Sign Language Recognition: Working with Limited Corpora

Helen Cooper and Richard Bowden

Centre for Vision, Speech and Signal Processing,
University Of Surrey, Guildford, GU2 7XH UK.
{H.M.Cooper,R.Bowden}@surrey.ac.uk

Abstract. The availability of video format sign language corpora limited. This leads to a desire for techniques which do not rely on large, fully-labelled datasets. This paper covers various methods for learning sign either from small data sets or from those without ground truth labels. To avoid non-trivial tracking issues; sign detection is investigated using volumetric spatio-temporal features. Following this the advantages of recognising the component parts of sign rather than the signs themselves is demonstrated and finally the idea of using a weakly labelled data set is considered and results shown for work in this area.

Keywords: Sign Language Recognition, Volumetric Features, Weakly Supervised Learning, Data Mining.

1 Introduction

One of the limiting factors towards obtaining accurate, automatic Sign Language Recognition (SLR) is the lack of adequately-labelled, good-quality training data. However, several television networks broadcast inset signers with their programs, alongside subtitle text. This offers a source of data containing many native signers, covering a wide variety of topics and regional accents. In order to make use of this data, two problems need to be overcome, the first is the ability to work with low resolution data, since the signer typically occupies only a small section, superimposed on the broadcast which is often cluttered by the moving video stream in the background. The second is to investigate ways of using weak linguistic labels rather than traditional frame by frame ground truth. Subtitle-sign alignment in these broadcasts is rarely a one to one mapping and words do not always occur in the same order as the signs due to the differing grammars between spoken and signed languages

Native signers typically sign at a rate which is far faster than that of most specially collected datasets. This combined with the low resolution makes tracking especially difficult, especially as SD broadcast footage is scan line interleaved which leaves fast moving objects (such as the hands) blurred and corrupted by interlacing artefacts.

C. Stephanidis (Ed.): Universal Access in HCI, Part III, HCII 2009, LNCS 5616, pp. 472–481, 2009.

Various alternatives to tracking have been proposed and this paper will present two approaches based upon spatio-temporal classifiers. Along with a method for utilising the large corpus of data available via subtitled broadcasts.

2 Previous Work

Many of the previous solutions to SLR use data gloves to acquire an accurate 3D position and trajectory of the hands [1] which, while facilitating a large vocabulary are cumbersome to the user. The majority of vision approaches are tracking based solutions with relatively small lexicons. Staner and Pentland [2] used colour to segment the hands for ease of tracking and reported classification results on a 40 sign lexicon. More recently, scalability has been addressed by turning to sign linguistics to aid classification. Vogler and Metaxas [3] initial work operated on a lexicon of 53 signs but later reported a scalable solution using parallel HMMs on both hand shape and motion to recognise a 22 sign lexicon. Kadir et al [4] took this further by combining head, hand and torso position as well as hand shape to create a system that could be trained on five or fewer examples on a large lexicon of 164 signs. It is this work that we will make a direct comparison with as the dataset is available and allows our detection approach to be compared with the results of tracking.

Detection/non-tracking based approaches have recently begun to emerge, Zahedi et al [5] apply skin segmentation combined with 5 types of differencing to each frame in a sequence which are then down sampled to get features. Wong and Cippola [6] use PCA on motion gradient images of a sequence to obtain their features. Blank et al used space-time correlation to identify activity [7] while Ke et al [8] employed boosted volumetric features in space-time to detect behaviour. All of these approaches are designed for gesture or behaviour recognition and typically only address a small number of gestures (< 10). It is not obvious how these approaches could be extended to larger lexicons in a scalable way.

In the genre of sign subtitle alignment there has been little work to date. Farhadi and Forsyth perform word spotting on 31 different words over an 80000 frame children's film [9]. In their case the word order is similar in both the signs and the subtitles and there is usually one sign for each occurrence of the word in the subtitle. This is not always the case in sign languages, especially when the content of the video moves away from children's stories and towards more complex concepts such as news.

