Lip Reading in the Wild

Joon Son Chung and Andrew Zisserman

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

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 two novel contributions: first, we develop a pipeline for fully automated large-scale 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 CNN architectures that are able to effectively learn and recognize hundreds of words from this large-scale dataset.

We also demonstrate a recognition performance that exceeds the state of the art on a standard public benchmark dataset.

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 and 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 [23] effect). For such reasons, lip-reading has been the subject of a vast research effort over the last few decades. It has also been the subject of excellent comedy sketches, e.g. Seinfeld "The Lip Reader", and its ambiguity and challenge can be exploited to replace/overdub actual speech, *e.g.* in the YouTube channel "Bad Lip Reading".

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.

In lip-reading there is a fundamental limitation on performance due to *homophemes.* 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 homophemes (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 [8, 21]. It is worth noting that the converse problem also applies: for example '*m*' and '*n*' are easily confused as audio, but are visually distinct. We take account of such homopheme 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.).

The usual approach to inference for temporal sequences is to employ sequence models such as Hidden Markov Models or Recurrent Neural Networks (e.g. LSTMs). For lip-reading such models can be employed for predicting individual characters or phonemes. In contrast, we investigate using Convolutional Neural Networks (CNNs) for directly recognizing individual *words* from a sequence of lip movements.

Clearly, visual registration 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 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 two areas: first, we develop 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 CNN architectures for classifying multi-frame time series of lips. In particular we propose and compare different input and temporal fusion architectures, and discuss their pros and cons (Section 3). We analyse the performance and ambiguity of the resulting classifications in Section 4.

As discussed in the related work below, in these three aspects: speaker independence, learning from continuous speech, and 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 4 by comparisons on the standard OuluVS benchmark dataset [1, 43].

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 [45], 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.* [28], 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.* [16] 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 [25] to predict *phonemes* in spoken Japanese. In the context of word recognition, [33] has used deep bottleneck features (DBF) to encode *shallow* input features such as LDA and GIF [36]. Similarly [29] 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 [45]. 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).

Name	**Env.**	**Output**	**I/C**	**# class**	**# subj.**	**Best perf.**
AVICAR [19]	In-car	Digits	C	10	100	37.9% [7]
AVLetter [22]	Lab	Alphabet	I	26	10	43.5% [43]
CUAVE [27]	Lab	Digits	I	10	36	83.0% [26]
GRID [4]	Lab	Words	C	8.5^*	34	79.6% [39]
OuluVS1 [43]	Lab	Phrases	I	10	20	89.7% [28]
OuluVS2 [1]	Lab	Phrases	I	10	52	73.5% [44]
OuluVS2 [1]	Lab	Digits	C	10	52	-
BBC TV	TV	Words	C	333/500	1000+	-

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 [4], 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.)

Word classification with large lexicons has not been attempted in lip reading, but [11] 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 ConvNets for action recognition that learn from multiple-frame image sequences such as [12, 13, 35], 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 programs. 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.

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

The key ideas are to: (i) obtain a temporal alignment of the spoken audio with a text transcription (broadcast as subtitles with the program). 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 Figure 2 and the individual stages are discussed in detail in the following paragraphs.

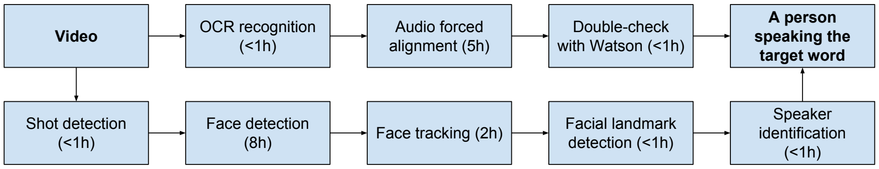
Video
OCR recognition (<1h)
Audio forced alignment (5h)
Double-check with Watson (<1h)
A person speaking the target word
Shot detection (<1h)
Face detection (8h)
Face tracking (2h)
Facial landmark detection (<1h)
Speaker identification (<1h)

Fig. 2. Pipeline to generate the text and visually aligned dataset. Timings are for a one-hour video.

