IJCV - SI: BMVC 2016 manuscript No.
(will be inserted by the editor)

Deep Sign: Enabling Robust Statistical Continuous Sign Language Recognition via Hybrid CNN-HMMs

Oscar Koller · Sepehr Zargaran · Hermann Ney · Richard Bowden

Received: date / Accepted: date

Abstract This manuscript introduces the end-to-end embedding of a CNN into a HMM, while interpreting the outputs of the CNN in a Bayesian framework. The hybrid CNN-HMM combines the strong discriminative abilities of CNNs with the sequence modelling capabilities of HMMs. Most current approaches in the field of gesture and sign language recognition disregard the necessity of dealing with sequence data both for training and evaluation. With our presented end-to-end embedding we are able to improve over the state-of-the-art on three challenging benchmark continuous sign language recognition tasks by between 15% and 38% relative reduction in word error rate and up to 20% absolute. We analyse the effect of the CNN structure, network pretraining and number of hidden states. We compare the hybrid modelling to a tandem approach and evaluate the gain of model combination.

Keywords Sign Language Recognition · Hybrid Approach · CNN-HMM · Statistical Approach · Sequence Modelling

1 Introduction

Face-to-face communication is often the preferred choice, when either important matters need to be discussed or informal links between individuals are established. Gesture is a key part in such human-to-human communication. It helps us to better understand the other party. However, the role of visual cues in spoken language is not well defined. As such, the task of gesture recognition is also not accurately defined. This renders comparison of algorithms and approaches difficult. Sign language on the other hand provides a clear framework with a defined inventory and grammatical rules that govern joint expression by hand (movement, shape, orientation, place of articulation) and by face (eye gaze, eye brows, mouth, head orientation). This makes sign languages, the natural languages of the deaf, a perfect test bed for computer vision and human language modelling algorithms targeting human computer interaction and gesture recognition. The rules governing the interaction of hands and body, referred to as the manual and non-manual parts - are well defined by sign language theory. Videos represent a time series of dynamic images and the recognition of sign language therefore needs to be able to cope with variable input sequences and execution speed. Different schemes are followed to achieve this ranging from sliding window approaches (Ong et al 2014) to temporal normalisations (Molchanov et al 2015) or dynamic time warping (Krishnan and Sarkar 2015). While the field of automatic speech recognition is dominated by Hidden-Markov-Models (HMMs), they remain rather unpopular in computer vision related tasks. For instance CVPR, by many regarded as the top conference of computer vision, had only three out of a total of over 700 submissions in the year 2017 that were using HMMs (Koller et al 2017; Richard et al 2017; Schober et al 2017). This may be related to the comparatively poor image modelling capabilities of Gaussian Mixture Models (GMMs), which had been traditionally used to model the observation probabilities within such a framework. More recently, deep Convolutional Neural Networks (CNNs) have outperformed other approaches in all computer vision tasks. In this work, we focus on integrating CNNs in a

O. Koller , S. Zargaran, H. Ney
Human Language Technology and Pattern Recognition
RWTH Aachen University
Aachen, Germany
E-mail: koller@cs.rwth-aachen.de

R. Bowden
Centre for Vision Speech and Signal Processing
University of Surrey
Guildford, UK

HMM framework, extending an interesting line of work (Koller et al 2015b; Le et al 2015; Wu and Shao 2014; Koller et al 2016a), which we will discuss more closely in Section 2.

This manuscript presents the extended version of our previous work (Koller et al 2016b), where we first presented a powerful embedding of a deep CNN in a HMM framework in the context of sign language and gesture recognition, while treating the outputs of the CNN as true Bayesian posteriors and training the system as a hybrid CNN-HMM in an end-to-end fashion. With this method we are able to achieve a large relative improvement of over 15% compared to the state-of-the-art on three challenging standard benchmark continuous sign language recognition data sets. In the scope of this extended manuscript, we make several additional contributions and have completely reran all experimental evaluation to allow us to provide more extensive results and deeper insights:

1. We significantly add to the theoretical explanation of the hybrid approach, with the aim of making its idea more accessible to newcomers to the field.
2. We analyse the effect of both CNN- and HMM-structure on the hybrid approach.
3. We investigate the effect of using out-of-domain data to train the network prior to finetuning using in-domain data.
4. We show that different training iterations provide complementary classifiers, which are able to further boost recognition when employed as ensembles of hybrid CNN-HMMs.

