Automatic Alignment of HamNoSys Subunits for Continuous Sign Language Recognition

Oscar Koller[1,2], Hermann Ney[1] and Richard Bowden[2]
[1] Human Language Technology and Pattern Recognition Group - RWTH Aachen University, Germany
[2] Centre for Vision Speech and Signal Processing - University of Surrey, Guildford, UK
{koller,ney}@cs.rwth-aachen.de, r.bowden@surrey.ac.uk

Abstract

This work presents our recent advances in the field of automatic processing of sign language corpora targeting continuous sign language recognition. We demonstrate how generic annotations at the articulator level, such as HamNoSys, can be exploited to learn subunit classifiers. Specifically, we explore cross-language-subunits of the hand orientation modality, which are trained on isolated signs of publicly available lexicon data sets for Swiss German and Danish Sign Language and are applied to continuous sign language recognition of the challenging RWTH-PHOENIX-Weather corpus featuring German Sign Language. We observe a significant reduction in word error rate using this method.

Keywords: Sign Language Recognition, Subunits, HamNoSys, Hand Orientation

1. Introduction

Traditionally, sign language corpora intended for machine learning have been annotated at the gloss level as annotation is a time consuming and expensive process. However, glosses used as basic modelling units do not scale well with increasing corpus sizes. Furthermore, singleton signs, which have only a single token for training, make it difficult to find smooth model distributions reflecting the sign accurately. This problem, often referred to as one-shot learning, requires a single training instance to generalise over all possible variations to be encountered in the test data. Shared subunits across the different types of a corpus reduce the negative effect of singleton signs, as the composing subunits usually occur many times throughout the corpus and can therefore be robustly estimated.

Nowadays, several lexical corpus collections exist (Braem, 2001; Jette H. Kristoffersen et al., 2008 2016; McKee et al., 2015; Finish Association of the Deaf, 2015) comprising HamNoSys or other subunit transcriptions. In order to exploit and combine existing annotation efforts from different corpora, we perform automatic alignment on the subunit level. Specifically, this work explores cross-language-subunits (trained on Swiss German and Danish Sign Language) describing the hand orientation articulator. This modality has so far been mostly unexplored, due to the large variability attributed to it. In this way, we propose a method to solve the problem of missing subunit annotations, while still being able to train linguistically derived subunits. The subunit alignments may be used to train a deep convolutional neural network which model subunit representations across different data sets and even sign languages. The Convolutional Neural Network (CNN) is pre-trained and 22 layers deep. Finally, we apply the learnt models as feature extractors on our initial gloss annotated machine learning corpus and perform continuous sign language recognition of challenging real-life data on the publicly available RWTH-PHOENIX-Weather corpus (Forster et al., 2014). We observe a significant reduction in word error rate using this method.

This paper is organised as follows: after introducing the related literature in Section 2, we present the employed data sets in Section 3. We then present the proposed approach in Section 4 and evaluate it in Section 5. The paper closes with a conclusion in Section 6.

2. Related Work

There is a large body of research looking at sign subunits for sign language recognition. There are two broad classes of approaches: (i) data driven subunits, (ii) linguistically derived subunits. Both approaches have been compared to each other, with different outcomes. In Pitsikalis et al. (2011) phonetically derived subunits outperform data driven subunits by 7% on average. However, generally speaking, it is often due to missing subunit-level annotations that researchers opt for the data driven approach. Data driven approaches usually split the signs up by a segmentation algorithm, which is often based on discontinuities in hand movement velocity, such as in Theodorakis et al. (2014). In Bauer and Kraiss (2002), a limited number of signs is arbitrarily segmented which then serves as seed for either an Expectation Maximization (EM)-like iterative refinement of subunits or k-means to find subunit clusters (Kong and Ranganath, 2014). Other approaches use sparse coding to generate a sign dictionary (Yin et al., 2015).

The first sign language recognition system, presented in Tamura and Kawasaki (1988), employed linguistically derived subunits. Usually, linguistic subunit annotations provide a way to break whole signs up into constituent parts and construct a lexicon (Vogler and Metaxas, 1999; Pitsikalis et al., 2011). Other approaches use iterative EM to derive mouth-subunits from pronounced words (Koller et al., 2014). Similarly, available annotations can be aligned based on HamNoSys (Pitsikalis et al., 2011) or SignWriting (Koller et al., 2013; Koller et al., 2016) to the signed footage. The deployment of the subunit classifiers is handled differently. In Cooper et al. (2012) and Kadir et al. (2004) subunits classifiers are learnt and then combined into a second stage sign-level classifier. Systematic comparisons between subunit and whole sign modelling exist.

