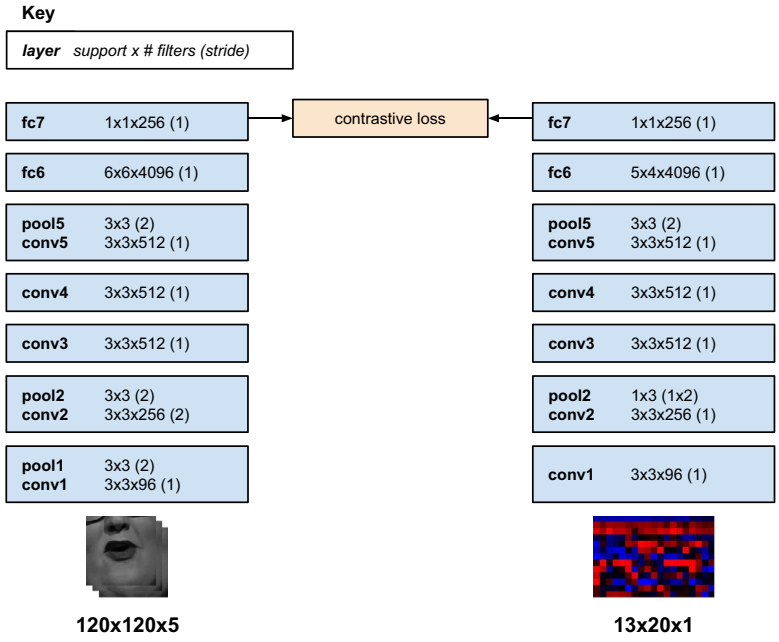

Fig. 7. SyncNet architecture. Both streams are trained simultaenously.

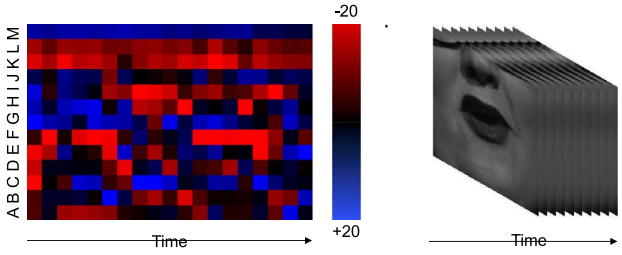

Fig. 8. Input representations. Left: temporal representations as heatmaps for audio. The 13 rows (A–M) in the audio image encode each of the 13 MFCC features representing powers at different frequency bins. **Right:** Grayscale images of the mouth area.

this dataset is insufficient to train a deep network. For all models apart from LSTM-5, we simply repeat the first and the last frames to fill the 1 s clip if the phrase is shorter than 25 frames. If the clip is longer, we take a random crop.

As can be seen in Table 5 the method achieves a strong performance, and sets the new state-of-the-art. Note that, without retraining the convolutional part of the network, we achieve these strong results on videos that are very different to ours in terms of lighting, background, camera perspective, etc. (Fig. 12), which shows that the model generalises well across different formats.

6. Summary and extensions

We have shown that CNN and LSTM architectures can be used to classify temporal lip motion sequences of words with excellent results. We also demonstrated a recognition performance that exceeds the state of the art on a standard public benchmark dataset, OuluVS2.

Extensions could include lip reading of profile views, and varying the architecture (in terms of depth, 3D CNNs etc) to improve performance – there is already evidence that there are benefits of using deeper architectures (Stafylakis and Tzimiropoulos, 2017) on our released dataset. It is worth noting that recent papers have combined CNNs with sequence models in order to recognize sentences rather than individual words (Assael et al., 2017; Chung et al., 2017).

The dataset is available for download at http://www.robots.ox.ac.uk/~vgg/data/lip_reading/ and the trained SyncNet is available at http://www.robots.ox.ac.uk/~vgg/software/lipsync/.

