Reading the Signs: A Video Based Sign Dictionary

Helen Cooper, Nicolas Pugeault and Richard Bowden
Centre for Vision, Speech and Signal Processing,
University Of Surrey, Guildford, UK
helen.cooper|n.pugeault|r.bowden@surrey.ac.uk

Abstract

This article presents a dictionary for Sign Language using visual sign recognition based on linguistic sub-components. We demonstrate a system where the user makes a query, receiving in response a ranked selection of similar results. The approach uses concepts from linguistics to provide sign sub-unit features and classifiers based on motion, sign-location and handshape. These sub-units are combined using Markov Models for sign level recognition. Results are shown for a video dataset of 984 isolated signs performed by a native signer. Recognition rates reach 71.4% for the first candidate and 85.9% for retrieval within the top 10 ranked signs.

1. Introduction

Being able to use a dictionary is a key aspect when learning a new language, yet all existing sign language dictionaries are complex to navigate due to the lack of universal indexing feature (like the alphabet in written language). This work attempts to address this by proposing an interactive video dictionary. The proposed dictionary can be queried by selecting a video example of the sign, the input query is matched to a database of 984 signs and a ranked list of the most similar signs is returned to the user along with linguistic meaning.

In a written language dictionary, words are usually ordered alphabetically and the user can look up a word with ease. In contrast, although there exists various notations for sign language [13, 10, 8, 3], they are mainly used by linguists rather than native sign language users. Moreover, there is no single attribute that can be used to order a sign language dictionary: it could be ordered by handshape (as in [3]), motion, location or by some more abstract concept such as grammatical type, meaning or translation into a spoken language (the latter two being used in books for students learning sign language). None of these are convenient, making the look-up of unknown signs particularly difficult for the average sign user; therefore an interactive dictionary is required. In this article, we make use of automatic sign recognition to query a dictionary and return candidate signs and their definitions. However, we go beyond mere sign recognition by basing our dictionary on the underlying linguistic principles. Therefore, this work has the potential to be further integrated into related sign language fields as an annotation aid [5].

Previous sign recognition systems using tracking-based, sub-unit classifiers, typically hard code basic sub-units [9] and use data driven approaches [7, 18]. From these previous works, the most similar is that of Wang *et al*., who created an American Sign Language (ASL) dictionary based on similarity between signs using a Dynamic Space-Time Warping (DSTW) approach [16]. They used an exemplar sign level approach and did not use Hidden Markov Models (HMMs) due to the high quantities of training data they require. In spite of this, they present results for a dictionary containing 921 signs and later extend this to much improved results on 1113 signs [17].

While these techniques can give good sign level results, they bear little relation to the linguistics of sign language and offer no bridge between the common user and the linguist. Most recently, Pitsikalis *et al*. [11] have proposed a method which uses linguistic labelling to split signs into their constituent parts. From this they learn models specific to their signer, which are then combined via HMMs to create a sign level classifier over 961 signs. In contrast to these previous works, this article describes an interactive sign language dictionary performing recognition based on signer independent linguistic sub-units. We show good performance on a database of 984 signs taken from a Greek Sign Language (GSL) lexicon. However, since the sub-units are linguistically based they are applicable to any available corpus of sign language.

2. Methodology

This section describes the sign recognition framework used by our interactive dictionary. As a first step, hand and head trajectories are extracted from the videos using the method outlined in [12]. These trajectories include not

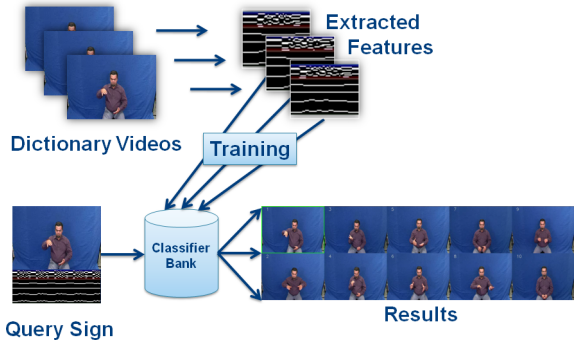
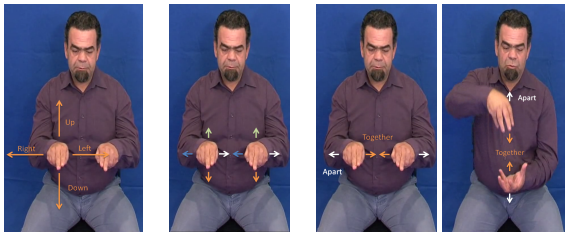


Figure 1: Overview of the recognition system



(a) Single handed (b) Bimanual: Synchronous (c) Bimanual: Together/Apart

Figure 2: Motions detected from tracking

only information about the sign being performed but also information specific to the signer, an accent of their signing style. In order to generalise across different signing styles it is necessary to extract from the base trajectories a signing transcription. To achieve this, a range of classifiers are used to describe the sign in terms similar to SigML [6] or HamNoSys notation [8][1]. The output of these classifiers is then combined using a bank of Markov Models trained to recognise signs in the lexicon. An overview of this process is shown in figure 1.

