Is automated conversion of video to text a reality?

Richard Bowden[a], Stephen Cox[b], Richard Harvey[b], Yuxuan Lan[b], Eng-Jon Ong[a] , Gari Owen[c] and Barry-John Theobald[b]

[a]University of Surrey, Guildford, GU2 7XH, UK.
[b]University of East Anglia, Norwich, NR4 7TJ, UK.
[c]Annywyn Solutions, Bromley, Kent, BR1 3DW, UK.

ABSTRACT

A recent trend in law enforcement has been the use of Forensic lip-readers. Criminal activities are often recorded on CCTV or other video gathering systems. Knowledge of what suspects are saying enriches the evidence gathered but lip-readers, by their own admission, are fallible so, based on long term studies of automated lip-reading, we are investigating the possibilities and limitations of applying this technique under realistic conditions. We have adopted a step-by-step approach and are developing a capability when prior video information is available for the suspect of interest. We use the terminology video-to-text (V2T) for this technique by analogy with speech-to-text (S2T) which also has applications in security and law-enforcement.

Keywords: Lip-reading, speech recognition, pattern recognition

1. INTRODUCTION

Much of the intelligence associated with the investigation of crime is based on what various people are saying. This ranges from gossip to conversations between those planning criminal or terrorist acts. It has always been common practice in the criminal community to be wary of being overheard and hence conversations often take place at randomly chosen locations such as street corners. The suspects are often recorded opportunistically by a variety of security CCTV networks and video cameras, but without audio. It would therefore be extremely useful under certain circumstances to extract audio from the video product. We refer to this as the conversion of video-to-text (V2T) by analogy of the more established technique of speech recognition: speech-to-text (S2T). Indeed, much of the philosophy and technology of V2T is derived from S2T, which has been established and evolving for about 50 years.

Human lip-readers have been used to interpret speech in video-product. However, there are few lip-readers available, transcription is often very slow and training and certification are not well developed. A full discussion of the performance of human lip-readers could occupy another paper but, in short, it is difficult to establish confidence intervals on human performance. There is therefore a desire for automated means of V2T conversion, where the process can be scaled for widespread use. As in the case of human lip-readers, the information for the conversion of video-to-text is derived from the movement of the lips of the speaker. It is possible that other information such as gestures could also be used to enhance the level of the information accessible but, for the time being we focus on lip-motions. A major difficulty of both human lip-reading and its machine counterpart is that similar lip gestures (sometimes called visemes) may be associated with different phonemes. Context is therefore all important. Ideally, we would like to be able to perform video-to-text conversion on all subjects, but this issue makes this extremely difficult. Accent and dialect can lead to multiple interpretations of the same viseme. Based on a well-established research base of fifteen years, we have adopted an engineering approach to demonstrate the technology of V2T conversion using the following approach*:

- We limit our attempts to the use of subjects where data has been captured already. This is by analogy of S2T where high performance can be obtained if training data for a subject is available. This is the typical approach taken for personal speech recognition systems now commercially available.

Further author information: (Send correspondence to RWH)
RWH.: E-mail: r.w.harvey@uea.ac.uk

*An explanation of an early version of our system can be found at `http://www.youtube.com/watch?v=Tu2vInqqHX8`.

- We limit our attempts to the recognition of pre-determined words (eg place names or digits).
- Our tests were initially performed under laboratory conditions with uniform lighting and a stable camera.

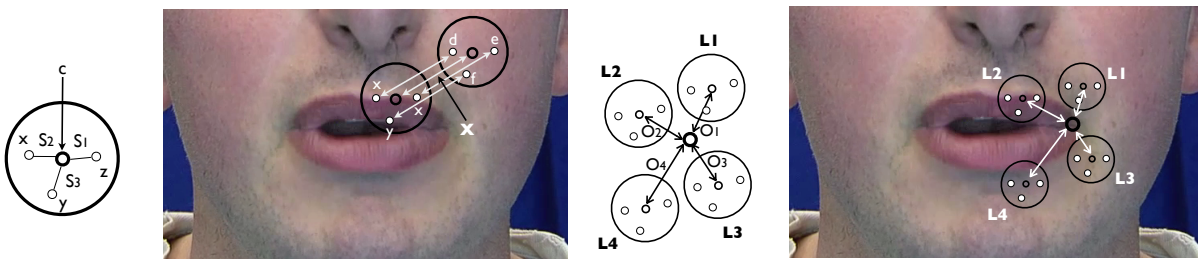
We have subsequently performed experiments under more challenging conditions of a realistic out of doors scenario, with a hand held (shaky) camera with varying pose. This approach is aimed to probe the boundaries of the technique and determine whether the technique has a practical utility.