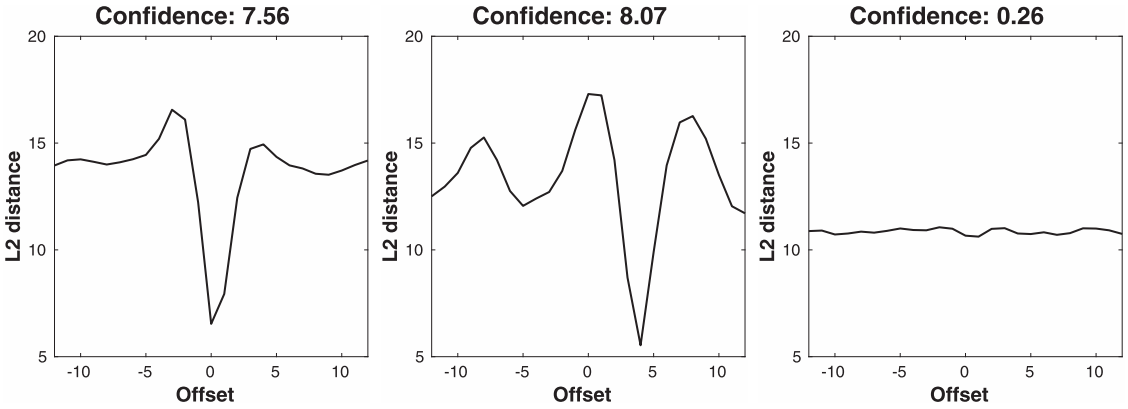

Fig. 9. Mean distance between the audio and the video features for different offset values, averaged over a clip. The actual offset lies at the trough. The three example clips shown here are for different scenarios. **Left:** synchronised AV data; **Middle:** the audio leads the video by 4 frames; **Right:** the audio and the video are uncorrelated.

Fig. 10. Still images from the Columbia dataset (Chakravarty and Tuytelaars, 2016).

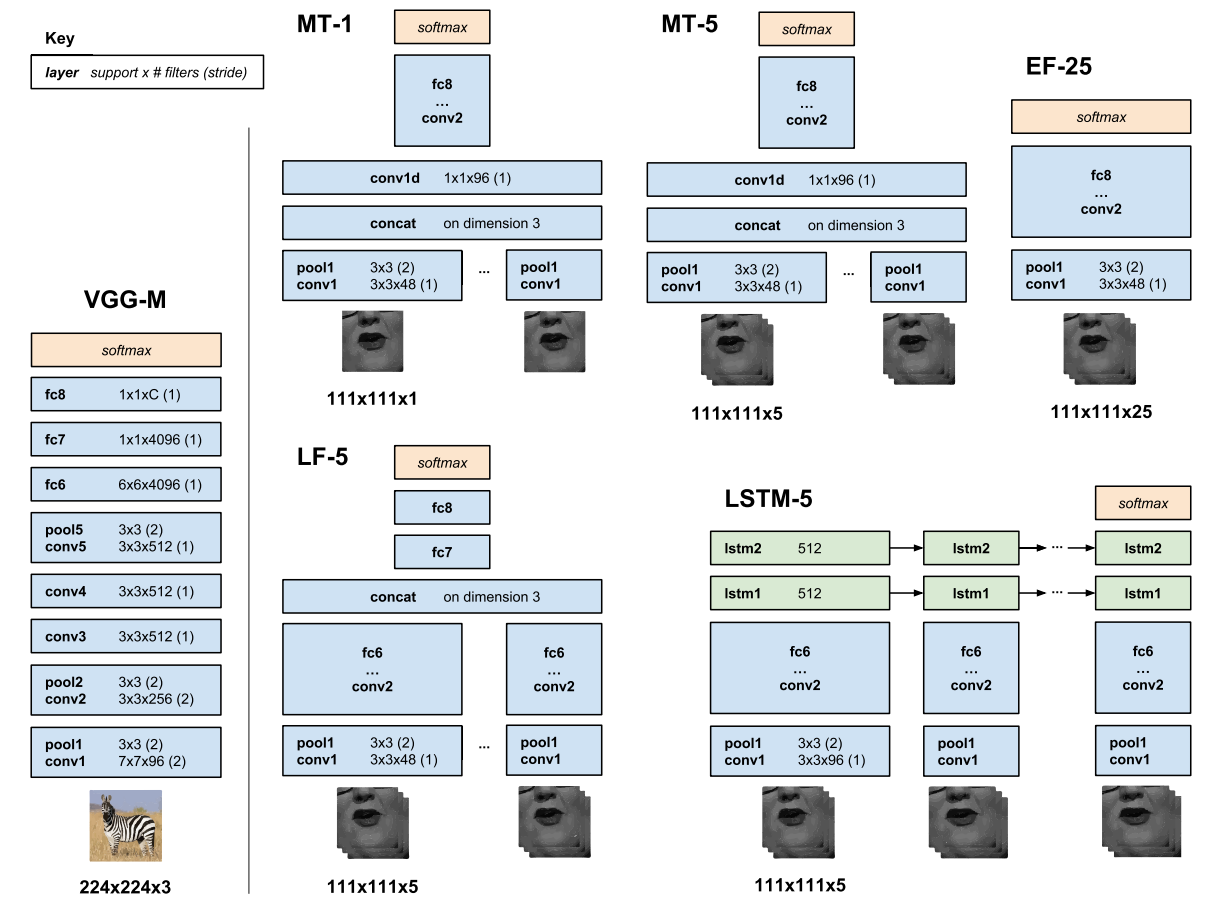

Fig. 11. Left: VGG-M architecture that is used as a base. **Right:** Network architectures for lip reading.