The rest of this manuscript is organised as follows: Section 2 discusses the related literature in depth. In Section 3 we introduce the theoretical basis of the presented hybrid approach. Differences w.r.t. the tandem approach are also described. The employed data sets are discussed in Section 4. Section 5 gives details on the implementation in order to ensure reproducibility, which is followed by the actual experimental evaluation in Section 6. Finally, we conclude the work in Section 7.

2 Related Work

After the recent success of CNNs (LeCun et al 1998) in many computer vision fields, they have also shown large improvements in gesture and sign language recognition (Neverova et al 2014; Huang et al 2015; Koller et al 2015b). However, in most previous CNN-based approaches the temporal domain of video data is not elegantly taken into consideration. Most approaches simply use a sliding window or circumvent the sequence properties by evaluating the output in terms of per-frame overlap with the ground truth, *e.g.* in Pigou et al (2018). Moreover, CNNs are usually trained on the frame-level. A few artificial data sets such as the Montalbano gesture data set (Escalera et al 2014) provide frame labels. However, this is usually not the case, especially for sign language footage or other real-life data sets. Available annotation usually consists of sequences of signs without explicit frame-level information. As such, the focus of the field needs to move towards approaches that deal with variable length inputs and outputs that do not require explicit frame labelling. The difficulty in accurately labelling single frames for evaluation further supports the need for such change. Graphical models such as HMMs lend themselves well to tasks with inputs of variable length. As will be shown in this work, we are able to combine the best of different worlds when integrating HMMs and CNNs. A few works have joined neural networks and HMMs before in the scope of gesture and sign language recognition. Wu and Shao (2014) use 3D CNNs to model the observation probabilities in a HMM. However, they interpret the CNN outputs as likelihoods $p(x|k)$ for an image x and a given class k. Conversely, Richard and Lippmann (1991) showed that neural network outputs are better interpreted as posterior probabilities $p(k|x)$ in a Bayesian framework. In the field of speech recognition, Bayesian hybrid neural network HMMs were first proposed by Bourlard and Wellekens (1989) and became the approach of choice, particularly after the recent rise of deep learning. Le et al (2015) followed this line of thought for gesture recognition, but only employed a shallow legacy neural network that was trained to distinguish twelve artificial actions. Koller et al (2013) and Koller et al (2014) achieved important results using GMM-HMMs for weakly supervised learning in the domain of sign language. However, hybrid models strongly outperformed the results (Koller et al 2016a), which constituted first and preliminary work in this direction. CNNs were employed in a hybrid Bayesian framework to perform weakly supervised training with the purpose of learning hand shape classifiers that generalise across data sets. The main differences with respect to this manuscript are that we learn the CNN top down using nothing more than the annotated sign-words (which are modelled by a fixed number of hidden states), whereas Koller et al (2016a) models signs bottom up with additional knowledge of the decomposition of sign-words into different hand shapes which form the building blocks for signs. Moreover, in this work, we learn the CNN-HMM in an end-to-end fashion from video input to gloss output, whereas in the previous work, the intermediate hand shape-CNN serves as feature extractor for an additional GMM-HMM sign model (similar to the tandem approach introduced in Section 3.3). In this so-called tandem modelling (refer to Section 3.3), the GMM-HMM needs to be completely retrained, which adds significant computational overhead. In the proposed