2. TECHNICAL APPROACH

Computer lip-reading can be analysed as the production of various sub-systems involved with tracking, feature extraction, classification and language modelling which is how we divide this section.

2.1 Tracking

Face detection is a very well studied problem but what is needed for lip-reading is an agile tracking system that is able to track the talker's lips independent of pose, expression and utterance. We currently regard it as acceptable to have a short training phase to learn the motions of a particular talker. Our initial approach was to learn a person-specific Active Appearance Model (described in the next section) and to search exhaustively from frame-to-frame. This is tricky to operationalise since the lips are very agile and such trackers often fail. Our most recent approaches use a set of linear predictors (LPs) as described in Ref. 1.



(a) A single LP (b) A flock of LPs

Figure 1. LP tracker as described Ref. 1.

The basis of an LP is that a point with coordinates $\mathbf{c} = [c_x, c_y]^T$ in an image taken from a video sequence, frame n, moves an amount $\mathbf{t} = [t_x, t_y]^T$ to frame $n+1$, with T denoting matrix transpose. The assumption is that $\mathbf{t}$ is related to the measured change in intensity via

$$\mathbf{t} = \mathbf{H}\delta\mathbf{p} \tag{1}$$

where $\mathbf{H}$ is some learnt mapping between intensity differences and position, and

$$[\delta\mathbf{p}]_i = V_i^{(n+1)} - V_i^{(n)} \tag{2}$$

where $V_i^{(n)}$ is the i^{th} support pixel grey-value in frame n.

Each point, $\mathbf{c}$, has an associated pixel support region which is defined via a set of (x, y) offsets, $\mathbf{S}$. Each point that we wish to track is therefore represented by a four-tuple vector

$$L = \{\mathbf{c}, \mathbf{H}, \mathbf{V}, \mathbf{S}\} \tag{3}$$

where $\mathbf{c}$ is the location of the point to be tracked, $\mathbf{H}$ is the learnt mapping for that point, $\mathbf{S}$ are offsets giving the support region, and $\mathbf{V}$ are the values of the support pixels.

Here, the offset positions, $\mathbf{S}$, are chosen as 80 points randomly positioned within a 30-pixel radius. To improve tracking each LP is grouped into a rigid flock. Each flock has 200 LPs. To track the lips and eyes we use 30 landmarks: each was associated with a rigid flock.

The training algorithm is quite subtle. It allows components of a flock to be accepted or rejected on the basis of the effectiveness at predicting $\mathbf{t}$ during training. The final displacement of a flock is the mean of the predicted displacements of its member LPs. †

Each person-specific LP is trained using between 9 to 31 training images. And for each image, a set of 52 landmarks are manually positioned on the contour of eyes and lips. See Figure 2 for examples of some of the landmarks. Note that landmarks around the eyes are tracked purely for the benefit of AAM tracking later.

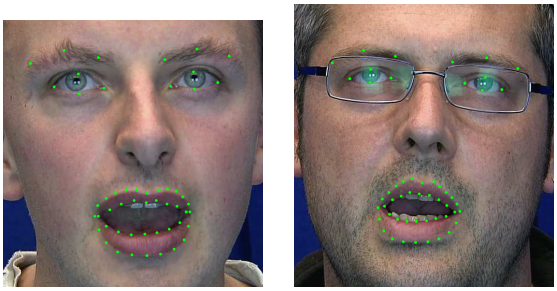


Figure 2. Examples of LP tracked landmarks.

Since our original work in 2009[1] we have made a number of improvements to this basic method which allow improved tracking over a range of poses and which allow some utterance classification directly from the LP features[2] but , for large-vocabulary connected-word recognition we tend use Active Appearance Model (AAM) features which we describe now.