Stage 1. Selecting program types. We require programs 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 programs. There is a significant variation of format across the programs – 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 (Figure 1).

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

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 seconds, with the face being that of the speaker.

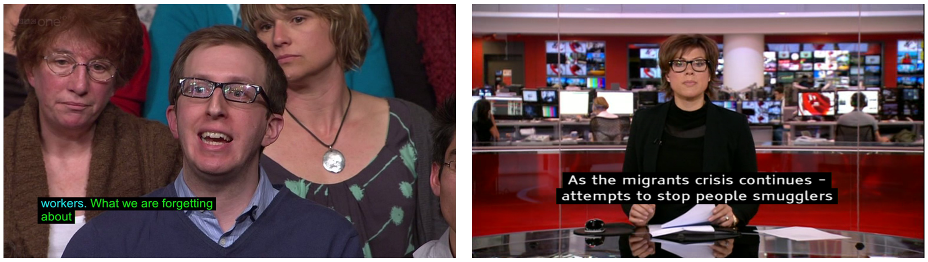
workers. What we are forgetting
about
As the migrants crisis continues –
attempts to stop people smugglers

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

Stage 2. Subtitle processing and alignment. We require the alignment between the audio and the subtitle in order to get a timestamp 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 [2, 6]. The subtitles are not time-aligned, and also not verbatim as they are generated live. The Penn Phonetics Lab Forced Aligner [9, 41] (based on the open-source HTK toolbox [40]) is used to force-align the subtitle to the audio signal. The aligner uses the Viterbi algorithm to compute the maximum likelihood alignment between the audio (modelled by PLP features [30]) 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 [20]. The HOG-based face detection method of [15] is performed on every frame of the video (Figure 4 left). As with most face detection methods, this results in many false positives and some missed detections. In a similar manner to [6], all face detections of the same person are grouped across frames using a KLT tracker [34] (Figure 4 middle). If the track overlaps with face detections on the majority of frames, it is assumed to be correctly tracking the face.

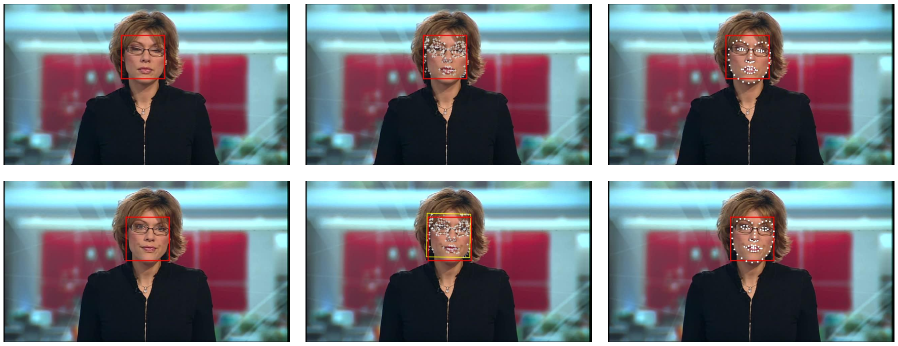

Fig. 4. Left: Face detections; **Middle:** KLT features and the tracked bounding box (in yellow); **Right:** Facial landmarks.

Stage 4. Facial landmark detection and speaker identification. Facial landmarks are needed to (1) determine the mouth position for cropping; and (2) for speaker/ non-speaker classification. Facial landmarks are determined in every frame of the face track using the method of [14] (Figure 4 right). To identify who is speaking, we assume that a person speaking will have lip movements that fall within a particular frequency range that is different to that arising from tracking noise. The 'openness' of the mouth is measured on every frame using the distance between the top and the bottom lip, normalised with respect to the size of the face in the video. For a speaking face, the openness signal contains the actual lip motion as well as the tracking noise, whereas for a non-speaking face (*e.g.* reaction shot, etc.), the only observed movement is the noise. A simple method of taking the Fourier transform of the mouth 'openness' temporal signal is performed to separate the lip movements that fall into different frequencies bins. A linear SVM classifier is trained on the frequency spectrum to make the distinction between a face that is speaker from a face that is not.