As the computer vision community looks towards the vast, freely-available data collections available on the internet; such as flickr photos and search engines offering image searches, methods are being developed to cope with the increase in size and the lack of ground truth data. One of the methods which has been used with good results is data mining. It has been implemented for object recognition by grouping together re-occurring spatial features [10], clustering similar images in large data sets [11] and action recognition by combining in videos [12]. These uses demonstrate its efficiency at finding discriminate features in noisy data and the concept can be extended to finding re-occurring signs in video, ignoring the irrelevant noise around them.

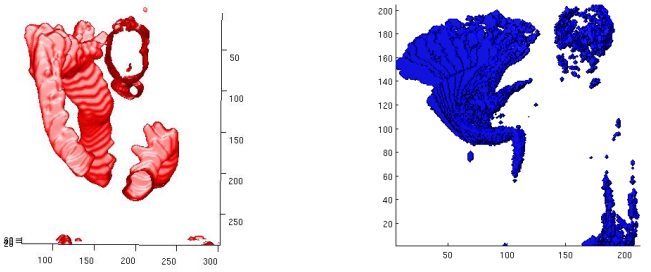

Fig. 1. Examples of signing motion over time

3 Volumetric Classifiers for Sign Detection

Consider a video; time can be viewed as simply the third dimension, stack the frames one behind the other in temporal order and a volume is created. Now instead of looking at sign recognition as being the transitions of positions between one frame and the next it can be viewed as object detection in a 3D block of video.

Take the example of a wave; if the footage is processed with a skin segmentation algorithm then the temporal volume will show how the hands move through time, this is shown in the left hand image of Fig. 1, on the right is the result if a video is processed with a frame differencing algorithm. There is a definite shape to the motion that can be detected.

This first approach uses the volumetric description of space-time and an extension of the types of classifiers used in 2D detection problems, specifically the natural extension of Haar like features into the temporal domain. These features are computed efficiently using an integral volume and are assembled into spatio-temporal classifiers using boosting. Since the boosting creates a single classifier per word with only minimal pre-processing being done on the video (frame differencing or skin segmentation) the compound errors found when classifying tracking results are avoided.

3.1 Integral Volume and Volumetric Features

One of the more well known and robust 2D object detectors is the one used for face detection by Viola and Jones [13]. It uses weak classifiers based on simple block differences with a threshold, these weak classifiers are combined together using boosting. One of the ideas that Viola and Jones introduced to the computer vision community is that of the integral image which allows block summations to be computed in constant time regardless of their size which enables their detector to work in real time. This concept can be directly extended into the temporal domain to create volumetric representations of video. In this any point in this integral volume IV will contain the sum of all points to its upper left plus those before it. This is shown in Fig. 2 and equation (1). Where V is the volume or video to be converted and (x,y,z) and (x',y',z') are points referenced to the top front left corner $(0,0,0)$. Using this

volume any block summation can be calculated using only four subtractions and three additions.

$$IV(x',y',z') = \sum_{x=0}^{x'} \sum_{y=0}^{y'} \sum_{z=0}^{z'} V(x,y,z) \tag{1}$$

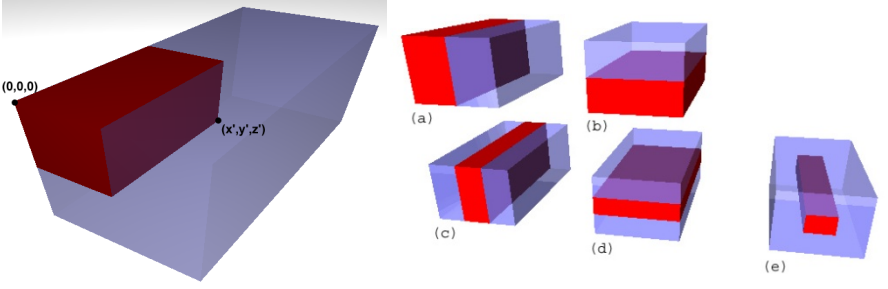

Fig. 2. The integral volume and the features associated with it

Also shown in Fig. 2 are the features used as classifiers, they are calculated as the difference between the translucent block and the solid block, the classifier then responds in a binary manner by applying a threshold to this value. Since sign language revolves around the signer it makes sense that the classifiers used should also be based around the signer. As such the features shown are scaled and positioned relative to the signer. The weak classifiers are then combined by boosting into a strong sign level classifier.