2.1. Motion Features

While hand tracking produces x and y co-ordinates, sign linguists describe sign motion in conceptual terms such as 'hands move left' or 'dominant hand moves up' [14, 15]. In order to link the x,y co-ordinates obtained from the tracking to the concepts used by sign linguists, rules are employed to extract HamNoSys based information from the trajectories. The approximate size of the head is used as a heuristic to discard ambient motion (that less than 0.25 the head size) and the type of motion occurring is derived directly from deterministic rules on the x and y co-ordinates of the hand position. The types of motions encoded are shown in figure 2,

[1]Note that conversion between the two forms is possible. However while HamNoSys is usually presented as a font for linguistic use, SigML is more suited to automatic processing.

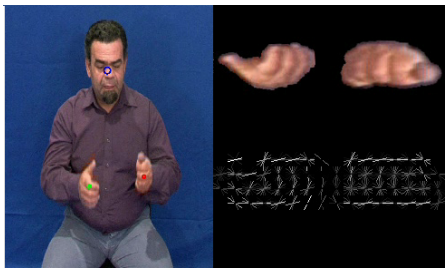


Figure 3: Example HOGs extracted from a frame

the single handed motions are available for both hands and the dual handed motions are orientation independent so as to match linguistic concepts.

2.2. Location Features

Linguistic location characteristics are also abstract from the base x and y co-ordinates; they happen in relation to the signer such as 'at the head' or 'by the elbow'. As such the x and y co-ordinates of the sign location need to be described relative to the signer rather than in absolute pixel positions. This is achieved via quantisation of the values into a codebook based on the signer's head position and scale in the image. For any given hand position (x_h, y_h) the quantised version (x'_h, y'_h) is achieved using the quantisation rules shown in equation 1, where (x_f, y_f) is the face position and (w_f, h_f) is the face size.

$$x' = (x_h - x_f)/w_f$$
$$y' = (y_h - y_f)/h_f \quad (1)$$

This gives values in the range of $y' \in \{0..10\}$ and $x' \in \{0..8\}$ (for a standard signing space) which can be expressed as a binary feature vector of size 40.

2.3. Handshape Features

While the motion and location of the signs can be used for recognition of many examples, it has been shown that adding the handshape can give significant improvement [9]. Histogram of Gradients (HOG) descriptors have proven efficient for sign language hand shape recognition [4] and these are employed as the base feature unit. In each frame, the signer's dominant hand is segmented using the x,y position and a skin model. These image patches are rotated to their principal axis and scaled to a square, 256 pixels in size. Examples of these image patches are shown in figure 3 beside the frame from which they have been extracted. HOGs are calculated over these squares at a cell size of 32 pixels square with 9 orientation bins and with 2x2 overlapping blocks, these are also shown in figure 3. This gives a feature vector of 1764 histogram bins which describes the appearance of a hand.

2.4. Handshape Classifiers

SigML lists 12 basic handshapes which can be augmented using finger bending, thumb position and openeness characteristics. These handshapes are then combined with palm and finger orientations to describe the final hand posture. Currently this work focusses on just the basic handshapes, building multi-modal classifiers to account for the different orientations. A list of these handshapes is shown in figure 4.

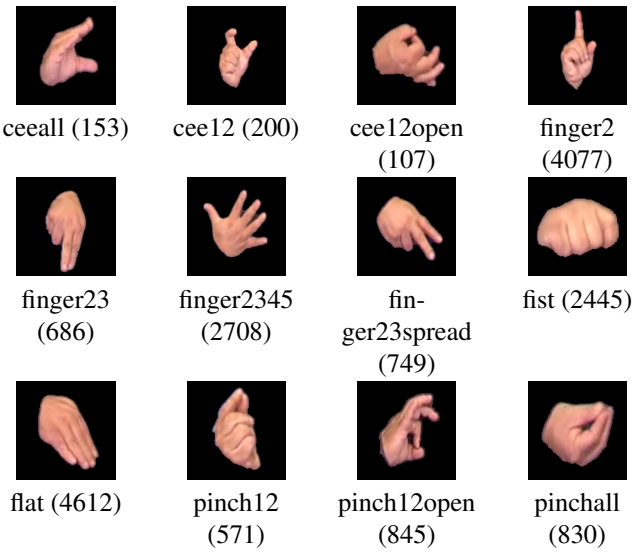


Figure 4: The base handshapes (Number of occurrences in the dataset)