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 (Figure 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 homophemes (*e.g.* 'bad', 'bat', 'pat', 'mat', etc. are all visually identical), and sentence-level context would be needed to disambiguate these.

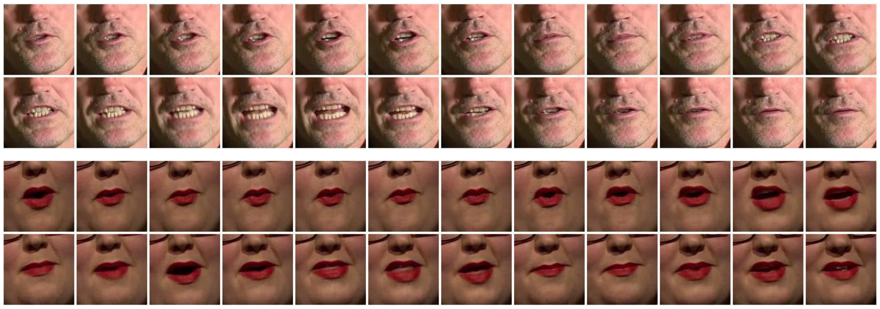

Fig. 5. One-second clips that contain the word '*about*'. Top: male speaker, bottom: female speaker.

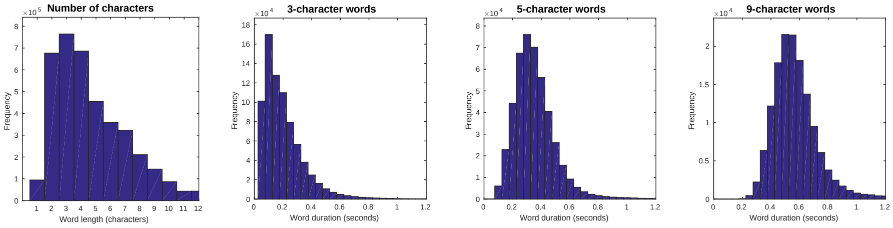

Fig. 6. Word statistics. Regardless of the actual duration of the word, we take a 1-second clip for training and test.

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.

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

Table 3. Dataset statistics.

3 Network Architecture and Training

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 Figure 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 from more noisy data, and tolerate some localisation irregularities both in position and in time.

3.1 Architecture

We cast the problem as one of multi-way classification, and so base our architecture on ones designed for image classification [3, 18, 32]. In particular, we build on the VGG-M model [3] since this has a good classification performance, but is much faster to train and experiment on than deeper models, such as VGG-16 [32]. We develop and compare four models that differ principally in how they 'ingest' the T input frames (where here T= 25 for a 1 second interval). These variations take inspiration from previous work on human action classification [12, 13, 35, 42]. 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.

We next describe the four architectures, summarized in Figure 7, followed by a discussion of their differences. Their performance is compared in Section 4.