2.2 Feature extraction

The visual features used in this work are based on Active Appearance Models (AAMs),[3] as it has been shown that this type of feature tends to outperform other features typically used in automated lip-reading, including 2D DCT, eigen-lips, and sieves.[4]

The *shape*, $\mathbf{s}$, of an AAM is the concatenation of the x and y-coordinates of a set of n vertices that delineate the features of interest on an object: $\mathbf{s} = (x_1, y_1, \ldots, x_n, y_n)^T$. A compact model that allows a linear variation in the shape is given by,

$$\mathbf{s} = \mathbf{s}_0 + \sum_{i=1}^{m} p_i \mathbf{s}_i, \tag{4}$$

where $\mathbf{s}_0$ is the mean shape and $\mathbf{s}_i$ are the eigenvectors corresponding to the m largest eigenvectors of the covariance matrix. The coefficients p_i are the shape parameters that define the contribution of each eigenvector in the representation of $\mathbf{s}$. The model usually is computed by applying Principal Component Analysis (PCA) to a set of shapes hand-labelled in a corresponding set of images. To obtain the shape vertices, $\mathbf{s}$, in Equation (4), we use a linear predictor-based tracker[1] as we have found that this is more able to robustly track the lip-contour than the AAM.

The *appearance*, A, of an AAM is defined by the pixels that lie inside the base mesh $\mathbf{s}_0$. AAMs allow linear appearance variation, so A can be expressed as a base appearance A_0 plus a linear combination of l appearance images A_i,

$$A = A_0 + \sum_{i=1}^{l} \lambda_i A_i, \tag{5}$$

where λ_i are the appearance parameters. As with shape, the base appearance A_0 and appearance images A_i are usually computed by applying PCA to the shape normalised training images.[3] A_0 is the mean shape normalised image and the vectors A_i are the (reshaped) eigenvectors corresponding to the l largest eigenvalues. Figure 3 is some examples of the variation encoded by the shape and appearance modes.

†For examples of LP tracking results, see `http://www.ee.surrey.ac.uk/Projects/LILiR/update/ej/tracking_web.html`.

(a) shape mode 1 (b) shape mode 2 (c) $A_0 - 1.5\sigma$ (d) A_0 (e) $A_0 + 1.5\sigma$

Figure 3. Shape and appearance variation encoded in the model. (a)(b): shape variation up to ± 1.5 standard deviations ($\pm 1.5\sigma$) in first two modes of the shape model. The first mode describes lip opening and closing, and the second mode lip rounding and spreading. (c)(d)(e): appearance variation encoded by first mode of the appearance model.

We consider two cases for combining the shape and appearance parameters. One is to combine them in a primitive way simply by concatenating the feature vectors (denoted as *cat*), and the second is to combine the features and further reduce the dimensionality using PCA,[5] which we denoted here as a *csam* feature. The combined features can be improved further by applying a Linear Discriminative Analysis (LDA) over a sliding window across multiple frames,[6,7] which are usually referred to as *Hi-LDA* or *hilda* features. In the case of full-frontal lip-reading, it has been shown that the latter two features are the most discriminative.[4] For all features, a per-speaker z-score normalisation is applied which has been shown to improve the separability of the features among classes.[6] We denote features undergoing this normalisation with a suffix *_sp*.

2.3 Classification

For classification, we have a variety of options depending on the task. For a large vocabulary connected-word system our current approach is to use Hidden Markov Models (HMMs) followed by a word-net language model as implemented by the Hidden Markov Toolkit (HTK).[8] This is the same approach as used by state-of-the-art acoustic classifiers so we are gaining the benefit of decades of research in speech recognition. However, for the less challenging task of key-word spotting we can use either HTK or a boosted temporal classifier trained directly on the LP features or a combination of the LP and AAM features.

3. RESULTS

3.1 LiLiR dataset

The variety of systems described in the previous section have been evaluation in a variety of ways. However the core dataset we use, the LiLiR set,[6] is a set of speakers recorded saying the set of sentences selected from the well-known Resource Management task.[9] Since the data are recorded at multiple angles we can explore the question of variability by speaker or by angle. The dataset consists of 20 speakers, 10 male and 10 female, each reciting 200 sentences selected from the Resource Management Corpus.[9] LiLiR was designed for the task of continuous speech recognition. It has a vocabulary size of just over 1000 words, and was recorded at full-frontal and full-profile view points using HD cameras and at 30°, 45° and 60° using SD cameras. All cameras were sync-locked during recording. The speakers were instructed to keep their head relatively still, and the recording of each speaker was done in a single sitting to ensure constant illumination. Figure 4 shows example frames from each pose in our dataset.