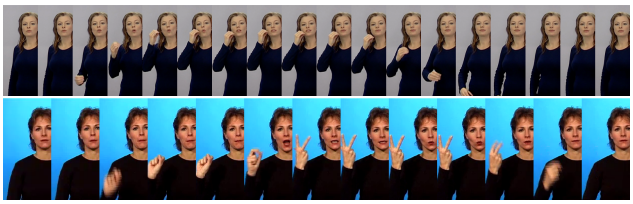

Figure 1: Showing employed data sets for training: Top to bottom, Danish sign language dictionary (Jette H. Kristoffersen et al., 2008 2016) and the Swiss German Sign Language dictionary (Braem, 2001).

In Vogler and Metaxas (1999), which is based on the movement and hold model with linguistic subunits, a powerglove hand tracker helps to perform continuous sign language recognition (CSLR). The authors conclude that sign level modelling slightly outperforms subunit modelling on a 22 sign vocab task trained on 400 sentences.
Hand location and movement are the most frequently encountered modalities used in subunit modelling schemes, closely followed by the handshapes. However, in Waldron and Kim (1995) they have been combined with 11 orientation subunits to recognise a 14 sign vocabulary.

3. Data Sets

Two different sign language dictionary data sets are employed for training the hand orientation classifier, which cover isolated signs. The first represents isolated signs from Danish Sign Language (Jette H. Kristoffersen et al., 2008 2016) with linguistic annotations, and the second features Swiss German Sign Language (Braem, 2001) with provided HamNoSys annotation (Prillwitz et al., 1989). The Danish data features high quality video footage recorded with 720×576 pixel, with very little motion blur. The Swiss German data originates from the year 2001 and is captured at a low resolution, the majority of videos being 320×240 pixel. It contains motion blur and the frames are interlaced. Figure 1 shows data examples of both sources. Both lexica provide hand orientation labels. The Danish data follows its own annotation scheme, which seems to be derived from HamNoSys. From a pattern recognition point of view annotations from both data sets are ambiguous, noisy and partly inconsistent. The chosen modality in this work are hand orientations. An isolated signed instance therefore consists of a finger orientation and a palm orientation annotation, sometimes sequences of two or more such annotations. This can be seen in Figure 1, where the top row depicts a signed instance comprising a single hand orientation and the bottom row shows an orientation sequence that transitions from "fingers:up palm:frontleft" to "fingers:up palm:back". The signer brings his hands from a neutral position to the place of sign execution, while transitioning from a neutral hand orientation to the target hand orientation. The sign may involve a hand movement, a rotation of the hand and changes in hand shape. The annotation may represent any of these hand orientations or an intermediate configuration that was considered linguistically dominant during the annotation. It is also important to note that most linguistic annotations are done for the canonical form, which does not necessarily reflect the exact articulated instance we have access to in the video. Statistics of the two employed data sets are given in Table 1. Garbage and hand orientation frame counts are estimated automatically by our algorithm, which is done separately for finger and palm orientations. Both setups yield slightly differing numbers, which are both presented in Table 1. Both data sets jointly feature nearly $100,000$ frames of hand orientation performed by about 28 different signers.

	Danish	Swiss
duration [min]	97	200
# frames	145,720	299,864
↳ autom. orient.	32,574 / 44,432	60,643 / 55,005
↳ autom. garbage	113,146 / 101,288	239,221 / 244,859
# signed sequences	2,149	4,730
# signs	2,149	4,730
# signers	6	~ 22

Table 1: Corpus statistics: Danish ('Danish') and Swiss German ('Swiss') Sign Language data sets used for training the finger and palm orientation classifier. 'orient.' stands for orientation. The automatic frame counts are given for the finger orientation and the palm orientation. Therefore, two different estimated numbers are presented.

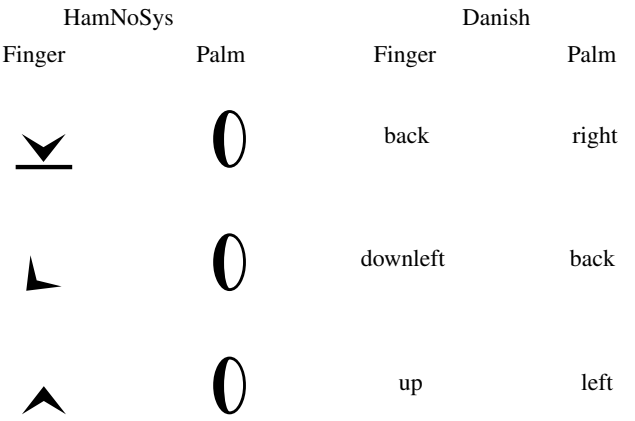

Figure 2: Showing an example mapping from HamNoSys to the Danish notation. It is apparent that in the HamNoSys annotation the palm orientation is coded in dependence of the finger orientation.

For the purpose of combining both lexicon data sets in the scope of this work we needed to create a mapping from HamNoSys to the Danish annotation. This was done manually and had to accommodate the fact that the Danish data set provided independent annotations for finger and palm orientation, whereas in HamNoSys the palm orientation is coded to be dependent on the finger orientation. This means that the same annotated palm orientation symbol can refer to different actual palm orientations depending on the current finger orientation. This is depicted in Figure 2.