3.2 Boosting

Boosting works by picking the best classifiers from a large set of relatively weak classifiers, working on the basis of strength in numbers it iteratively selects the best classifiers each time until either it exhausts the supply of adequate weak classifiers or it manages to separate the training data. This basic principle boosts the strength of any one weak classifier by combining it with others. There are various flavours of boosting, the most common being AdaBoost which applies weights to the examples and at each iteration adapts these weights to encourage the boosting to pick weak classifiers which separate the more difficult to classify examples. The weights are updated at each iteration based on how well the classifier works on them. The equation for this is shown in (2). Where $w_{n;x}$ is the weight at iteration n of example x, e_n is the error of the weak classifier selected in iteration n and x_c is 0 if the example is correctly classified by that weak classifier and 1 otherwise

$$w_{n+1,x} = w_{n,x} \left(\frac{e_n}{1 - e_n} \right)^{1-x_c} \tag{2}$$

Implementation considerations are discussed in [14] since the shift from a 2D representation to a 3D representation brings with it some memory and processing issues.

(a) Hello (b) Friend

Fig. 3. The first 20 weak classifiers chosen by boosting for the two signs hello and friend

3.3 Results and Conclusions

Some examples of the classifiers built are shown in Fig. 3 As can be seen the features selected are the ones located around where the motion is taking place. The classifiers built from these features can achieve recognition rates of around 90% with false recognition rates below 5%. Further results and more details are available in [14]

While this process proves the concept that it is possible to do sign language recognition without tracking, the results shown are over a small data set. This solution lacks the ability to work with large lexicons since as each new word is encountered, a new classifier must be learnt.

4 Large Lexicon Sign Detection

Linguists use phonemes to describe a sign, each one covering a sub-unit of the sign itself e.g. motion, hand shape, location etc. Phoneme-level detection offers the advantage of increasing the sign lexicon significantly without having to similarly increase the number of classifiers required and therefore the time taken to process data. The benefits of sub word units in the form of phonemes has been proven in the speech recognition community and can similarly be adopted with great effect in the SLR field.

4.1 Stage I: Phoneme Level Features

It has been demonstrated that tracking and some hard coded heuristics can provide suitable features for phonemes [4]; however as discussed in section 0 tracking hand positions for sign is non-trivial and as such this work uses an approach which is based on learnt phoneme detectors.

Three types of phonemes are addressed: *Tab*; the location where the sign is happening in relation to the signers body, *Sig*; the motion the hands make and *Ha*; the hand arrangement. All of these are learnt using boosting as described in section 0 but the features required for each one are different. In the first instance the signer is skin segmented to provide the head and hands. To describe *Tab* a grid is place over the