Fig. 12. Original video frames for 'hello' on OuluVS. Compare this to the our original input frames in Fig. 3.

Acknowledgements

Funding for this research is provided by the EPSRC Programme Grant Seebibyte EP/M013774/1. We are very grateful to Rob Cooper and Matt Haynes at BBC Research for help in obtaining the dataset.

References

Anina, I., Zhou, Z., Zhao, G., Pietikäinen, M., 2015. Ouluvs2: a multi-view audiovisual database for non-rigid mouth motion analysis. Automatic Face and Gesture Recognition (FG), 2015 11th IEEE International Conference and Workshops on, Vol. 1. IEEE, pp. 1–5.

Assael, Y. M., Shillingford, B., Whiteson, S., de Freitas, N., 2016. Lipnet: Sentence-level lipreading. Arxiv:1611.01599.

Buehler, P., Everingham, M., Zisserman, A., 2009. Learning sign language by watching TV (using weakly aligned subtitles). Proc. CVPR.

Chakravarty, P., Tuytelaars, T., 2016. Cross-modal supervision for learning active speaker detection in video. Arxiv:1603.08907.

Chatfield, K., Simonyan, K., Vedaldi, A., Zisserman, A., 2014. Return of the devil in the details: delving deep into convolutional nets. Proc. BMVC.

Chopra, S., Hadsell, R., LeCun, Y., 2005. Learning a similarity metric discriminatively,