Unfortunately, linguists annotating sign, do so only at the *sign* level while most subunits occur for only *part* of a sign. Also, not only do handshapes change throughout the sign, they are made more difficult to recognise due to motion blur. Using the motion of the hands, the sign can be split into its component parts (as in Liddell *et al*. [10]), that are then aligned with the sign annotations. The frames most likely to contain a static handshape (i.e those with limited or no motion) are extracted for training.

Note that, as shown in figure 5, a single SigML class (in this case 'finger2') may contain examples which vary greatly in appearance, making visual classification an extremely difficult task.

The extracted hand shapes are classified using a multi-class random forest. Random forests were proposed by Amit & Geman [1] and Breiman [2]. They have been shown to yield good performance on a variety of classification and regression problems, and can be trained efficiently in a parallel manner, allowing training on large feature vectors and datasets. In our system, the forest is trained from automatically extracted samples of all 12 handshapes in the dataset, shown in figure 4. Since signs may have multiple handshapes or several instances of the same handshape, the total occurrences are greater than the number of signs, however they are not equally distributed between the handshape classes. The large disparities in the number of examples between classes (see figure 4) may bias the learning, therefore the training set is rebalanced before learning by selecting 1,000 random samples for each class, forming a new balanced dataset. The forest used consists of $N = 100$ multiclass decision trees T_i, each of which is trained on a random subset of the training data. Each tree node splits the feature space in two by applying a threshold on one dimension of the feature vector. This dimension (chosen from a random subset) and the threshold value are chosen to yield the largest reduction in entropy in the class distribution. This recursive partitioning of the dataset continues until a node contains a subset of examples that belong to one single class, or if the tree reaches a maximal depth (set to 10). Each leaf is then labelled according to the mode of the contained samples. As a result, the forest yields a probability distribution over all classes, where the likelihood for each class is the proportion of trees that voted for this class. Formally, the confidence that feature vector x describes the handshape c is given by:

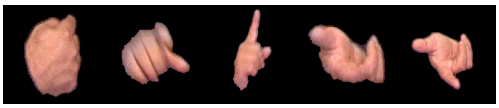


Figure 5: A variety of examples for the HamNoSys/SigML class 'finger2'.

$$p[c] = \frac{1}{N} \sum_{i<N} \delta_c(T_i(x)), \tag{2}$$

where N is the number of trees in the forest, $T_i(x)$ is the leaf of the ith tree T_i into which x falls, and $\delta_c(a)$ is the Kronecker delta function ($\delta_c(a) = 1$ iff. $c = a$, $\delta_c(a) = 0$ otherwise).

The performance of this hand shape classification on the test set is recorded on table 1, where each row corresponds to a shape, and each column corresponds to a predicted class (empty cells signify zero). Lower performance is achieved for classes that were more frequent in the dataset. The more frequently a sign occurs in the dataset the more orientations it is likely to be used in. This in turn makes the appearance of the class highly variable; see, *e.g*., figure 5 for the case of 'finger2'—the worst performing case. Also noted is the high confusion between 'finger2' and 'fist' most likely due to the similarity of these classes when the signer is pointing to themselves.

The handshape classifiers are evaluated for the right hand only during frames when it is not in motion. We evaluated

handshape	predictions											
flat	**0.35**	0.19	0.09	0.03	0.08	0.06	0.03	0.06	0.06	0.01	0.03	0.01
fist	0.03	**0.69**	0.02	0.04	0.11	0.05		0.02	0.03			0.02
finger2345	0.16	0.19	**0.36**	0.02	0.03	0.05		0.06	0.02	0.03	0.06	0.01
finger2	0.02	0.33	0.07	**0.31**	0.11	0.05	0.02	0.03	0.02		0.04	
pinchall	0.03	0.09	0.04	0.01	**0.65**	0.11	0.01	0.01				0.04
pinch12	0.02	0.20	0.01	0.02	0.13	**0.56**	0.01		0.01		0.01	0.02
finger23	0.05	0.17	0.04	0.02	0.05	0.04	**0.54**	0.01			0.07	0.01
pinch12open	0.03	0.12	0.07	0.01	0.15	0.04	0.01	**0.56**				0.01
cee12	0.01	0.05	0.01	0.03	0.04			0.01	**0.82**		0.01	
cee12open					0.01					**0.99**		
finger23spread	0.01	0.15	0.02		0.06	0.01	0.05	0.02			**0.65**	
ceeall	0.01	0.08	0.03		0.08	0.01	0.02	0.01			0.01	**0.77**