hybrid approach, no GMM-retraining is necessary and in the experimental evaluation of this manuscript we will show that our approach clearly outperforms Koller et al (2016a). Wu et al (2016) published a paper that is also closely related to this work, but they do not interprete the CNN outputs in a Bayesian way, they use different inputs to the CNN (full body RGB and depth, as opposed to using a cropped hand patch) and different inputs to the HMM. Later, Granger and el Yacoubi (2017) provided a comparison between a hybrid neural network HMMs and a recurrent neural network (RNN) on a gesture task, finding that both perform comparably, while the state-based representation of the HMM allows better insights in the internals of the model for potential error analysis. Recently, Connectionist Temporal Classification (CTC) by Graves and Schmidhuber (2005) has received attention by the computer vision community in general (Assael et al 2016; Cui et al 2017; Rao et al 2017). CTC is a training criterion for recurrent neural networks and very related to HMMs. CTC has been shown to be a special case of the hybrid full-sum HMM alignment with a specific HMM architecture. As such CTCs are related to this work. However, we do not use recurrent or long short term memory (LSTM) networks in this work. The interested reader may consult Bluche et al (2015) for details on the comparison of CTC and HMMs.

Finally, this manuscript represents a more thorough version of Koller et al (2016b), with much more extensive experiments. In addition, this manuscript analyses the effect of both CNN- and HMM-structure on the hybrid approach. It also investigates the effect of using out-of-domain data to pretrain the network prior to finetuning using in-domain data and the use of ensembles of CNN-HMMs in model combination to further boost performance. Koller et al (2017) even drop the dependence on a hand tracking system and take the re-alignment of hybrid models for sign recognition further.

Another related approach has been introduced by Bengio and Frasconi (1996), where a RNN is used to extract temporally local information whereas a HMM integrates long-term constraints. The so-called input output HMM has been used by Marcel et al (2000) in a basic gesture system that distinguishes between two gesture classes, deictic and symbolic.

3 Continuous Sign Language Recognition

The problem to be solved is a sequence learning task, which means we want to predict a sequence of output symbols w_1^N, in our case sign words (so-called "glosses", representing the semantics of the described word). Given an input video as a sequence of full images $X_1^T = X_1, \ldots, X_T$ and the resulting preprocessed (*e.g.* tracked and mean-normalised) images $x_1^T = x_1, \ldots, x_T$, automatic continuous sign language recognition tries to find an unknown sequence of sign-words w_1^N for which x_1^T best fit the learned models. We assume that images and sign-words occur in an ordered fashion. It has to be noted that this requirement clearly distinguishes the problem of sign language recognition from the problem of translating from sign language to spoken language where re-orderings are necessary and monotonicity cannot be assumed.

3.1 Legacy GMM-HMM Approach

To find the best fitting sequence, we follow the statistical paradigm (Bahl et al 1983) using the maximum-a-posteriori simplification of Bayes' decision rule, which has been successfully applied to Automatic Speech Recognition (ASR), hand writing recognition and statistical machine translation since the early 1970s. Given a loss function $\mathcal{L}\left[w_1^N, \tilde{w}_1^N\right]$ between the true output sequence w_1^N and the hypothesised output sequence $\tilde{w}_1^N$, Bayes' Decision Rule minimises the expected loss:

$$x_1^T \rightarrow \left[w_1^N\right]_{\text{opt}} = \underset{\tilde{w}_1^N}{\text{argmin}} \left\{ \sum_{w_1^N} Pr\left(w_1^N | x_1^T\right) \cdot \mathcal{L}\left[w_1^N, \tilde{w}_1^N\right] \right\} \quad (1)$$

Often Bayes Decision Rule is simplified to the maximum-a-posteriori (MAP) rule, which is known to be equivalent for the case of the simple 0-1-loss.

$$x_1^T \rightarrow \left[w_1^N\right]_{\text{opt}} = \underset{w_1^N}{\text{argmax}} \left\{ Pr\left(w_1^N | x_1^T\right) \right\} \quad (2)$$

In sign language recognition the 0-1-loss corresponds to a minimisation of the expected sentence error rate, which counts an output sentence as wrong if a single recognised sign-word is wrong. However, for longer sentences, the sentence error rate does not correlate with the word error rate (WER) which is also known as edit distance and what we seek to minimise. As shown by Schlüter et al (2012), the MAP rule is equivalent to the Bayes Rule for the WER as a loss function if

$$\max_{w_1^N} \left\{ Pr\left(w_1^N | x_1^N\right) \right\} > 0.5 \quad (3)$$

Therefore, we follow the MAP rule as the optimisation criterion and maximise the class posterior probability distribution $Pr(w_1^N | x_1^T)$ over the whole utterance, as given in Equation 2.