After joining both annotation schemes, there is a total of 24 finger orientation classes and 26 palm orientation classes.

Finally, we evaluate on the publicly available continuous sign language data set benchmark RWTH-PHOENIX-Weather 2014 Multisigner corpus (Forster et al., 2014), which is a challenging real-life continuous sign language corpus that can be considered to be one of the largest pub-

lished continuous sign language corpora. It covers unconstrained sign language of 9 different signers with a vocabulary of 1081 different signs. The data set is presented in detail in Koller et al. (2015).

4. Approach

This paper builds on our previous work (Koller et al., 2016), which is extended to the modality of hand orientations and to cover HamNoSys annotations. In the following subsections, we briefly explain the developed HamNoSys parsing, first introduce our weakly supervised learning framework and then describe how to incorporate the learnt subunit classifiers into continuous sign language recognition.

4.1. HamNoSys Parsing

The data set annotations are coded in HamNoSys, an established annotation scheme primarily developed for linguistic purposes. It contains sufficient detail to directly animate an avatar. Each sign described by HamNoSys is composed of clusters of handshape, orientation, place of articulation and movement. HamNoSys does not contain explicit segmentation information. Due to the economic writing style, HamNoSys is very minimalistic, but also needs a parsing that corrects missing information.

We first convert the HamNoSys annotations to SIGML (Glauert and Elliott, 2011). In order to be able to compensate for the palm orientations being dependent on the finger orientations, we need to ensure that a palm orientation occurs always in the context of a finger orientation. However, in transitions from a specific palm orientation to another, HamNoSys dismisses those modalities that do not change. The parser needs to take care of adding this missing information back in. After that, the mapping from HamNoSys to a non-dependant annotation scheme can be easily accomplished. An example of such a mapping is presented in Figure 2. Finally, finger orientation and palm orientation annotations are separated in order to train them as single classifiers.

4.2. Weakly Supervised Subunit Learning

Our weakly supervised CNN training algorithm constitutes a successful solution to the problem of weakly supervised learning from noisy sequence labels to correct frame labels. Figure 3 gives an overview of the approach applied to the learning of hand orientation subunits. The input images are cropped around the tracked hands, which forms the input to the weakly supervised CNN training. The iterative learning algorithm is initialised with a 'flat start', linearly partitioning the input frames to an available best guess annotation, usually a single hand orientation class preceded and followed by instances of the garbage class (as the orientation subunit is expected to happen in the middle of the sequence). The algorithm iteratively refines the temporal class boundaries and trains a CNN that performs single image hand orientation recognition (being a separate finger and palm orientation classification). While refining the boundaries, the algorithm may drop the label sequence or exchange it for one that better fits the data. The iterative process is similar to a forced alignment procedure, however, rather than using Gaussian mixtures as the probabilistic component we use the outputs of the CNN directly.

4.2.1. Problem Formulation

Following Koller et al. (2016), we have a sequence of images $x_1^T = x_1, \ldots, x_T$ and an ambiguous class label $\tilde{l}$ for the whole sequence, we want to jointly find the true label l for each frame and train a model such that the class symbol posterior probability $p(k|x)$ over all images and classes is maximised. We assume that a lexicon ψ of possible mappings from $\tilde{l} \rightarrow l$ exists, where l can be interpreted as a sequence of up to L class symbols k,

$$\psi = \left\{ \tilde{l} : l_1^L \mid l \in \{k_1, \ldots, k_N, \varnothing\} \right\} \tag{1}$$

Optionally, l may be an empty symbol corresponding to a garbage class. Each $\tilde{l}$ can map to multiple symbol sequences (which is important as $\tilde{l}$ is ambiguous and a one-to-one mapping would not be sufficient). In terms of sequence constraints, we only require each symbol to span an arbitrary length of subsequent images as we assume that symbols (in our application: hand orientation subunits) are somewhat stationary and do not instantly disappear or appear.

Due to the promising discriminatory capabilities of CNNs, we solve the problem in a iterative fashion with the EM algorithm (Dempster et al., 1977) in a Hidden-Markov-Model (HMM) setting and use the CNN to model the visual appearance of hand orientations. EM iteratively updates the assignment of class labels to images (E-Step) and then re-estimates the model parameters to adapt to the change (M-Step). We closely follow Koller et al. (2016) and, inspired by the hybrid approach (Bourlard and Morgan, 2012) known from Automatic Speech Recognition (ASR), we include the CNN's posterior output to likelihoods given the class counts in our data using Bayes' rule.