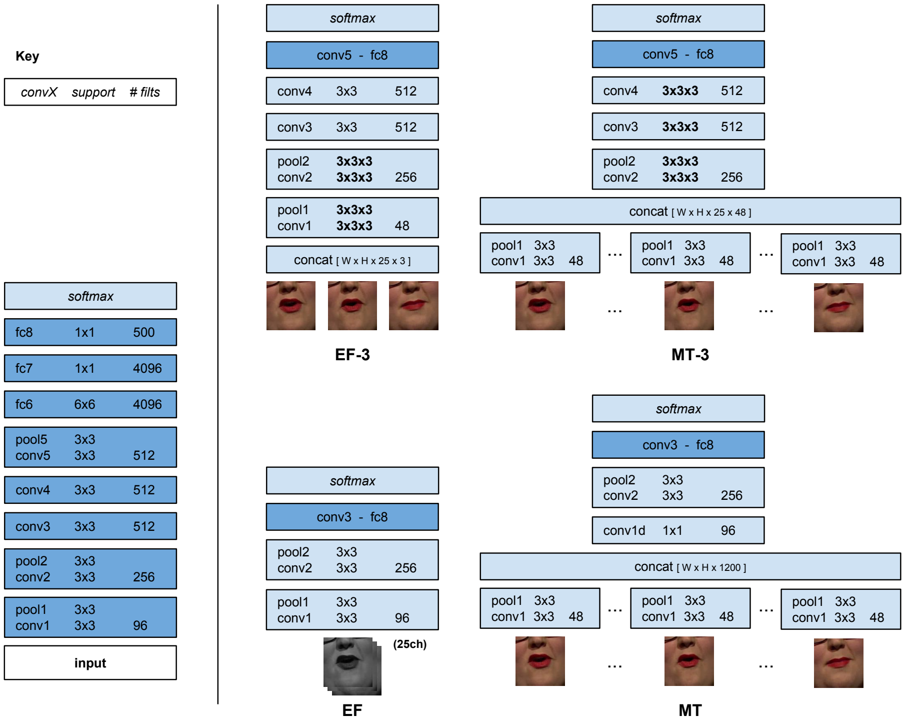

Fig. 7. CNN architectures. **Left:** VGG-M architecture that is used as a base. **Right: EF-3:** 3D Convolution with Early Fusion; **MT-3:** 3D Convolution with Multiple Towers; **EF:** Early fusion; **MT:** Multiple Towers.

3D Convolution with Early Fusion (EF-3). This architecture is inspired by the work of [12] on human action recognition using 3D ConvNets. The general structure resembles that of an ordinary CNN used for image classification, but instead of taking H×W×3 input, it takes H×W×T×3 input. The convolutional and pooling filters operate and move along all three dimensions.

3D Convolution with Multiple Towers (MT-3). The model shares its basic design principles with the architecture of **EF-3**, however there is no explicit time-domain connectivity between frames before *conv2*. There are T= 25 towers with common *conv1* layers (with shared weights), each of which takes an input frame. Here, the activations at *pool1* are concatenated along a new dimension, and the 3D convolutions from *conv2* are performed in the same manner as [12] and **EF-3**.

Early Fusion (EF). The network ingests a T-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 [13], 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 4).

Multiple Towers (MT). There are T= 25 towers with common *conv1* layers (with shared weights), each of which takes an 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.

Discussion. There are two basic divisions of the architectures: between early fusion and multiple towers, and between 2D and 3D convolutions. We will discuss these in turn. The early fusion architectures, **EF-3** and **EF**, share similarities with previous work on human action recognition using ConvNets [12, 13, 42] in the way that they assume registration between frames. The models perform time-domain operations beginning from the first layer to precisely capture local motion direction and speed [13]. 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 multiple towers architectures, **MT-3** and **MT**, both delay 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** it is possible to learn different weights for each frame. The experimental results show that these registration-tolerant models gives a modest improvement over their counterparts, and the performance improvement is likely to be more significant where the tracking quality is less ideal.

The reason for including 3D convolutions (the architectures **EF-3** and **MT-3**) is that intuitively a 3D convolution (that can have small spatial and *temporal* kernel size) should be able to match well a spatio-temporal feature, such as a particular lip shape over a particular sub-sequence. In contrast the 2D convolutions extend over the entire temporal range, and thus might be thought to waste parameters or require redundancy when trying to respond to such spatio-temporal features. Despite this intuition, the experimental results show that the 2D convolutions are superior to their 3D counterparts.

One other design choice is the size of the input images. This was chosen as 112×112 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 112×112 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.

3.2 Training

Data augmentation. Data augmentation often helps to improve validation performance by reducing overfitting in ConvNet image classification tasks [18]. We apply the augmentation techniques used on the ImageNet classification task by [18, 32] (*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 seconds, 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 25fps.

Details. Our implementation is based on the MATLAB toolbox MatConvNet [37] and trained on a NVIDIA Titan X GPU with 12GB memory. The network is trained using SGD with momentum 0.9 and batch normalisation [10], 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} to 10^{-4} was used, decreasing on log scale.