Decision theory allows us to split up the class posterior probability into the class prior $Pr(w_1^N)$ and the class-conditional probability $Pr(x_1^T | w_1^N)$, which can then be modelled by different information sources. The first term can be interpreted as word sequence knowledge which can be approximated by a n-gram language model estimating

$p(w_1^N)$. The second term represents the actual visual knowledge, which historically used to be modelled by generative GMMs.

$$\left[w_1^N\right]_{\text{opt}} = \underset{w_1^N}{\text{argmax}} \left\{ p\left(w_1^N\right) \cdot p\left(x_1^T | w_1^N\right) \right\} \tag{4}$$

Expressing the class-conditional probability in terms of a HMM adds the hidden variable s_1^T:

$$p\left(x_1^T | w_1^N\right) = \sum_{s_1^T} p\left(x_1^T, s_1^T | w_1^N\right) \tag{5}$$

$$= \sum_{s_1^T} \prod_{t=1}^{T} p\left(x_t, s_t | x_1^{t-1}, s_1^{t-1}, w_1^N\right) \tag{6}$$

$$= \sum_{s_1^T} \prod_{t=1}^{T} p\left(x_t | x_1^{t-1}, s_1^t, w_1^N\right) \cdot p\left(s_t | x_1^{t-1}, s_1^{t-1}, w_1^N\right) \tag{7}$$

$$= \sum_{s_1^T} \prod_{t=1}^{T} p\left(x_t | s_t, w_1^N\right) \cdot p\left(s_t | s_{t-1}, w_1^N\right) \tag{8}$$

where the sum in Equation 5 expresses all viable paths that lead to the same output sequence w_1^N. Equation 6 and 7 constitute reformulations with help of the chain rule. Assuming s to be non-observable and a first order Markov process leads to Equation 8. After applying the viterbi approximation, which considers only the most likely path and substituting everything into Equation 4, we get:

$$\left[w_1^N\right]_{\text{opt}} = \tag{9}$$
$$\underset{w_1^N}{\text{argmax}} \left\{ p\left(w_1^N\right) \cdot \max_{s_1^T} \left\{ \prod_{t=1}^{T} p\left(x_t|, s_t, w_1^N\right) \cdot p\left(s_t | s_{t-1}, w_1^N\right) \right\} \right\}$$

where in the legacy Gaussian mixture model (GMM)-hidden Markov model (HMM) approach for sign language recognition $p\left(x_t|, s_t, w_1^N\right)$ has typically been modelled as

$$p\left(x_t|, s_t, w_1^N\right) = \sum_{m=1}^{M} c_m \cdot \mathcal{N}(x, \mu_m, \Sigma) \tag{10}$$

$$\sum_{m=1}^{M} c_m = 1 \tag{11}$$

where $\mathcal{N}(x, \mu, \Sigma)$ is a multi-variate Gaussian with mean μ, covariance matrix Σ and M is the number of mixture components (can differ between states of the same word). Legacy systems typically employed a globally pooled covariance matrix Σ to cope with the low amount of training samples per state and word. The expectation maximization (EM) algorithm is used to estimate the sufficient statistics of the GMMs. The number of EM iterations is usually optimised on held out data during the training phase of the system.

$p(s_t|s_{t-1}, w_1^N)$ (referring to Eq. 9) represents the state transition model, which is empirically known as part of the model having limited impact on the final result and can therefore be pooled across all HMM states. In log-domain we often refer to it as the Time Distortion Penalties (TDPs). The dependency on the sequence of words w_1^N may be dropped, since the temporal sequence of states s_1^T is defined to be a sequence of HMM states corresponding to a specific path through the word sequence w_1^N, which we implement as concatenation of automatons for w_1^N (using the word-to-state decomposition defined by the pronunciation lexicon and the word sequence annotations of the corpus).