4.3. Convolutional Neural Network Architecture

Knowing the weakly supervised characteristics of our problem, we would like to incorporate as much prior knowledge as possible to guide the search for the true symbol class labels. Pre-trained CNN models constitute such a source of knowledge, which seems reasonable as the pre-trained convolutional filters in the lower layers may capture simple edges and corners, applicable to a wide range of image recognition tasks. We opt for a model previously trained in a supervised fashion for the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) 2014. We choose a 22 layer deep network architecture following (Szegedy et al., 2014) which achieves a top-1 accuracy of 68.7% and a top-5 accuracy 88.9% in the ILSVRC. The network involves an inception architecture, which helps to reduce the numbers of free parameters while allowing for a very deep structure. Our model has about 6 million free parameters. All convolutional layers and the last fully connected layer use rectified linear units as non-linearity. Additionally, a dropout layer with 70% ratio of dropouts is used to prevent over-fitting. We base our CNN implementation on Jia et al. (2014), which is an efficient C++ implementation using the NVIDIA CUDA Deep Neural Network GPU-accelerated library. We replace the last pre-trained fully connected layers before the output layers with those matching the number of classes in our problem (plus one garbage

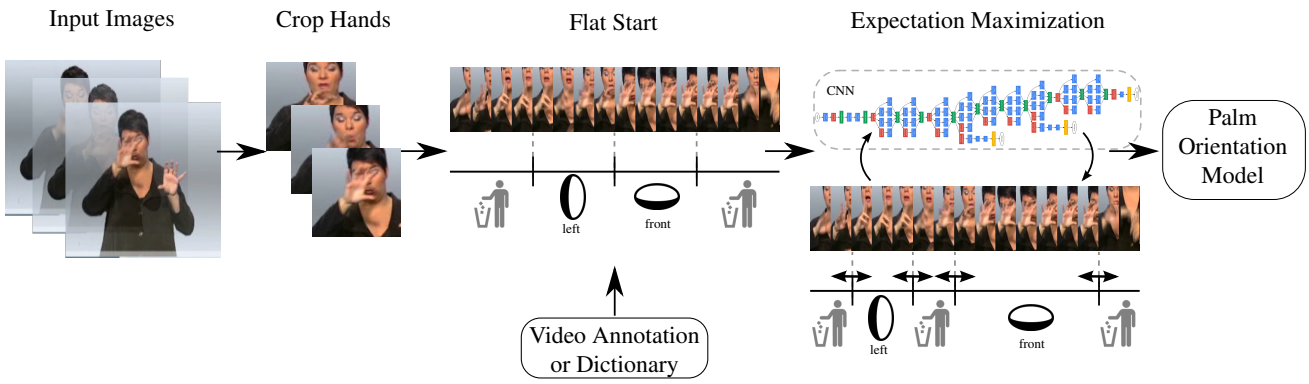

Figure 3: Overview of weakly supervised learning with HamNoSys subunits.

class), which we initialise with zeros. As a preprocessing step, we apply a global mean normalisation to the images prior to fine-tuning the CNN model with Stochastic Gradient Descent (SGD) and a softmax based cross-entropy classification loss.

4.4. Sign Language Recognition with Subunit Classifiers

In the previous subsections, we discussed learning a hand orientation classifier, based on available sign language lexicons with linguistic annotations. Continuous sign language recognition is the final task to be accomplished. However, suitable corpora (for machine learning), such as the RWTH-PHOENIX-Weather data set, do not provide subunit annotations. Therefore, we cannot apply the learned subunit classifiers directly, as there is no knowledge on how to break signs of the given corpus up into subunits.

A viable solution is to use the learned subunit classifiers as feature extractors and retrain a GMM system. This allows us to make use of the external subunit annotations (of corpora which are not intended for pattern recognition purpose) to improve the recognition on a given gloss annotated machine learning corpus, such as RWTH-PHOENIX-Weather.

The procedure is as follows: The hand orientation classifiers are trained to classify single images. During training, no sample from our target machine learning corpus was part of the training set. However, due to the CNN's ability to generalise, the unseen images forwarded through the trained network still provide good features for CSLR. With this work's experiments we will investigate if the classification results of the final softmax layer, the output scores of the last fully connected layer or the preceding last convolutional layers constitute the best features. We further evaluate how to preprocess these extracted features prior to modelling them in a standard HMM-GMM gloss-based CSLR system (Rybach et al., 2011). We compare no preprocessing to variance normalisation and dimensionality reduction by principal components analysis (PCA).

5. Experimental Results

We present the experimental evaluation in this section. In the first subsection, we focus on weakly supervised learning, whereas in the later subsection we apply the learnt subunit extractor to a state-of-the-art CSLR pipeline.

5.1. Weakly Supervised Subunit Learning

As described in the previous section, the task is to jointly estimate a good alignment for the noisy subunit labels and to model the given subunits robustly. The algorithm converges after a couple of iterations. For this work, we run it for 9 iterations. Figure 4 shows exemplar alignments of the palm orientation subunits in the initialising condition and after the last iteration. Looking at the initial alignments in the first and third line in Figure 4, we see that the majority of labels are already correctly aligned. However, at the positions where labels change, there are some alignment errors. After the convergence of the algorithm (row 2 and 4 in Figure 4), we see that all labels have been correctly aligned.