We typically build separate shape and appearance model to encode each view independently, thus the *shape* feature (parameter $\mathbf{p}$), and *app* feature (parameter λ) all are view-dependent.

3.2 Baseline results

We have designed a lip-reading system[6] that is based on a set of Hidden Markov Models (HMMs), which are built and manipulated using HTK.[8] All HMMs are viseme level models. A pronunciation dictionary is translated from the phone-level pronunciation to the corresponding viseme-level pronunciation using a standard phone-to-viseme map.[11] The 14 HMMs are trained from the visual features: one for each viseme and one to model 'visual silence'. A 'short pause' model is tied to the middle state of the silence model. Left-to-right HMMs with three states and a diagonal covariance Gaussian Mixture Model (GMM) associated with each state are used. Single

(a) 30° (b) 45° (c) 60°

(d) 0° (e) 90°

Figure 4. Different views of LiLIR dataset[10]

Gaussian HMMs are initialised using flat start training, and this is followed by a series of embedded training. The number of Gaussian mixture components is increased from 1 to 2, 4, 6, 8, 10, and 12. A bigram word language model is constructed from training data. During recognition, various insertion penalties $p = \{-20, 0, 10\}$ and the grammar scale factor $s = \{0, 1, 5, 10, 15, 20, 25, 30\}$ are tested when calling a HTK executable `HVite`. Various features described in Section 2.2 are evaluated using the LiLIR audio-visual dataset described earlier.

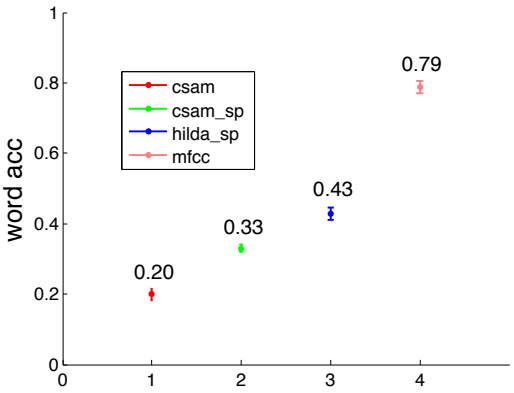


Figure 5. Results for the speaker-independent recognition, evaluated using 12-fold cross-validation. The mean viseme accuracy is plotted with the error bars showing ±1 standard error. Only the highest accuracy rate is reported here for each feature type.

The system is speaker-independent: speakers whose speech is used for testing is not presented during training phase. Data from 12 speakers was selected and 12-fold cross-validation was carried out where for each iteration, a different speaker was assigned to the test. The performance of the classifier is measured using the viseme accuracy rate acc, where acc $= \frac{H-I}{N}$, where N is the total number of viseme instances to be recognised, H is the number of correctly recognised viseme instances, and I is the number of insertion errors. The results for all recognisers are shown in Figure 5. As a matter of interest one can repeat the experiments on professional human

lip-readers.[12] Typical word accuracies for human lip-readers on this set range from 0 to 69%. So, although there is still a large gap between visual and acoustic recognition in Figure 5 the computer accuracy is bracketed by the accuracies measured on professional human lip-readers.

3.3 View-independent lip-reading

Lip-reading systems are usually designed to work using a full-frontal view of the face as was the case for all experiments described in the previous section. However, many human experts tend to prefer to lip-read using an angled view. To find the best angle for automated lip-reading, a system was designed to train and test on the same viewing angle. We call this a *view-dependent system*.[10] AAM features, including *shape*, *app*, *cat*, *csam*, and *hilda* are extracted from the respective views and these are appended with their second derivatives (_ΔΔ).