3.2 Hybrid CNN-HMM Approach

Up to this point, we have deduced the standard HMM formula for recognition using a generative model for the emission probability. However, in the scope of the presented work we model the emission probability of the HMM $p(x_t|s_t, w_1^N)$ by an embedded CNN, which is known to possess much more powerful image modelling capabilities than generative models such as GMMs. However, as pointed out by Richard and Lippmann (1991) and Bourlard and Morgan (1993), the CNN is a discriminative model whose outputs are estimates of the posterior probability and therefore cannot directly be inserted in the optimisation formula. Inspired by the hybrid approach known from ASR (Bourlard and Morgan 1993), we use Bayes' rule to convert the posterior probability of the CNN to a likelihood. For easier understanding we introduce the sub-word label $\alpha := s, w_1^N$, representing the state s belonging to the word sequence w_1^N. The CNN will hence be trained to model $p(\alpha|x_t)$. We apply Bayesian inference, converting the posteriors to class-conditional likelihoods following Bayes' rule:

$$p\left(x_t|\alpha\right) = p\left(x_t\right) \cdot \frac{p\left(\alpha|x_t\right)}{p\left(\alpha\right)} \tag{12}$$

where the prior probability $p(\alpha)$ can be approximated by the relative state label frequencies in the frame-state-alignment used to train the CNN.

For practical usage, we add several hyper-parameters to the implementation. These allow us to control the effect of the language model (γ) and the CNN label prior (β). Neglecting the constant frame prior $p\left(x_t\right)$, we finally optimise the following equation to find the best output sequence:

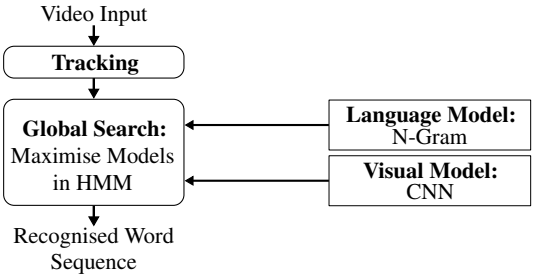

Fig. 1 Overview of the proposed CNN-HMM hybrid approach for continuous sign language recognition.

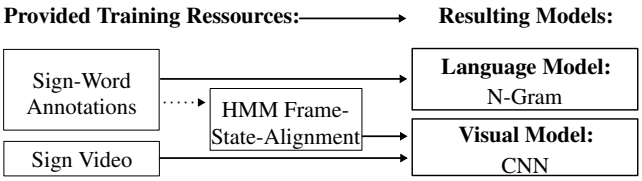

Fig. 2 Showing the employed resources (in light boxes on the left) to train the models for the hybrid CNN-HMM approach. The frame-state-alignment has been generated from the sign-word (gloss-) annotations using a GMM-HMM system from Koller et al (2016a).

$$\left[w_1^N\right]_{\text{opt}} = \underset{w}{\operatorname{argmax}}\left\{p\left(w_1^N\right)^{\gamma} \cdot \max_{s_1^T}\left\{\prod_{t=1}^{T} \frac{p\left(\alpha|x_t\right)}{p\left(\alpha\right)^{\beta}} \cdot p\left(s_t|s_{t-1}, w_1^N\right)\right\}\right\} \tag{13}$$

A general overview of the proposed hybrid CNN-HMM algorithm for recognition can be found in Figure 1. The hybrid approach has the positive property that during training only the CNN and the language model (LM) need to be retrained, while the HMM requires no training. For testing, the best hyper parameter values for γ, β and the pooled state transition model $p(s_t|s_{t-1}, w_1^N)$ are found using a grid search.

Figure 2 summarises the resources we need to successfully apply the hybrid approach: a dual corpus of sign videos (sentence-wise segmented) and corresponding sign-word annotations. In this work, we further employ the HMM frame-state-alignment coming from a HMM-GMM system as frame labelling, which can be replaced by an appropriate re-alignment scheme as shown in Koller et al (2017).