Table 1: Confusion matrix of the handshape recognition, for all 12 classes.

our sign recognition system using two different encodings for the detected hand shapes. As will be described in section 2.5, the next stage classifier requires inputs in the form of binary feature vectors. Two types of 12 bit binary feature vector can be produced from the classifier results. The first method applies a strict Winner Takes All (WTA) on the multiclass forest's response: the class with the highest probability is set to one, and the others to zero. For every non-motion frame, the vector contains a true value in the highest scoring class. The second method applies a fixed threshold ($\tau = 0.25$) on the confidences provided by the classifier for each of the 12 handshapes classes. Handshapes that have a confidence above threshold ($p[c] > \tau$) are set to one, and the others to zero. This soft approach carries the double advantage that a) the feature vector may encode the ambiguity between handshapes, which may itself carry information, and b) may contain only zeros if confidences in all classes are small. The two methods are compared in table 2.

2.5. Sign Level classification

The three types of binary feature vectors are combined to create a single feature vector per frame. The frame vectors are then fed into a sign level classifier, similar to that used in Kadir *et al.*'s work [9]. In order to represent the temporal transitions, which are indicative of a sign, a Markov chain is constructed for each word in the lexicon. This is possible as the symbolic nature of HamNoSys allows the discrete time series of events to be modelled without a hidden layer. Another advantage of this binary, abstracted approach is that it allows for better generalisation requiring far less training data than approaches which must generalise over both a continuous input space as well as the variability between signs (e.g. HMMs). An ergodic model is used and a Look Up Table (LUT) employed to maintain as little of the chain as is required. Code entries not contained within the LUT are assigned a nominal probability. This is done to avoid otherwise correct chains being assigned zero probabilities if noise corrupts the input signal. The result is a sparse state transition matrix, $P_\omega(s_t|s_{t-1})$, for each word ω giving a classification bank of Markov chains. Comparisons are also drawn between using only the 1st order transitions and including transitions where the transition matrix includes $P_\omega(s_t|s_{t-2})$. This is similar to adding skip transitions to the left-right hidden layer of a HMM which allows deletion errors in the incoming signal. A simplified model for the sign 'Anger' is shown in figure 6, it shows the 3 main states of the starting position, the motion and the final position. The possible skip transition is shown by the dotted line, simplified, if the starting position and the end position are seen then there is some probability that the sign has occurred even if the motion has not been seen. While it could be argued that the linguistic features constitute discrete emission probabilities; the lack of a doubly stochastic process and the fact that the hidden states are determined directly from the observation sequence, separates this from traditional HMMs which cannot be used due to their high training requirements.

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 s. Symbols are found in the symbol LUT using an L1 distance on the binary vectors. The probability of a model matching the observation sequence is calculated as $P(\omega|s) = \upsilon \prod_{t=1}^{l} P_\omega(s_t|s_{t-1})$, where l is the length of the word in the test sequence and υ is the prior probability of a chain starting in any one of its states, as in a dictionary setting all words are equally likely and there is no language model υ is set to 1.

3. Results

The data set used for these experiments contains 984 GSL signs with 5 examples of each performed by a single signer (for a total of 4920 samples). The handshape classifiers are learnt on data from the first 4 examples of each sign. The sign level classifier is trained on the same 4 examples, the remaining sign of each type is reserved for testing. Since the application of this method is a dictionary, the ac-