4 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.

4.1 Comparison of architectures

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. We also report the character-level edit distance [17], which is the minimum number of character-level operations required to convert the predicted string to the ground truth. This metric imposes smaller penalties where the predicted string is similar to the ground truth (*e.g.* 'concerned' and 'concerns' have an edit distance of 2) and larger penalties where the words are very different (*e.g.* 'concerned' and 'company' have an edit distance of 6).

Results. As discussed in Section 3.1, the **MT-3** and **MT** variants have the advantage of being more tolerant to registration errors compared to their early fusion counterparts. The results in Table 4 and Figure 8 confirm this, where we see a modest (3.2% on average for *top-1*) but consistent improvement in performance across the experiments. The performance of 3D ConvNets fall short of the 2D architectures by an average of around 14%.

The recall curves in Figure 8 rise sharply for all models at low-K; the *top-10* figure for the **EF** and **MT** models being over 85%, despite the modest *top-1* figure of around 60%. This is a result of ambiguities in lip reading, which we will discuss next.

Net	**500-class**			**333-class**	
	Top-1	**Top-10**	**ED**	**Top-1**	**Top-10**
EF-3	43.9%	81.0%	3.13	55.7%	87.9%
MT-3	46.2%	82.4%	2.97	56.8%	88.7%
EF	57.0%	88.8%	2.32	63.2%	91.8%
MT	**61.1%**	**90.4%**	**2.06**	**65.4%**	**92.3%**

	OuluVS1	**OuluVS2**
	Top-1	**Top-1**
[29]	81.8%	-
[44]	85.6%	73.5%
[28]	89.7%	-
MT	**91.4%**	**93.2%**

Table 4. Word classification results. **Left:** On the BBC data for the four different architectures. **ED** is the edit distance. **Right:** On OuluVS1 and OuluVS2 (short phrases, frontal view).

4.2 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 5 shows the most common confusions between words in the test set. 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 in our dataset so this can be due to co-articulation; or (ii) a known homopheme visual ambiguity (explained in Section 1) where the words cannot be distinguished using visual information alone (*e.g.* 'billion' and 'million', 'worse' and 'worst').

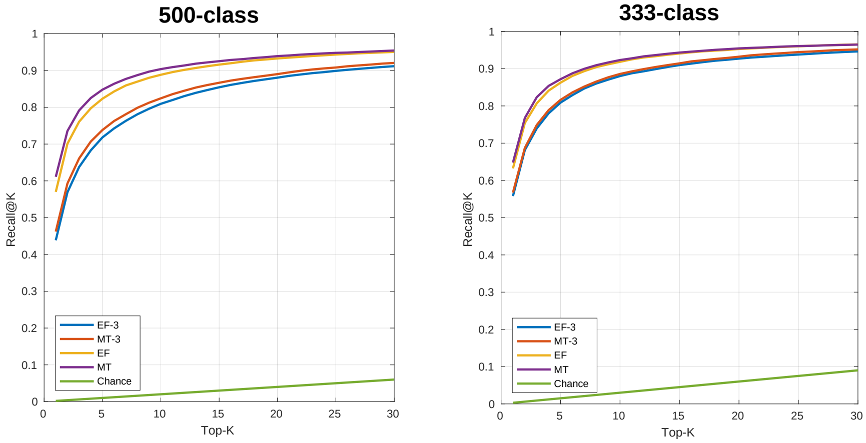

Fig. 8. Recall vs Rank curves for the word classification.

500-class

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

333-class

0.30	BORDER	IMPORTANT
0.29	PROBABLY	PROBLEM
0.27	TAKING	TAKEN
0.25	PERSONAL	PERSON
0.23	CLAIMS	GAMES
0.22	AROUND	GROUND
0.21	TONIGHT	NIGHT
0.21	PROBLEM	PROBABLY
0.19	SEVERAL	SEVEN
0.19	CHALLENGE	CHANGE
0.18	PRICES	PERSON
0.18	WARNING	MORNING
0.18	CAPITAL	HAPPENED
0.18	OTHER	ANOTHER
0.17	AHEAD	AGAIN
0.16	WORKERS	WORDS
0.16	MEDIA	MEETING
0.16	UNITED	NIGHT
0.16	NEVER	SEVEN
0.15	WORLD	WORDS

Table 5. Most frequently confused word pairs.