The results of this experiment are shown in Figure 6(a). The optimal viewing angle for the primitive features, i.e., those that are not subject to a third PCA or an LDA (i.e. *shape*, *app* and *cat*) seems to be 30° or 45°, and the performance drops off at 0°. The performance of the lip-reading systems that use more sophisticated features, i.e. *csam* and *hilda*, are more consistent across view (the corresponding curves in Figure 6 are flatter), which indicates improved robustness to viewpoint change. For 0° and each of the primitive feature types, the system has a noticeable degradation in performance, indicating it is not the best angle for the system.

The result in Figure 6(a) shows, firstly, if the viewing angle is known, say, 30°, then the best performing feature for this angle is *cat*_ΔΔ. Secondly, if the angle is not known, the most robust feature across all views could be used, i.e. *csam*_ΔΔ feature, or *hilda* features. From Figure 6(a) it is also clear that full-frontal view is not the optimal angle for computer lip-reading. This is in agreement with the opinion of many human expert lip-readers who tend to prefer to lip-read at a slight angle. Among all of the views tested here, we choose 30° as the optimal view. This decision is based on the overall performance of all features on each view [‡].

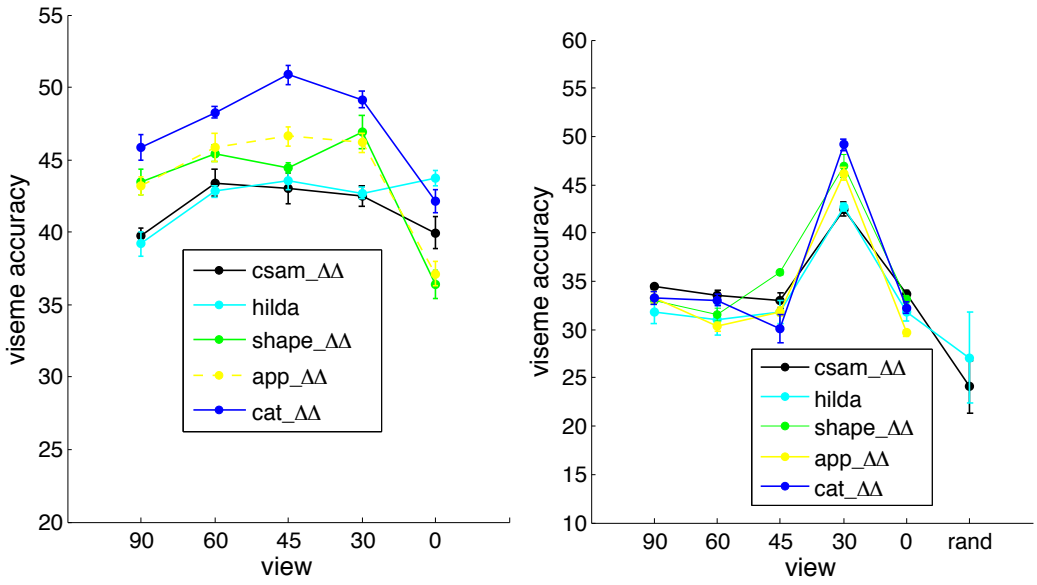


Figure 6. The performance (mean viseme accuracy) of view-dependent lip-reading system (a), and view independent system (b) when training on 30° and testing on all other views. The error bars show ± standard error. The x-axis shows the view angle, the y-axis the percent correct viseme accuracy.[10]

If the optimal angle for lip-reading is 30° then we can extend the system so that it can be applied to angles other than the one on which it was trained, a *view-independent system*. Firslty we train a system on data

[‡]The average performance over all features on 0°, 30°, 45°, 60°, and 90° view are 39.01%, 43.50% 42.98%, 42.73% and 39.85% respectively.

captured at 30° and test the system on data from other views, which allows us to measure the degradation in performance as the viewing angle moves away from the optimal angle as in Figure 6(b). It is worth noting that the features used here are the same as in view-dependent system, i.e., the features themselves are view-dependent, but the recogniser is tested on data from multiple views. To determine the influence of the language model in the accuracy achieved by the recogniser, we also include a test condition where random feature vectors with the same mean, covariance and dimension of our AAM features are input to the system.[10] The results in Figure 6(b) suggests that there is a significant drop in performance for all features compared to Figure 6(a). This is mainly caused by the view-dependency of the features.