3.3 Tandem Approach

An intermediate step between GMM-HMM and the hybrid CNN-HMM systems is the so-called tandem approach. It is very similar to the hybrid approach in the sense that it uses both a CNN and HMM. However, the CNN is not used as a classifier, but rather as a feature extractor. In the so-called tandem approach (Hermansky et al 2000) the activations of a fully connected layer or the feature maps of a convolutional layer are dumped, post-processed (Koller et al 2016a) and then modelled in a GMM-HMM framework. This creates a significantly higher computational cost than the hybrid approach for extracting features and retraining a GMM system. Golik et al (2013) found that in speech and handwriting recognition the hybrid approach shows equal or superior performance compared to the tandem approach. We will verify this statement for sign language recognition in Section 6.3. As discussed in Section 2, in the gesture and sign language recognition literature to date, most other works either use the CNN outputs not in a Bayesian interpretation (Wu et al 2016) or employ the CNN as feature extractor comparable to the tandem approach. Figure 3 shows the tandem and the hybrid approach side by side. We denote that the only difference is the visual model.

Table 1 Key statistics of the employed data sets. OOV stands for Out-Of-Vocabulary, *e.g.* words that occur in test, but not in train. Dev refers to the development set.

	PHOENIX 2012		PHOENIX 2014			SIGNUM	
	Train	Test	Train	Dev	Test	Train	Test
# signers	1	1	9	9	9	1	1
hours	0.51	0.07	8.88	0.84	0.99	3.86	1.06
frames	46282	6751	799006	75186	89472	416620	114230
∼still frames	-	-	20%	-	-	38%	-
running words	3309	487	65227	5540	6504	11127	2805
∅ frames/word	14.0	-	9.8	-	-	23.2	
vocabulary	266	-	1080	-	-	465	-
OOVs running	-	8	-	28	35	-	9
OOVs [%]	-	1.6	-	0.5	0.5	-	0.3

4 Data Sets

The experiments are carried out on three state-of-the-art continuous sign language data sets that have been used extensively to compare recent methods for continuous sign language recognition: RWTH-PHOENIX-Weather 2012, RWTH-PHOENIX-Weather 2014 and SIGNUM. Here, we provide an essential summary and some additional statistics on the word-class distributions. However, for further details on the data sets, the interested reader is directed to Koller et al (2015a).

Single images of the corpora are depicted in Figure 4. Brief statistics on the three data sets can be found in Table 1. Both RWTH-PHOENIX-Weather corpora (2012 and 2014) were first introduced by Forster et al (2012) and Forster et al (2014) and represent direct recordings of the broadcast news, being limited to the weather forecast domain. As such, the data can be regarded as challenging real-life footage covering most difficulties you would expect from natural data (motion blur, transmission artifacts, fast signing, incomplete sentences, mis-signed words, interpretation errors, different

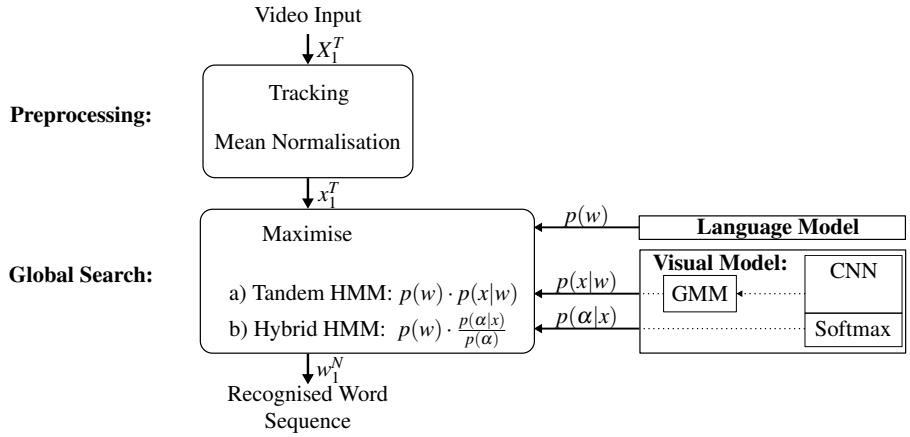

Fig. 3 Illustrating the difference between CNN-HMM tandem and hybrid approach. The former uses the CNN only as a feature extractor to train a subsequent GMM, while the later directly uses the CNN's normalised posteriors probabilities for a label α given input x.