with application to face verification. Proc. CVPR. 1. IEEE, pp. 539–546.
Chung, J.S., Senior, A., Vinyals, O., Zisserman, A., 2017. Lip reading sentences in the wild. IEEE Conference on Computer Vision and Pattern Recognition.
Cooke, M., Barker, J., Cunningham, S., Shao, X., 2006. An audio-visual corpus for speech perception and automatic speech recognition. J. Acoust. Soc. Am. 120 (5), 2421–2424.
Everingham, M., Sivic, J., Zisserman, A., 2006. "Hello! My name is··· Buffy" – automatic naming of characters in TV video. Proc. BMVC.
Fu, Y., Yan, S., Huang, T.S., 2008. Classification and feature extraction by simplexization. Inf. Forensics Secur. IEEE Trans. 3, 91–100.
Geras, K. J., Mohamed, A. r., Caruana, R., Urban, G., Wang, S., Aslan, O., Philipose, M., Richardson, M., Sutton, C., 2016. Compressing lstms into cnns. Arxiv:1511.06433.
Goldschen, A.J., Garcia, O.N., Petajan, E.D., 1996. Rationale for Phoneme-viseme Mapping and Feature Selection in Visual Speech Recognition. Speechreading by Humans and Machines. Springer, pp. 505–515.
Hermansky, H., 1990. Perceptual linear predictive (PLP) analysis of speech. J. Acoust. Soc. Am. 87 (4), 1738–1752.
Ioffe, S., Szegedy, C., 2015. Batch normalization: accelerating deep network training by reducing internal covariate shift. Arxiv:1502.03167.
Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A., 2014. Synthetic data and artificial neural networks for natural scene text recognition. Workshop on Deep Learning, NIPS.
Ji, S., Xu, W., Yang, M., Yu, K., 2013. 3D convolutional neural networks for human action recognition. IEEE PAMI 35 (1), 221–231.
Jia, Y., 2013. Caffe: An Open Source Convolutional Architecture for Fast Feature Embedding. http://caffe.berkeleyvision.org/.
Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L., 2014. Large-scale video classification with convolutional neural networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1725–1732.
Kazemi, V., Sullivan, J., 2014. One millisecond face alignment with an ensemble of regression trees. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1867–1874.
King, D.E., 2009. Dlib-ml: a machine learning toolkit. J. Mach. Learn. Res. 10, 1755–1758.
Koller, O., Ney, H., Bowden, R., 2015. Deep learning of mouth shapes for sign language. Proceedings of the IEEE International Conference on Computer Vision Workshops. pp. 85–91.
Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. ImageNet classification with deep convolutional neural networks. NIPS. pp. 1106–1114.
Lee, B., Hasegawa-Johnson, M., Goudeseune, C., Kamdar, S., Borys, S., Liu, M., Huang, T.S., 2004. Avicar: audio-visual speech corpus in a car environment. INTERSPEECH. Citeseer.
Lienhart, R., 2001. Reliable transition detection in videos: a survey and practitioner's guide. Int. J. Image Graph.
Lucey, P., Martin, T., Sridharan, S., 2004. Confusability of phonemes grouped according to their viseme classes in noisy environments. Proc. of Australian Int. Conf. on Speech Science & Tech. pp. 265–270.
Matthews, I., Cootes, T.F., Bangham, J.A., Cox, S., Harvey, R., 2002. Extraction of visual features for lipreading, pattern analysis and machine intelligence. IEEE Trans. 24 (2), 198–213.
McGurk, H., MacDonald, J., 1976. Hearing lips and seeing voices. Nature 264, 746–748.
Ng, J.Y.-H., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., Toderici, G., 2015. Beyond short snippets: deep networks for video classification. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4694–4702.
Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., Ng, A.Y., 2011. Multimodal deep learning. Proceedings of the 28th international conference on machine learning (ICML-11). pp. 689–696.
Noda, K., Yamaguchi, Y., Nakadai, K., Okuno, H.G., Ogata, T., 2014. Lipreading using convolutional neural network. INTERSPEECH. pp. 1149–1153.
Papandreou, G., Katsamanis, A., Pitsikalis, V., Maragos, P., 2009. Adaptive multimodal fusion by uncertainty compensation with application to audiovisual speech recognition, audio, speech, and language processing. IEEE Trans. 17 (3), 423–435.
Patterson, E.K., Gurbuz, S., Tufekci, Z., Gowdy, J.N., 2002. Cuave: a new audio-visual database for multimodal human-computer interface research. Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on. 2. IEEE, pp. II-2017–II-2020.
Pei, Y., Kim, T.K., Zha, H., 2013. Unsupervised random forest manifold alignment for lipreading. Proceedings of the IEEE International Conference on Computer Vision. pp. 129–136.
Petridis, S., Pantic, M., 2016. Deep complementary bottleneck features for visual speech recognition. ICASSP. pp. 2304–2308.
Rubin, S., Berthouzoz, F., Mysore, G.J., Li, W., Agrawala, M., 2013. Content-based tools for editing audio stories. Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology. ACM, pp. 113–122.
Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, S., Karpathy, A., Khosla, A., Bernstein, M., Berg, A., Li, F., 2015. Imagenet large scale visual recognition challenge. IJCV.
Saitoh, T., Zhou, Z., Zhao, G., Pietikäinen, M., 2016. Concatenated frame image based cnn for visual speech recognition. Asian Conference on Computer Vision. Springer, pp. 277–289.
Simonyan, K., Zisserman, A., 2015. Very deep convolutional networks for large-scale image recognition. International Conference on Learning Representations.
Stafylakis, T., Tzimiropoulos, G., 2017. Combining residual networks with lstms for lipreading. Interspeech.
Tamura, S., Ninomiya, H., Kitaoka, N., Osuga, S., Iribe, Y., Takeda, K., Hayamizu, S., 2015. Audio-visual speech recognition using deep bottleneck features and high-performance lipreading. 2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA). IEEE, pp. 575–582.
Tomasi, C., Kanade, T., 1992. Selecting and tracking features for image sequence analysis. Rob. Autom.
Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M., 2015. Learning spatiotemporal features with 3d convolutional networks.
Ukai, N., Seko, T., Tamura, S., Hayamizu, S., 2012. Gif-lr: Ga-based informative feature for lipreading. Signal & Information Processing Association Annual Summit and Conference (APSIPA ASC), 2012 Asia-Pacific. IEEE, pp. 1–4.
Vedaldi, A., Mahendran, S., Tsogkas, S., Maji, S., Girshick, R., Kannala, J., Rahtu, E., Kokkinos, I., Blaschko, M.B., Weiss, D., Taskar, B., Simonyan, K., Saphra, N., Mohamed, S., 2014. Understanding objects in detail with fine-grained attributes. Proc. CVPR.
Wand, M., Koutn, J., et al., 2016. Lipreading with long short-term memory. 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, pp. 6115–6119.
Woodland, P.C., Leggetter, C., Odell, J., Valtchev, V., Young, S.J., 1995. The 1994 htk large vocabulary speech recognition system. Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on. 1. IEEE, pp. 73–76.
Yuan, J., Liberman, M., 2008. Speaker identification on the scotus corpus. J. Acoust. Soc. Am. 123 (5), 3878.
Zhao, G., Barnard, M., Pietikäinen, M., 2009. Lipreading with local spatiotemporal descriptors, multimedia. IEEE Trans. 11 (7), 1254–1265.
Zhou, Z., Hong, X., Zhao, G., Pietikäinen, M., 2014. A compact representation of visual speech data using latent variables. IEEE Trans. Pattern Anal. Mach. Intell. 36 (1), 181–186.
Zhou, Z., Zhao, G., Hong, X., Pietikäinen, M., 2014. A review of recent advances in visual speech decoding. Image Vis. Comput. 32 (9), 590–605.