Figures 5 and 6 show the distribution of the aligned subunit classes across the nine iterations of weakly supervised learning for palm orientations and finger orientations respectively. We see that after a couple of iterations the palm orientations stabilise to four main orientation subunits (being 'left', 'front', 'down', 'back'). The training distribution of finger orientation subunits in Figure 6 look different. Here, the 'up' subunit dominates the others in terms of occurrence. Besides that, nine other finger orientations ('upleft', 'left', 'frontup', 'frontupleft', 'front', 'frontleft', 'down', 'downleft', 'backup') are less frequent in the data. This suggests that finger orientations are less stable than palm orientations. 'Stable' may refer to the production of sign language, to the annotation quality or to the modelling itself.

Within one iteration of weakly supervised learning, we continuously finetune the CNN model and measure the model's accuracy on a held out set (being 10% of the training data). We decide when to stop the CNN learning based on this accuracy. Figure 7 shows the correlation between the accuracy and the word error rate (WER). It has to be noted, however, that the WER is measured on a different data set and is therefore not directly comparable. We see a clear trend of increasing top-1 accuracy during the first training epoch (steps 1 to 8), from then onward the accuracy seems to oscillate a bit. The red WER on dev curve oscillates from the beginning, there is no clear trend visible. However, the green WER on test curve (lower is better) seems to con-

Figure 4: Palm orientation alignment visualisation. First and third rows show the sample alignments at the initialisation of the algorithm. Second and last rows show it after 9 iterations of the weakly supervised learning. It is visible how the learning helps to find a good frame alignment between the palm orientation subunits and the video footage. 'si' refers to the garbage class. Every fourth frame is shown.

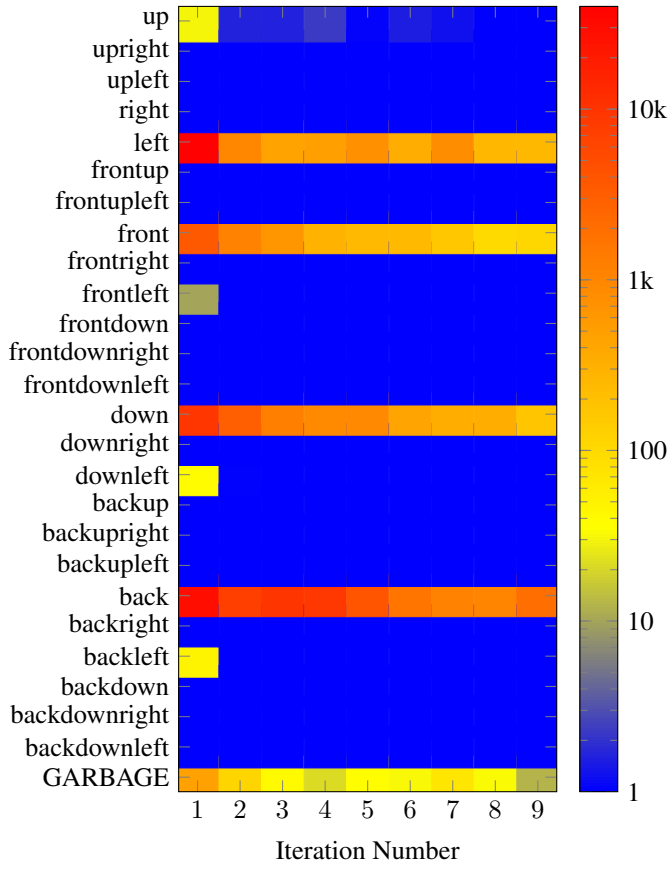

Figure 5: Figure shows unique training samples per palm orientation class as distribution across 9 iterations of the weakly supervised learning.

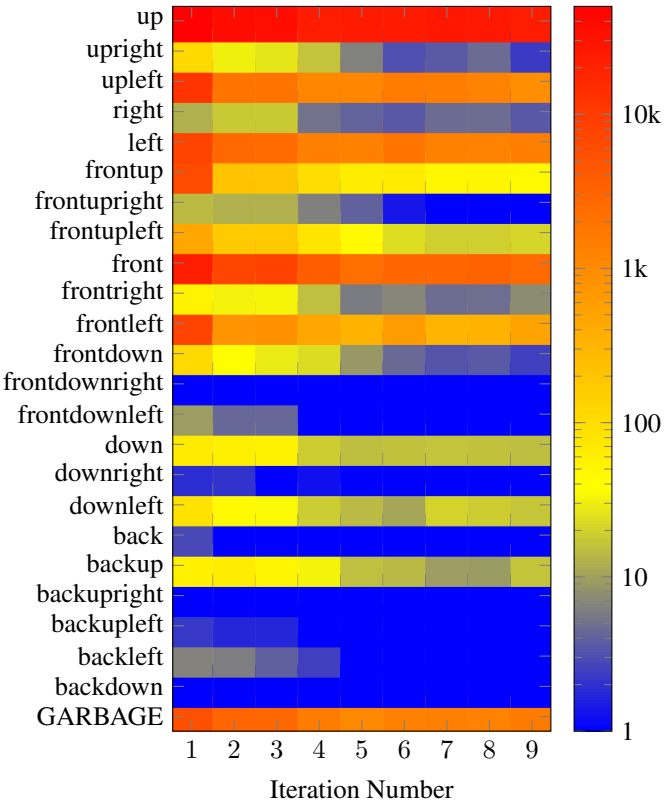

Figure 6: Figure shows unique training samples per finger orientation class as distribution across 9 iterations of the weakly supervised learning.