Fig. 4 Example images showing the data sets employed in this work. RWTH-PHOENIX-Weather on the left and SIGNUM on the right.

clothing, *etc*.). RWTH-PHOENIX-Weather 2012 features a single signer interpreting the news into sign language, while RWTH-PHOENIX-Weather 2014 contains nine individuals covering varying amounts of the recorded programs.

SIGNUM was first introduced by von Agris et al (2008b) and was recorded in a laboratory environment while carefully controlling the signing and recording conditions. However, deviations from word counts in Table 1 w.r.t. previous work are errata, while the underlying data has not changed. All data sets feature user-dependent setups as all individuals occur both in the training and in the test/development (dev) partitions. The RWTH-PHOENIX-Weather is freely available[1]. It has to be noted that the actual annotation of PHOENIX 2014 and SIGNUM cover a larger variety of words than what the actual testing regime foresees. Therefore both data sets provide some mapping in order to join certain classes. This mainly arises due to the difficulty of the gloss annotation and manifests itself partly in inflected forms of the same words and in different words that are visually identical or very close. All referenced publications that report results on the data sets have been applying this simplification scheme, which is distributed with the data. The final number of classes that are distinguished in evaluation (see row 'vocabulary' in Table 1) is 266, 1080 and 465 for PHOENIX 2012, PHOENIX 2014 and SIGNUM respectively. On SIGNUM the vocabulary is 10 words larger than the reported vocabulary by the authors (von Agris et al 2008a). It is unclear what the cause for this is. Unfortunately the original authors cannot be reached anymore. The still frames in Table 1 refer to frames that have been automatically labelled as background during the HMM alignment.

Figures 5, 6 and 7 show the distribution of word counts on PHOENIX 2012, PHOENIX 2014 and SIGNUM respectively. It can be seen that both PHOENIX 2012 and 2014 contain a large number of words with only a single occurrence during training (so-called singletons), while SIGNUM statistics are different. On SIGNUM even the least frequent words occur at least 3 times, while most of them can be found at least 10 times in the training data. This is good for training and demonstrates SIGNUM's artificial characteristic which (among other reasons) manifests itself in very low WERs.

5 Implementation Details

In this section, we describe the details to allow exact reproducibility of our experiments. Note, that we input single (still) frames to the CNN and the HMM covers the temporal

[1] It can be obtained at http://www-i6.informatik.rwth-aachen.de/~koller/RWTH-PHOENIX/

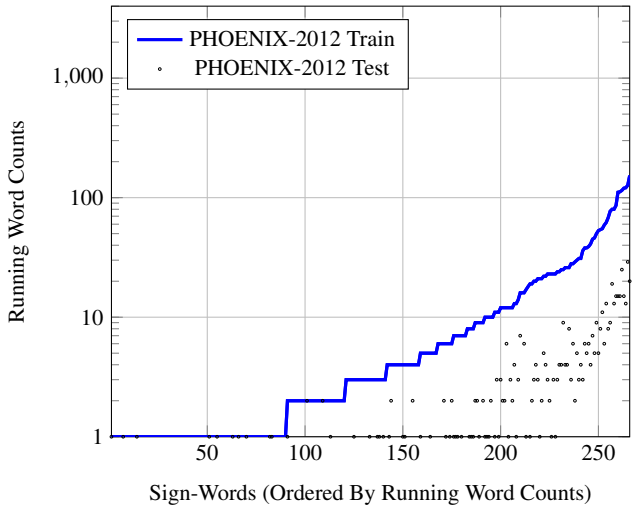

Fig. 5 Showing the distribution of words (and their counts) on the train and test partition of the RWTH-PHOENIX-Weather 2012 corpus. It can be seen that there are less than 100 sign-words occurring just a single time (singletons) imposing difficulties on the task.