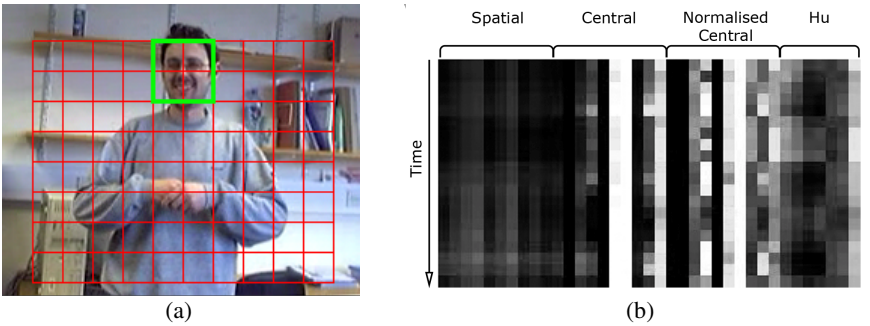

Fig. 4. The grid (a) and moment features (b) used to build phoneme detectors

signer which is relative in size and scale to the signers face, see Fig. 4(a), this produces a group of features, cells of the grid, which fire 1 when they contain more than 50% skin pixels and 0 otherwise. In the case of Ha phonemes, 4 different types of moments are taken for each frame: spatial, central, normalised central and Hu, this offers a good range of variance and invariance to various properties such as rotation and translation. These values are threshold and boosted as with the volumetric features described previously. *Sig* phonemes require temporal information to say what motion is occurring and to this end the features used are those that look for temporal changes in the moment features used by *Ha*. These features are stored as binary patterns, a 1 for an increase in a moment's value and a 0 for a decrease or no change. This means that when looking for a motion such as *'hands move apart'* the boosting would be able to pick binary patterns which show increases in moments linked to eccentricity and vice versa for *'hands move together'*.

4.2 Stage II: Word Level Combination

The boosted phoneme classifiers are combined to create a binary feature vector which is fed into a second stage. In order to represent the temporal transitions which are indicative of a sign, a 1st order assumption is made and a Markov chain is constructed for each word in the lexicon. An ergodic model is used with a Look Up Table (LUT) to maintain as little of the chain as is required. The result is a sparse state transition matrix for each word giving a classification bank of Markov chains.

During classification, the model bank is applied to incoming data in a similar fashion to HMMs. The objective is to calculate the chain which best describes the incoming data i.e. has the highest probability that it produced the observation sequence.

4.3 Results and Conclusions

Whilst any one set of phonemes is unable to distinguish accurately between signs (see[15] for more in depth results) when combined together using the second stage classifier recognition rates reach 72.6% over a lexicon of 164 signs. This work was done in direct comparison with the previous work based on tracking [4] and over the same lexicon the perfectly tracked data could recognise 79.2%. This means that basic

spatial features which are quicker to compute and more robust to poor data quality can achieve almost equivalent performance to tracked gloved data.

Another advantage of this phoneme based method is that fewer training examples are needed since phonemes occur in more than one sign and a few basic phonemes can be combined to create a large vocabulary which reduces the problems associated with co-articulation and its modelling.

5 Weakly Supervised Learning of Sign

While phoneme detection allows the recognition of large lexicons, the problem still stands that the data sets to support these lexicons do not currently exist. As mentioned however there is a large quantity of data being broadcast daily which is weakly labelled in the form of subtitles. If correlations can be found between these weak labels and the signs being performed then the data could be used for learning signs.

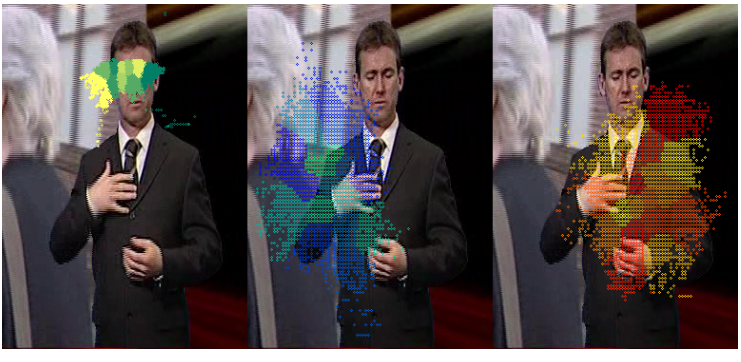

Fig. 5. The quantisation used for the head and hand positions

5.1 Features

An effective tracking system has been built to work on broadcast data [16] and as such the luxury of head and hand positions can be used to describe what is happening on a frame by frame basis. These head and hand positions can be quantised using K-means clustering into a subset of positions. This allows generalisation between frames and also reduces the dimensionality of the problem in hand. The quantisation used is shown in Fig. 5. Note how more positions are required for the more mobile hands than for the head and also that the dominant hand, in this case the signers right, covers a wider range of positions than the non-dominant hand.