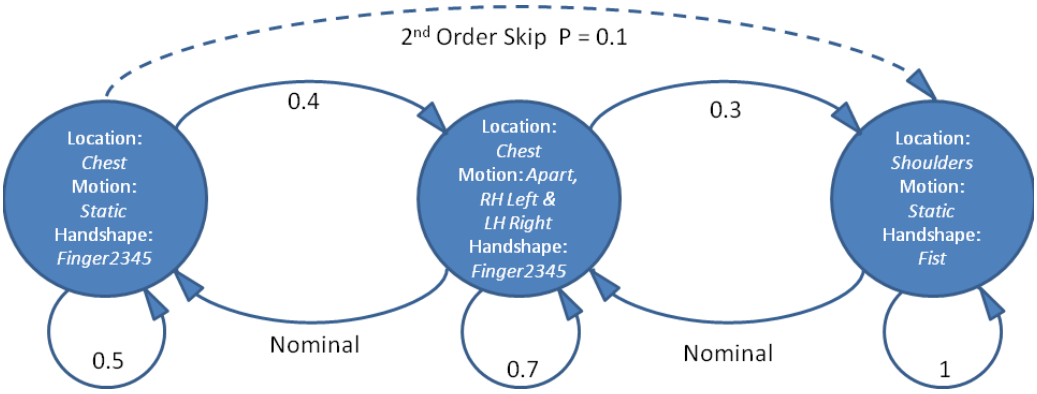


Figure 6: A simplified Markov model for the sign for 'Anger'

curacy results are presented as how often the correct result appears within the top n returned signs. As with any search system, the higher up the results the correct sign appears, the better the ranking system. For this reason, results are shown for various values of n. Ideally accuracy should be high for low values of n, and results are shown for $n = 1$ and $n = 10$.

Table 2 shows sign level classification results. It is apparent from these results that of the independent vectors the location information is the strongest. This is due to the strong combination of a detailed location feature vector and the temporal information encoded by the Markov chain. While neither the motion or handshapes perform well on their own, by combining either of them with location a gain of 6% at $n = 1$ is achieved.

Also of interest is the high results achieved by using just location and handshape. While the combination of all features is better for high ranking results, this combination appears to outperform the others when $n = 10$. This is likely due to the rigidity of the location features and the temporal nature of the Markov chain. When combined they produce similar information to that contained within the motion features. However, it is less discriminating and only enhances the results as more results are returned ($n \rightarrow \inf$). Since the level of n is how many results a signer must look at before finding the sign they require, higher performance at lower values of n is desirable.

Shown also is the improvement afforded by using the handshape classifiers with a threshold vs a WTA implementation. By allowing the classifiers to return multiple possibilities more of the data about the handshape is captured. Conversely, when none of the classifiers is confident, a 'null' response is permitted which reduces the amount of noise. Using the non-mutually exclusive version of the handshapes in combination with the motion and location the percentage of signs returned when $n = 1$ is 68.4% rising to 85.3% when $n = 10$. By including the 2nd order transitions whilst building the Markov chain, the return rate when $n = 10$ is not significantly improved (85.9%) however there is a 3% boost to 71.4% when $n = 1$.

While the work of Wang *et al.* [16, 17] uses a different dataset so direct comparisons cannot be drawn, theirs is of a comparable lexicon size (921 or 1,113 signs vs 984) and uses the same dictionary ranking measure. However, the work of Pitsikalis *et al.* [11] uses the same dataset so more meaningful comparisons can be drawn with this work. While their work also uses linguistic feature sets they learn theirs from the data. They achieve a rate of 62%[2] compared to the 71.4% achieved by the proposed method.

4. Conclusions and Future Work

This article presented an interactive sign language dictionary based on visual sign language recognition. The dictionary exists as a video based tool, allowing a signer to query the dictionary without having prior knowledge of linguistic concepts.

The approach uses computer vision sign language recognition based on hand shapes, motion, position and temporal sequence. The results show that a combination of these cues overcomes the high ambiguity and variability in the dataset to achieve excellent recognition performance: on a large dataset of 984 signs, the correct sign appears as first candidate in 73% of searches, and within the ten best in 85.9% of cases. Moreover, and in contrast to existing approaches, our method is based on non-language specific linguistic sub-units, this opens new perspectives for linguistic annotation, a task which is currently done almost entirely by hand.

Future work should consider moving the location features to a more signer-centric approach to complement those used by the linguists and extending this dictionary to a larger dataset, containing multiple signers. Quantitative results from a larger dataset would not only aid sign recognition but also the sub-unit classifiers which will be of use

[2] read from their graph

n	Motion (M)	Location (L)	Handshape (H)	M + L	M + H	L + H	All: WTA	All: Thresh	All + Skips	[17]	[11]
1	25.1%	60.5%	3.4%	66.5%	36.0%	66.1%	52.7%	68.4%	**71.4%**	44.0%	62%
10	48.7%	82.2%	17.3%	82.7%	60.7%	86.9%	59.1%	85.3%	**85.9%**	78.4%	N/A

Table 2: Sign level classification results

The first three columns show the results when using the features independently with the Markov chain (The handshapes used are non-mutually exclusive). The following three columns show the advantages gain by combining two different feature vector types together. The next three columns give the results of using all the different feature vectors. Including the improvement gained by allowing the handshapes to be non-mutually exclusive (thresh) versus the WTA option. The final suggested method is the combination of the superior handshapes with the location, motion and the Markov chain skips. The last two columns show the results of Wang *et al*. [17] who also use the dictionary ranking measure and Pitsikalis *et al*. [11] who use the same data set.

for linguistic annotation.