tinuously decrease with increasing epochs. Thus, it seems that the CSLR retraining and parameter tuning using extracted features on the development set allows us to fit the

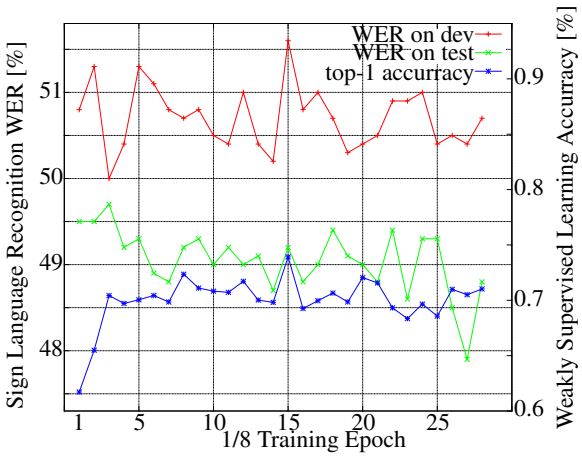

Figure 7: Correlation between WER measured on RWTH-PHOENIX-Weather test set (finger orientation + 1-Mio-Hands features) and accuracy 8 times per epoch of the weakly supervised finger orientation learning on 10% of the training data throughout the last iteration of the learning algorithm.

HMM-GMM model equally well to a worse subunit classifier. However, this then does not generalise to an unseen test set. It is good to see that with increasing CNN training this generalisation continuously improves (test WER is going down).

5.2. Continuous Sign Language Recognition

In this section we evaluate the trained hand orientation subunit classifiers integrated into a state-of-the-art continuous sign language recognition pipeline that predicts unseen sequences of RWTH-PHOENIX-Weather 2014 Multisigner. We employ the classifiers after 9 iterations of weakly supervised learning. As previously described, they have been trained solely with out-of-task (Danish and Swiss German) data. In the recognition pipeline, we use them as feature extractors, with a single input image from PHOENIX being forwarded through the CNN, which is composed of the different layers as mentioned in Section 4.3. The forward pass can be stopped at any of them and the output constitutes the extracted features. Table 2 compares the performance of using features originating from different layers of the trained subunit CNN. Not being trained on the task data directly (as no subunit annotations are available for PHOENIX), we expect the final classification output to be quite noisy. This assumption seems to hold, as the softmax features (lines 1-3 in Table 2) are all largely outperformed by the other layers. The last fully connected output (denoted as 'last FC' in Table 2) performs marginally better (when being variance normalised across the whole data set) than the PCA reduced output of the last convolutional layer (actually, the output of the last pooling layer, which has 1024 dimensions). For subsequent experiments, we perform recognition always with features originating from the last fully connected layer, which are then variance normalised.

We further analyse which CNN model initialisation scheme we should follow. Table 3 compares two different pre-training schemes, which either rely on the imagenet data set (over one million hand labelled objects from $1,000$ categories, very diverse in size, appearance and capture conditions) or on the 1-Mio-Hands model, which was trained to distinguish handshapes orientation independently using more than one million hand images from the Danish, New Zealand and German Sign Language (see (Koller et al., 2016) for details.) We denote that the 1-Mio-Hands model helps to learn a better stand-alone subunit classifier. However, when combined with the original initialisation model and applied to the CSLR task, it lacks complementary information. We therefore use Imagenet to pre-train our CNNs in all subsequent experiments.

	Extraction Layer					WER	
	Softmax	Last FC	Last Conv	Var Norm	Dim	Dev	Test
1	X			no	26	65.0	63.7
2	X			yes	26	63.9	63.4
3	X			yes	26pca	52.1	51.3
4		X		yes	26pca	51.9	50.3
5		X		no	26	50.4	49.4
6			X	yes	50pca	50.4	**48.2**
7		X		yes	26	**50.1**	48.3

Table 2: Comparing different feature extractor layers. All experiments represent the palm orientation in feature combination (stacking) with the 1-Mio-Hands classifier (Koller et al., 2016). 'Dim' stands for dimension, 'var norm' for variance normalisation, 'fc' for fully connected layer, 'conv' for convolutional layer. WER in [%].