Therefore, we generate a second test set where we eliminate these two types of known ambiguities. We first group the words according to the aforementioned criteria (*e.g.* 'billion', 'million' and 'millions' would form a single group), and keep only the most frequently occuring word in the training set for each group, eliminating the ambiguous words for that group. This process produces a new balanced test set containing a lexicon of 333 word-classes.

The network is finetuned on this new vocabulary for 1 epoch, before being re-evaluated. The results reported in Table 4 and Figure 8 that are labelled '333-word' are evaluated on this vocabulary. The *top-10* performance increases from 90.4% (for the 500 word-class test set) to 92.3% (for the 333 word-class test set). This is an improvement, but still not perfect. The reason is that even excluding the known homopheme and plural ambiguities does not remove all confusion. Table 5 shows the common errors remaining, and these are phonetically understandable. For example, some of the most common confusions, e.g. 'claims' which is phonetically (K L EY M Z) and 'games' (G EY M Z) , 'probably' (P R AA B AH B L IY) and 'problem' (P R AA B L AH M), 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.

4.3 Visualisation of salient mouth shapes

Our aim here is to visualize the frames of the temporal sequence that are most discriminative for the word. Simonyan *et al.* [31] have shown that it is possible to infer the localization of visual objects in an image as a saliency map for a network trained to classify images. We adapt this method to find the salient temporal information in a time-sequence.

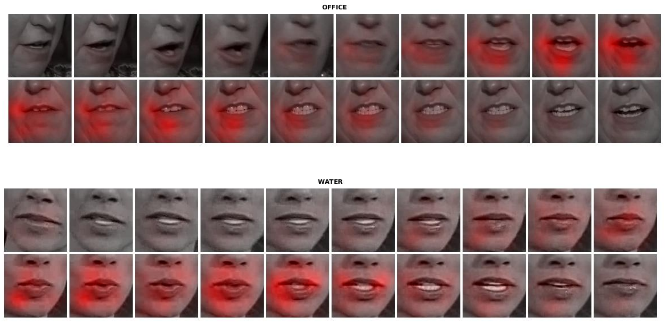

Fig. 9. Salient visual features of sequences 'office' and 'water' are highlighted in red.

The method approximates the relation between the class score S and the input image I (represented as a vector) as $S(I) = w^T I + b$. The vector w is the same size as the input image, and the magnitude of its elements signify the influence of the corresponding elements of the image on the class score. Hence the magnitude of w determines a saliency map on the image. The vector w can be obtained as $w = \frac{\partial S_c}{\partial I}\Big|_{I_0}$ and this derivative is obtained by back-prop from the class score $S_0(I_0)$ to the image.

The resulting salient regions are shown in Figure 9. For example, the most distinctive mouth shape for 'office' (AO F AH S) is the 'AH' with the mouth open and 'F' with the top teeth biting the bottom lip.

4.4 Comparison to state of the art

It is worth noting that the *top-1* classification accuracy of 65%, shown in Table 4, is comparable to that of many of the recent works [7, 24, 29] performed on lexicon sizes that are orders of magnitude smaller (Table 1).

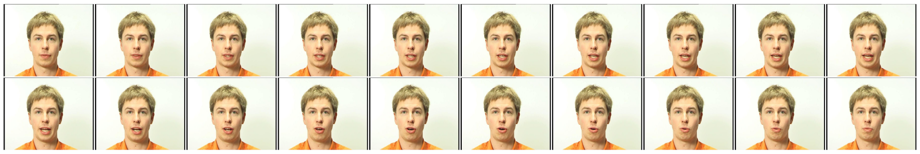

Fig. 10. Original video frames for ‘hello’ on OuluVS. Compare this to the our original input frames in Figure 3.