Since all visual features are derived from view-dependent models, features from different view are independent of each other, and one would expect that when the system is trained and tested with features from different views the performance will match that of random noise. Experimental results show otherwise and this is an indication that the same feature across different views are somehow related. The assumption is supported by a quick inspection into the modes of variation of shape models for different views. This reveals that across all models, the same mode of variation has the same semantic meaning. For example, the first mode of variation for different models tends to represent lip opening and closing, the second tends to represent lip rounding and spreading and so on. This also indicates that the relationship among the *shape* features across different views might be modelled as a linear transform.

Extracting visual features from speech that was recorded simultaneously from two views. Let $\mathbf{Y} = \{\mathbf{y}_1, ..., \mathbf{y}_N\}$ represent the set of features on the optimal view, and $\mathbf{X} = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ the feature set for the non-desired view. Assume there is a linear transform $\mathbf{T}$ that allows: $\mathbf{Y} = \mathbf{TX}$. $\mathbf{T}$ can be learned on the training data using a ridge regression:[13]

$$\mathbf{T} = \mathbf{YX}^T(\mathbf{XX}^T + h\mathbf{I})^{-1}, \tag{6}$$

in which I is an identity matrix, h is a parameter to prevent over-fitting and is set to 10^{-9} in our experiments.

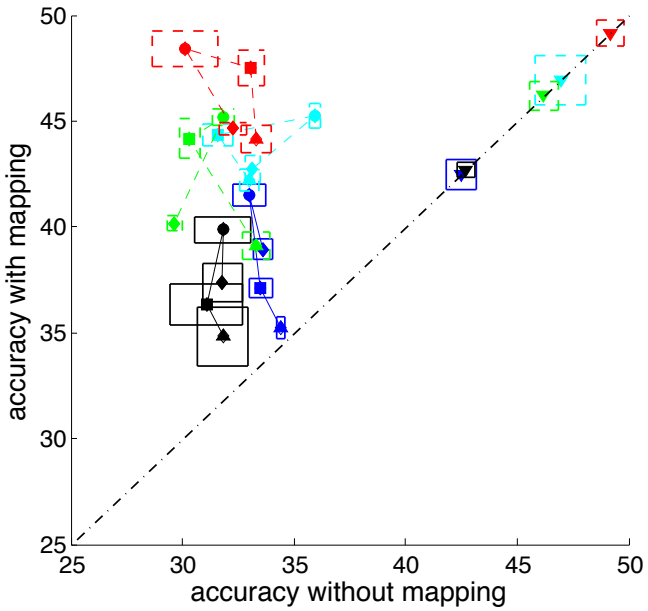


Figure 7. System performance with and without applying the mapping, with x-axis the viseme accuracy where the system is trained on features of 30° view and tested on features of other views without applying the mapping, and with y-axis where the mapping is applied. Features are encoded using lines with different color and marker. Marker $\bigtriangledown$ denotes the performance when system is trained and tested on the 30° angle.[10]

Figure 7 compares the performance of lip-reading systems both before and after mapping. Along the x-axis is the viseme accuracy before mapping, and along y-axis is the performance after applying the mapping. All features reported in the previous two experiments have been tested. Figure 7 selectively plots only four features

to avoid cluttering the figure. Error bars are also plotted and form squares in the space. Different shaped markers denote a different test angle, whose features are mapped back to 30°. If the mapping is ideal and 100% accurate, then the features for the optimal view are re-constructed without error and the performance will be identical to a system trained and tested on the 30° view. This ideal case is also plotted in the figure with a marker $\bigtriangledown$.

A black dashed line partitions the space into two equal halves. If the learning is effective, viseme accuracy will improve after mapping, thus the plot will be above the dashed black line, and if the performance degrades after mapping it will be below the dashed line. For all features, the performance is in the top half, which is a good indication that the ridge regression is working. Of course, some views work better with mapping than others. All features have performance peaks at 60°, i.e, it is the best angle from which features should be mapped to 30°. The absolute difference for viseme accuracy for the ideal case to this angle is 5.38%, 6.35%, 2.60%, 2.03% and 1.66% for *csam*_$\Delta\Delta$, *hilda*, *shape*_$\Delta\Delta$, *app*_$\Delta\Delta$ and *cat*_$\Delta\Delta$ features respectively. We believe that a linear transform is appropriate for the primitive features but the more sophisticated features may be better suited to a non-linear transform because the PCA and the LDA projections used during the feature construction are too complicated to be modelled by a simple linear ridge regression. The best performing features are *cat*_$\Delta\Delta$ features: the performance across all views is the highest, and is also closest to the performance of the ideal case.