5.2 Mining

Once these symbols have been calculated for each frame it is necessary to find correlations between sections of video which share a similar weak label/subtitle. The problem is that the label is only weak and the region of video surrounding a subtitle

may or may not contain one or more examples of the sign. In addition, the distance between the sign and the subtitle can be as much as 200. This is a similar problem to that faced in the data mining community and as such the use of data mining tools is employed. In this case the Apriori mining algorithm[17] has been adapted for use with video data. Apriori mining works by finding an exhaustive list of commonly reoccurring sets of symbols in given examples. In this case we constrain the sets to be within a given temporal window since we know that symbols relating to a single sign will be temporally close in the example. By doing this the number of symbol sets returned is also reduced. Once these sets have been found a response across an example can be obtained. The area of video that contains the most of the frequently occurring sets can be used to repeat the process and iteratively hone in on the desired sign.

Results can be improved by giving the mining algorithm negative examples which should not display the target sign, and therefore its symbol sets. Sections of video to be used as negative data are chosen automatically using a contextual search. To do this the positive data is first examined for subtitle words which occur but which are not the target word,. Common English words which rarely occur in British Sign Language (BSL) as words in their own right are removed from this list. Then a subtitle search is performed for the remaining contextually similar non-target words. These form the negative example set on the basis that they contain signs which are similar to those included in the positive set but which are not the target sign.

Another method for increasing detection rates is to group together words which while separate in English share a common sign in BSL. Examples of this are 'Army' and 'Soldier' or 'Obese' and 'Overweight'. The latter example shows 2 signs which take the same form but there is a sliding scale between them. In English we will chose the word which will offend least or to avoid repetition. In BSL there is less distinction between the two since one is just a more extreme form of the other.

5.3 Results and Conclusions

Using this method 23 signs were mined from a half hour news broadcast. The number of words that can be found is limited by the number of times which they appear in the broadcast. In several cases the words found had as few as 4 positive examples. While combining similar words helps to boost numbers for some signs it's not possible for all. Some example results are shown in Fig. 6. The top bar (blue) shows the original

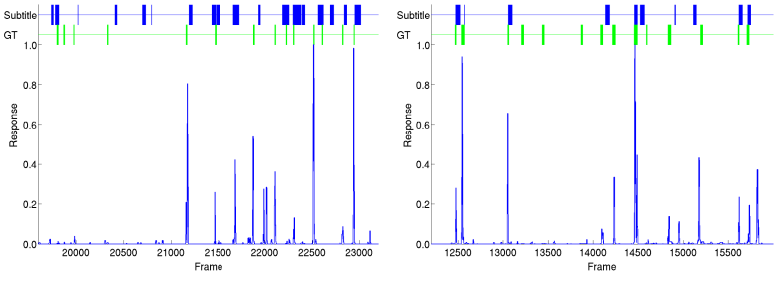

Fig. 6. Results for the signs Army/Soldier (left) and Obese/overweight (right)

subtitle positions, the next (green) shows the ground truth positions of the sign and the main graph shows the response of the mining. A more detailed version of these results as well as the method can be found in [18].

Having shown that it is possible to mine correlations and find signs using near perfect tracked data the next step is to return to a phoneme based representation with appearance based features to avoid the complexities of tracking and to cover more of the phonemes such as hand shape and orientation.

6 Discussions

It has been shown that when the need arises to work with low quality data, tracking is not necessary for sign recognition. This paradigm has then been extended to a phoneme level detection system which can be used with HMMs in much the same way as the comparable phonemic speech recognition systems. This advance increases the lexicon which can be detected and results in a need for data sets which contain more signs, since these are time consuming and non-trivial to create the use of freely available data has been investigated. By using the weak correspondences between subtitles and signs a new source of data is made available to the sign language recognition community. This new corpus is more realistic and therefore more complex than any previous data set; as such it opens up opportunities to focus on the intricacies of sign language and develop techniques which can be used by native signers.