OuluVS. We evaluate our method on the OuluVS datasets. OuluVS1 [43] consists of 20 subjects uttering 10 phrases (*e.g.* ‘thank you’, ‘hello’, etc.), and has been widely used in previous works. OuluVS2 [1] (short phrases) consists of 52 subjects uttering the same phrases as [43]. 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 model on the BBC data, and fine-tune the fully-connected layers. Training from scratch on OuluVS underperforms as the size of this dataset is insufficient to train a deep network. If the phrase is shorter than 25 frames, we simply repeat the first and the last frames to fill the 1-second clip. If the clip is longer, we take a random crop.

As can be seen in Table 4 our 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. (Figure 10), which shows that our model generalises well across different formats.

5 Summary and extensions

We have shown that CNN architectures can be used to classify temporal sequences with excellent results. On the 333-word test set, we achieve *top-1* accuracy of 65.4%, which exceeds state-of-the-art [7, 43] on multiple datasets [19, 22] that have lexicon sizes that are orders of magnitude smaller, and a *top-10* accuracy of 92.3%. We also demonstrate a recognition performance that exceeds the state of the art on a standard public benchmark dataset, OuluVS.

Next steps include extending to lip reading of profile views, and combining the CNNs pre-trained using this approach with LSTMs trained with a language model [5, 38], in order to recognize sentences rather than individual words. Of course, the visual only speech recognition method developed here can also be combined with audio only speech recognition to both their benefits.

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.

Bibliography

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

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

[3] Chatfield, K., Simonyan, K., Vedaldi, A., Zisserman, A.: Return of the devil in the details: Delving deep into convolutional nets. In: Proc. BMVC. (2014)

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

[5] Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., Darrell, T.: Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2625–2634 (2015)

[6] Everingham, M., Sivic, J., Zisserman, A.: "Hello! My name is... Buffy" – automatic naming of characters in TV video. In: Proc. BMVC. (2006)

[7] Fu, Y., Yan, S., Huang, T.S.: Classification and feature extraction by simplexization. Information Forensics and Security, IEEE Transactions on 3(1), 91–100 (2008)

[8] Goldschen, A.J., Garcia, O.N., Petajan, E.D.: Rationale for phoneme-viseme mapping and feature selection in visual speech recognition. In: Speechreading by Humans and Machines, pp. 505–515. Springer (1996)

[9] Hermansky, H.: Perceptual linear predictive (plp) analysis of speech. the Journal of the Acoustical Society of America 87(4), 1738–1752 (1990)

[10] Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015)

[11] Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Synthetic data and artificial neural networks for natural scene text recognition. In: Workshop on Deep Learning, NIPS (2014)

[12] Ji, S., Xu, W., Yang, M., Yu, K.: 3D convolutional neural networks for human action recognition. IEEE PAMI 35(1), 221–231 (2013)

[13] Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 1725–1732 (2014)

[14] Kazemi, V., Sullivan, J.: One millisecond face alignment with an ensemble of regression trees. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1867–1874 (2014)

[15] King, D.E.: Dlib-ml: A machine learning toolkit. The Journal of Machine Learning Research 10, 1755–1758 (2009)

[16] Koller, O., Ney, H., Bowden, R.: Deep learning of mouth shapes for sign language. In: Proceedings of the IEEE International Conference on Computer Vision Workshops. pp. 85–91 (2015)

[17] Kondrak, G.: A new algorithm for the alignment of phonetic sequences. In: Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference. pp. 288–295. Association for Computational Linguistics (2000)

[18] Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: NIPS. pp. 1106–1114 (2012)

[19] Lee, B., Hasegawa-Johnson, M., Goudeseune, C., Kamdar, S., Borys, S., Liu, M., Huang, T.S.: Avicar: audio-visual speech corpus in a car environment. In: INTERSPEECH. Citeseer (2004)

[20] Lienhart, R.: Reliable transition detection in videos: A survey and practitioner's guide. International Journal of Image and Graphics (Aug 2001)