	Initialisation		WER [%]	
	Imagenet	1-Mio-Hands	Dev	Test
1-Mio-Hands alone	X		51.6	50.2
Subunit alone		X	53.1	53.0
Subunit alone	X		72.9	72.4
Subunit + 1-Mio-Hands		X	50.2	50.3
Subunit + 1-Mio-Hands	X		50.8	49.6

Table 3: Impact of initialisation. An initialisation from a better model trained on the same data yields a better stand-alone classifier, but lacks complementary information in combination with the original initialisation model. Results on RWTH-PHOENIX-Weather 2014 Multisigner. 1-Mio-Hands have been presented in Koller et al. (2016). '+' denotes feature stacking prior to GMM-HMM training.

Table 4 compares the finger and the palm orientation classifiers and their combination. We see that the palm orientation outperforms the finger orientation and as expected both contain complementary information with respect to each other, as their fusion is clearly better than each classifier alone. Moreover, both orientation subunit classifiers add complementary information to the strong 1-Mio-Hand handshape baseline, which improves from $51.6\% \rightarrow 49.6\%$ on dev and from $50.2\% \rightarrow 48.2\%$ on test.

Table 5 shows how much complementary information the hand orientation classifiers add to a strong multi-modal baseline consisting of jointly modelled (stacked) features from Koller et al. (2015) (being HoG3D, right to left hand distance, movement/trajectory of dominant hand, place of

	Dev		Test	
	del/ins	WER	del/ins	WER
HoG-3D	25.8/4.2	60.9	23.2/4.1	58.1
1-Mio-Hands	19.1/4.1	51.6	17.5/4.5	50.2
Finger orientation	33.0/3.1	72.9	31.3/3.1	72.4
Palm orientation	25.4/4.1	68.7	24.4/4.5	66.9
Finger + Palm	26.3/3.3	63.8	24.3/3.3	62.3
Finger + 1-Mio-Hands	16.3/5.3	50.8	15.0/5.6	49.6
Palm + 1-Mio-Hands	17.5/4.6	50.1	16.0/4.6	48.3
Finger+Palm+1-Mio-Hands	17.5/4.7	**49.6**	15.9/4.6	**48.2**

Table 4: Hand-only continuous sign language recognition results on RWTH-PHOENIX-Weather 2014 Multisigner. 1-Mio-Hands have been presented in Koller et al. (2016). '+' denotes feature stacking prior to GMM-HMM training. WER in [%].

articulation normalised by the nose and facial features) and the 1-Mio-Hand handshape features from Koller et al. (2016). We note, that the orientation subunits can improve the result on the dev set, but the improvement does not carry over to the test set. Most likely, more data would be required to achieve a better generalisation. Including RWTH-PHOENIX-Weather into the subunit training stage may also boost results (cf. (Koller et al., 2016)).

		Dev		Test	
		del/ins	WER	del/ins	WER
1	(Koller et al., 2015) cmllr	21.8/3.9	55.0	20.3/4.5	53.0
2	(Koller et al., 2015)	23.6/4.0	57.3	23.1/4.4	55.6
3	+ 1-Mio-Hands	16.3/4.6	47.1	15.2/4.6	**45.1**
4	+ Finger + Palm	18.0/4.5	**46.6**	16.5/4.8	45.5

Table 5: Multi-modal continuous sign language recognition results on RWTH-PHOENIX-Weather 2014 Multisigner. 1-Mio-Hands have been presented in Koller et al. (2016). '+' denotes feature stacking prior to GMM-HMM training. WER in [%].

6. Conclusion

In this work, we presented our recent advances in the field of subunit modelling for continuous sign language recognition. We demonstrated how generic annotations at the articulator level, such as HamNoSys, can be exploited to learn subunit classifiers. We explored cross-language-subunits, which were trained on isolated signs of publicly available lexicon data sets for Swiss German and Danish Sign Language. We therefore employed a weakly supervised learning framework that helped to jointly find those subunits that occur in the data and to model them robustly.

We analysed the alignment of the weakly supervised learning, finding that palm orientations seem to be more stable than finger orientations. Furthermore, we systematically determine the best extraction scheme to include the learnt CNN as feature extractors in a GMM-HMM system. Finally, we evaluated palm orientation and finger orientation subunits to perform CSLR on the publicly available RWTH-PHOENIX-Weather corpus (Forster et al., 2014). We find that the modalities improve a handshape only system by 2% absolute WER, while still improving a multi-modal baseline (manual and non-manual features) by 0.5%.

7. Acknowledgements

We thank Penny Boyes Braem and Sarah Ebling for enabling the access to the Swiss German data which was crucial for this work. We further thank Thomas Troelsgård and Jette H. Kristoffersen for providing the employed Danish Sign Language annotations and videos.

Bauer, B. and Kraiss, K. F. (2002). Video-based sign recognition using self-organizing subunits. In *16th International Conference on Pattern Recognition, 2002. Proceedings*, volume 2, pages 434–437. IEEE.