Acknowledgements. The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement nº 231135.

Bibliography

1. Fang, G., Gao, W., Ma, J.: Signer-Independent SLR Based on SOFM/HMM. In: IEEE Int. Conf. on Computer Vision Workshop on Recognition, p. 90 (2001)
2. Starner, T., Pentland, A.: Real-time American Sign Language recognition from video using hidden Markov models. In: Int. Symposium on Computer Vision, p. 265 (1995)
3. Vogler, C., Metaxas, D.: ASL Recognition Based on a Coupling Between HMMs and 3D Motion Analysis. In: Proceedings of the Int. Conf. on Computer Vision, pp. 363–369 (1998)
4. Kadir, T., Bowden, R., Ong, E., Zisserman, A.: Minimal Training, Large Lexicon, Unconstrained Sign Language Recognition. In: British Machine Vision Conf., Kingston, vol. 2, pp. 939–948 (2004)
5. Zahedi, M., Keysers, D., Ney, H.: Appearance-Based Recognition of Words in American Sign Language. In: Second Iberian Conf. in Pattern Recognition and Image Analysis, vol. 1, pp. 511–519 (2005)
6. Wong, S.-F., Cipolla, R.: Real-time Interpretation of Hand Motions using a Sparse Bayesian Classifier on Motion Gradient Orientation Images. In: British Machine Vision Conf., vol. 1, pp. 379–388 (2005)

7. Gorelick, L., Blank, M., Shechtman, E., Irani, M., Basri, R.: Actions as Space-Time Shapes. Trans. on Pattern Analysis and Machine Intelligence 29, 2247–2253 (2007)
8. Ke, Y., Sukthankar, R., Hebert, M.: Efficient Visual Event Detection Using Volumetric Features. In: Int. Conf. on Computer Vision, pp. 166–173 (2005)
9. Farhadi, A., Forsyth, D.: Aligning ASL for Statistical Translation Using a Discriminative Word Model. In: IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, pp. 1471–1476 (2006)
10. Quack, T., Ferrari, V., Leibe, B., Gool, L.: Efficient Mining of Frequent and Distinctive Feature Configurations. In: Int. Conf. on Computer Vision, pp. 1–8 (2007)
11. Chum, O., Matas, J.: Web Scale Image Clustering, Large Scale Discovery of Spatially Related Images. Technical Report, CMP, CTU (2008)
12. Gilbert, A., Illingworth, J., Bowden, R.: Scale Invariant Action Recognition Using Compound Features Mined from Dense Spatio-temporal Corners. In: EU Conf. on Computer Vision, Marseille, pp. 222–233 (2008)
13. Viola, P., Jones, M.: Robust Real-time Object Detection. Int. Journal of Computer Vision (0920-5691), 137–154 (2004)
14. Cooper, H., Bowden, R.: Sign Language Recognition Using Boosted Volumetric Features. In: Machine Vision and Application, Tokyo, pp. 359–362 (2007)
15. Cooper, H., Bowden, R.: Large Lexicon Detection of Sign Language. In: IEEE Int. Conf. on Computer Vision Workshop on HCI, pp. 88–97 (2007)
16. Buehler, P., Everingham, M., Huttenlocher, D., Zisserman, A.: Long Term Arm and Hand Tracking for Continuous Sign Language TV Broadcasts. In: British Machine Vision Conf., Leeds (2008)
17. Agrawal, R., Imielinski, T.: Mining association rules between sets of items in large databases. In: ACM SIGMOD Conf. on Management of Data, pp. 207–216 (1993)
18. Cooper, H., Bowden, R.: Learning Signs from Subtitles: A weakly Supervised Approach to Sign Language Recognition. In: IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, Miami (to appear, 2009)