[21] Lucey, P., Martin, T., Sridharan, S.: Confusability of phonemes grouped according to their viseme classes in noisy environments. In: Proc. of Australian Int. Conf. on Speech Science & Tech. pp. 265–270 (2004)

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

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

[24] Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., Ng, A.Y.: Multimodal deep learning. In: Proceedings of the 28th international conference on machine learning (ICML-11). pp. 689–696 (2011)

[25] Noda, K., Yamaguchi, Y., Nakadai, K., Okuno, H.G., Ogata, T.: Lipreading using convolutional neural network. In: INTERSPEECH. pp. 1149–1153 (2014)

[26] Papandreou, G., Katsamanis, A., Pitsikalis, V., Maragos, P.: 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)

[27] Patterson, E.K., Gurbuz, S., Tufekci, Z., Gowdy, J.N.: Cuave: A new audio-visual database for multimodal human-computer interface research. In: Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on. vol. 2, pp. II–2017. IEEE (2002)

[28] Pei, Y., Kim, T.K., Zha, H.: Unsupervised random forest manifold alignment for lipreading. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 129–136 (2013)

[29] Petridis, S., Pantic, M.: Deep complementary bottleneck features for visual speech recognition. ICASSP pp. 2304–2308 (2016)

[30] Rubin, S., Berthouzoz, F., Mysore, G.J., Li, W., Agrawala, M.: Content-based tools for editing audio stories. In: Proceedings of the 26th annual ACM symposium on User interface software and technology. pp. 113–122. ACM (2013)

[31] Simonyan, K., Vedaldi, A., Zisserman, A.: Deep inside convolutional networks: Visualising image classification models and saliency maps. In: Workshop at International Conference on Learning Representations (2014)

[32] Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations (2015)

[33] Tamura, S., Ninomiya, H., Kitaoka, N., Osuga, S., Iribe, Y., Takeda, K., Hayamizu, S.: Audio-visual speech recognition using deep bottleneck features and high-performance lipreading. In: 2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA). pp. 575–582. IEEE (2015)

[34] Tomasi, C., Kanade, T.: Selecting and tracking features for image sequence analysis. Robotics and Automation (1992)

[35] Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks (2015)

[36] Ukai, N., Seko, T., Tamura, S., Hayamizu, S.: Gif-lr: Ga-based informative feature for lipreading. In: Signal & Information Processing Association Annual Summit and Conference (APSIPA ASC), 2012 Asia-Pacific. pp. 1–4. IEEE (2012)

[37] Vedaldi, A., Lenc, K.: Matconvnet – convolutional neural networks for matlab. CoRR abs/1412.4564 (2014)

[38] Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: A neural image caption generator. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3156–3164 (2015)

[39] Wand, M., Koutník, J., Schmidhuber, J.: Lipreading with long short-term memory. arXiv preprint arXiv:1601.08188 (2016)

[40] Woodland, P.C., Leggetter, C., Odell, J., Valtchev, V., Young, S.J.: The 1994 htk large vocabulary speech recognition system. In: Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on. vol. 1, pp. 73–76. IEEE (1995)

[41] Yuan, J., Liberman, M.: Speaker identification on the scotus corpus. Journal of the Acoustical Society of America 123(5), 3878 (2008)

[42] Yue-Hei Ng, J., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., Toderici, G.: Beyond short snippets: Deep networks for video classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4694–4702 (2015)

[43] Zhao, G., Barnard, M., Pietikäinen, M.: Lipreading with local spatiotemporal descriptors. Multimedia, IEEE Transactions on 11(7), 1254–1265 (2009)

[44] Zhou, Z., Hong, X., Zhao, G., Pietikäinen, M.: A compact representation of visual speech data using latent variables. IEEE transactions on pattern analysis and machine intelligence 36(1), 1–1 (2014)

[45] Zhou, Z., Zhao, G., Hong, X., Pietikäinen, M.: A review of recent advances in visual speech decoding. Image and vision computing 32(9), 590–605 (2014)