5. 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 - DictaSign

References

[1] Y. Amit and D. Geman. Shape quantization and recognition with randomized trees. *Neural Computation*, 9:1545–1588, 1997. 3

[2] L. Breiman. Random forests. *Machine Learning*, pages 5–32, 2001. 3

[3] British Deaf Association. *Dictionary of British Sign Language/English*. Faber and Faber, 1992. 1

[4] P. Buehler, M. Everingham, and A. Zisserman. Learning sign language by watching TV (using weakly aligned subtitles). In *Procs. of CVPR*, pages 2961 – 2968, Miami, FL, USA, June 20 – 26 2009. 2

[5] P. Dreuw and H. Ney. Towards automatic sign language annotation for the elan tool. In *Procs. of Int. Conf. LREC Wkshp : Representation and Processing of Sign Languages*, Marrakech, Morocco, June 2008. 1

[6] R. Elliott, J. Glauert, J. Kennaway, and K. Parsons. D5-2: SiGML Definition. *ViSiCAST Project working document*, 2001. 2

[7] J. Han, G. Awad, and A. Sutherland. Modelling and segmenting subunits for sign language recognition based on hand motion analysis. *Pattern Recognition Letters*, 30(6):623 – 633, Apr. 2009. 1

[8] T. Hanke and C. Schmaling. *Sign Language Notation System*. Institute of German Sign Language and Communication of the Deaf, Hamburg, Germany, Jan. 2004. 1, 2

[9] T. Kadir, R. Bowden, E. Ong, and A. Zisserman. Minimal training, large lexicon, unconstrained sign language recognition. In *Procs. of BMVC*, volume 2, pages 939 – 948, Kingston, UK, Sept. 7 – 9 2004. 1, 2, 4

[10] S. K. Liddell and R. E. Johnson. American sign language: The phonological base. *Sign Language Studies*, 64:195 – 278, 1989. 1, 3

[11] V. Pitsikalis, S. Theodorakis, C. Vogler, and P. Maragos. Advances in phonetics-based sub-unit modeling for transcription alignment and sign language recognition. In *Procs. of Int. Conf. CVPR Wkshp : Gesture Recognition*, Colorado Springs, CO, USA, June 21 – 23 2011. 1, 5, 6

[12] A. Roussos, S. Theodorakis, V. Pitsikalis, and P. Maragos. Hand tracking and affine shape-appearance handshape sub-units in continuous sign language recognition. In *Procs. of Int. Conf. ECCV Wkshp : SGA*, Heraklion, Crete, Sept. 5 – 11 2010. 1

[13] W. C. Stokoe. Sign language structure: An outline of the visual communication systems of the american deaf. *Studies in Linguistics: Occasional Papers*, 8:3 – 37, 1960. 1

[14] R. Sutton-Spence and B. Woll. *The Linguistics of British Sign Language: An Introduction*. Cambridge University Press, 1999. 2

[15] C. Valli, C. Lucas, and K. J. Mulrooney. *Linguistics of American Sign Language: An Introduction*. Gallaudet University Press, 2005. 2

[16] H. Wang, A. Stefan, and V. Athitsos. A similarity measure for vision-based sign recognition. In *Proceedings of the 5th International Conference on Universal Access in Human-Computer Interaction. Part III: Applications and Services*, UAHCI '09, pages 607–616, Berlin, Heidelberg, 2009. Springer-Verlag. 1, 5

[17] H. Wang, A. Stefan, S. Moradi, V. Athitsos, C. Neidle, and F. Kamangar. A system for large vocabulary sign search. In *ECCV International Workshop on Sign, Gesture, and Activity*, Crete, Greece, Sept. 2010. 1, 5, 6

[18] P. Yin, T. Starner, H. Hamilton, I. Essa, and J. M. Rehg. Learning the basic units in american sign language using discriminative segmental feature selection. In *Procs. of ASSP*, pages 4757 – 4760, Taipei, Taiwan, Apr. 19 – 24 2009. 1