Bourlard, H. A. and Morgan, N. (2012). *Connectionist speech recognition: a hybrid approach*, volume 247. Springer Science & Business Media.

Braem, P. B. (2001). A multimedia bilingual database for the lexicon of Swiss German Sign Language. *Sign Language & Linguistics*, 4(1/2):133–143.

Cooper, H., Ong, E.-J., Pugeault, N., and Bowden, R. (2012). Sign language recognition using sub-units. *The Journal of Machine Learning Research*, 13(1):2205–2231.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38.

Finish Association of the Deaf. (2015). Suvi Viittomat. `http://suvi.viittomat.net/`.

Forster, J., Schmidt, C., Koller, O., Bellgardt, M., and Ney, H. (2014). Extensions of the Sign Language Recognition and Translation Corpus RWTH-PHOENIX-Weather. In *Language Resources and Evaluation*, pages 1911–1916, Reykjavik, Island, May.

Glauert, J. and Elliott, R. (2011). Extending the SiGML notation—a progress report. In *Second International Workshop on Sign Language Translation and Avatar Technology (SLTAT)*, volume 23.

Jette H. Kristoffersen, Thomas Troelsgård, Anne Skov Hardell, Bo Hardell, Janne Boye Niemelä, Jørgen Sandholt, and Maja Toft. (2008-2016). Ordbog over Dansk Tegnsprog. `http://www.tegnsprog.dk/`.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T. (2014). Caffe: Convolutional Architecture for Fast Feature Embedding. *arXiv preprint arXiv:1408.5093*.

Kadir, T., Bowden, R., Ong, E.-J., and Zisserman, A. (2004). Minimal Training, Large Lexicon, Unconstrained Sign Language Recognition. In *BMVC*, pages 1–10. 00079.

Koller, O., Ney, H., and Bowden, R. (2013). May the Force be with you: Force-Aligned SignWriting for Automatic Subunit Annotation of Corpora. In *IEEE International Conference on Automatic Face and Gesture Recognition*, pages 1–6, Shanghai, PRC, April.

Koller, O., Ney, H., and Bowden, R. (2014). Read My Lips: Continuous Signer Independent Weakly Supervised Viseme Recognition. In *Proceedings of the 13th*

European Conference on Computer Vision, pages 281–296, Zurich, Switzerland, September.

Koller, O., Forster, J., and Ney, H. (2015). Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers. *Computer Vision and Image Understanding*, 141:108–125, December.

Koller, O., Ney, H., and Bowden, R. (2016). Deep Hand: How to Train a CNN on 1 Million Hand Images When Your Data Is Continuous and Weakly Labelled. In *IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, June.

Kong, W. W. and Ranganath, S. (2014). Towards subject independent continuous sign language recognition: A segment and merge approach. *Pattern Recognition*, 47(3):1294–1308, March.

McKee, D., McKee, R., Alexander, S. P., and Pivac, L. (2015). The Online Dictionary of New Zealand Sign Language. `http://nzsl.vuw.ac.nz/`.

Pitsikalis, V., Theodorakis, S., Vogler, C., and Maragos, P. (2011). Advances in phonetics-based sub-unit modeling for transcription alignment and sign language recognition. In *2011 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1–6, June.

Prillwitz, S., Leven, R., Zienert, H., Zienert, R., and Hanke, T. (1989). *HamNoSys. Version 2.0*. Int'l Studies on Sign Language and Communication of the Deaf. Signum, Hamburg.

Rybach, D., Hahn, S., Lehnen, P., Nolden, D., Sundermeyer, M., Tüske, Z., Wiesler, S., Schlüter, R., and Ney, H. (2011). RASR - The RWTH Aachen University Open Source Speech Recognition Toolkit. In *IEEE Automatic Speech Recognition and Understanding Workshop*, Waikoloa, HI, USA, December.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2014). Going Deeper with Convolutions. *arXiv:1409.4842 [cs]*, September.

Tamura, S. and Kawasaki, S. (1988). Recognition of sign language motion images. *Pattern Recognition*, 21(4):343–353.

Theodorakis, S., Pitsikalis, V., and Maragos, P. (2014). Dynamic–static unsupervised sequentiality, statistical subunits and lexicon for sign language recognition. *Image and Vision Computing*, 32(8):533–549. 00003.

Vogler, C. and Metaxas, D. (1999). Toward scalability in ASL recognition: Breaking down signs into phonemes. *Gesture-Based Communication in Human-Computer Interaction*, pages 211–224.

Waldron, M. B. and Kim, S. (1995). Isolated ASL sign recognition system for deaf persons. *IEEE Transactions on Rehabilitation Engineering*, pages 261–271.

Yin, F., Chai, X., Zhou, Y., and Chen, X. (2015). Semantics constrained dictionary learning for signer-independent sign language recognition. In *Image Processing (ICIP)*, pages 3310–3314. IEEE.