4. DISCUSSION

Compared to acoustic speech recognition, visual lip-reading is an incredibly under-explored subject. Furthermore the difficulty of pure lip-reading often encourages people to study audio-visual recognition in which an acoustic recogniser is combined with a lip-reading system. Unfortunately, in our experience, it is very easy to mask the performance limits when building audio-visual systems so the difficult parts of the problem remain even more under-explored. A further feature of lip-reading is the problem of difficulty scaling: as the complexity of the dataset increases (more words, more talkers etc), new effects come into play which means that good results on the digits "zero" to "ten" do not translate into good results on the LiLiR set. There are good reasons for this: co-articulation, for example, has a much longer duration in visual speech than in acoustic speech so is more tricky to model; it appears that visual speech is more variable by person than acoustic speech and so on. The variability between speakers has a considerable implication on the requirement for training data which has not been collected on such a vast scale as for speech technology.

Our approach therefore has been to focus on a relatively large vocabulary dataset across multiple talkers and then to develop sub-tasks, such as keyword spotting, as we wish to develop insights into particular aspects of the science. It is hoped that by this relatively ambitious approach we can ultimately make the conversion of video to text a useful reality for the next generation of crime fighters and law enforcement officials.

ACKNOWLEDGMENTS

The work described in this paper has been funded by a number of sources included the Engineering and Physical Research Council (EPSRC) and the UK Home Office.

REFERENCES

[1] Ong, E., Lan, Y., Theobald, B., R., H., and Bowden, R., "Robust facial feature tracking using selected multi-resolution linear predictors," in [*In Proceedings of the International Conference Computer Vision (ICCV)*], (2009).

[2] Harvey, R., "Keynote speech: Lip-reading: science fact or science fiction," in [*IDEAL 2011: The 12th International Conference on Intelligent Data Engineering and Automated Learning*], (2011).

[3] Cootes, T., Edwards, G., and Taylor, C., "Active appearance models," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **23**, 681–685 (June 2001).

[4] Lan, Y., Harvey, R., Theobald, B., Ong, E.-J., and Bowden, R., "Comparing visual features for lipreading," in [*Proc. of International Conference on Auditory-visual Speech Processing*], 102–106 (2009).

[5] Cootes, T. and Taylor, C., "Statistical models of appearance for computer vision," tech. rep., Imaging Science and Biomedical Engineering, University of Manchester (2004).

[6] Lan, Y., Theobald, B., Harvey, R., Ong, E.-J., and Bowden, R., "Improving visual features for lip-reading," in [*Proceedings of Proceedings of International Conference on Auditory-Visual Speech Processing*], (2010).

[7] Potamianos, G., Neti, C., Luettin, J., and Matthews, I., "Audio-visual automatic speech recognition: An overview," in [*Issues in Visual and Audio-visual Speech Processing*], MIT Press (2004).

[8] Young, S., Evenmann, G., Kershaw, D., Moore, G., Odell, J., Ollason, D., Valtchev, V., and Woodland, P., *The HTK Book (version 3.2.1)* (2002).

[9] Fisher, W., Doddington, G., and Goudie-Marshall, K., "The DARPA speech recognition research database: specifications and status.," in [*In Proceedings of the DARPA Speech Recognition Workshop.*], (1986).

[10] Lan, Y., Theobald, B.-J., and Harvey, R., "View independent computer lip-reading," in [*IEEE Conference on Multimedia and Expo (ICME 2012)*], IEEE (July 2012).

[11] Fisher, C. G., "Confusions among visually perceived consonants," *Journal of Speech and Hearing Research* **11**, 796–804 (1968).

[12] Lan, Y., Harvey, R., and Theobald, B.-J., "Insights into machine lip reading," in [*Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*], 4825 –4828 (march 2012).

[13] Bishop, C. M., [*Pattern Recognition and Machine Learning*], Springer (2006).