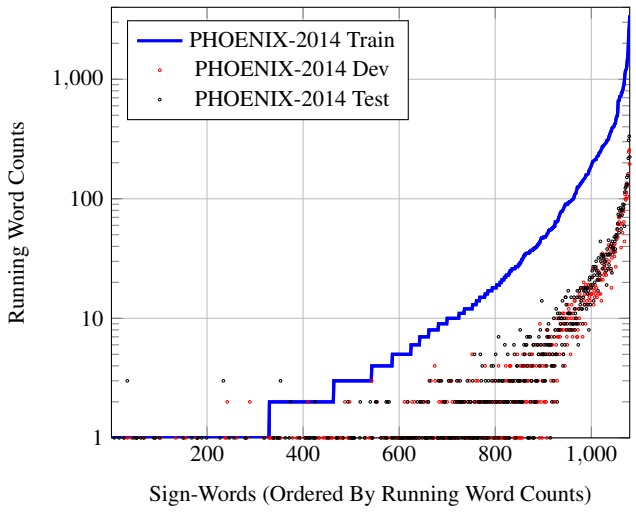

Fig. 6 Showing the distribution of words (and their counts) on the train, dev and test partition of the RWTH-PHOENIX-Weather 2014 corpus. It can be seen that there are more than 300 sign-words occurring just a single time (singletons) while few other classes occur more than 1000 times imposing difficulties on the task. Dev refers to the development set.

modeling. Input frames are cropped hand images. The system has no explicit information on the location other than from the background of the cropped images.

Image preprocessing. To track the right hand across all sequences of images we use a dynamic programming based approach (Dreuw et al 2006). In all data sets the right hand corresponds to the signer's dominant hand, which is the hand that plays the principle role in signing. On the RWTH-PHOENIX-Weather corpora, we crop a rectangle of 92x132 pixel around the centre of the hand. The original images suffer a constant distortion due to the broadcast nature of the videos, which corresponds to a scaling of the image width by a factor of 0.7. To compensate for this distortion we enlarge the cropped rectangles to the square size of 256x256. On SIGNUM we directly crop a square patch of size 100x100 pixel and scale it up to 256x256. Thereafter the pixel-wise mean of all images in the training set is subtracted from each image. Finally, for data augmentation we follow an online cropping scheme, which randomly crop out a 224x224 (GoogLeNet) or a 227x227 pixel (LeNet and AlexNet) rectangle to match the size of images in our model which was pre-trained on ImageNet. The input to the CNNs consists of single cropped hand patches.

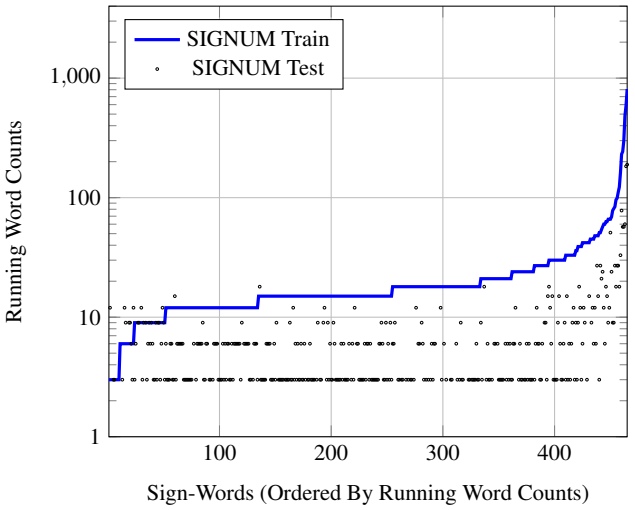

Fig. 7 Showing the distribution of words (and their counts) on the train, dev and test partition of the SIGNUM single signer corpus. In large contrast to the RWTH-PHOENIX-Weather corpora, it can be seen that hardly any sign-words occur just a single time (singletons). This shows the artificial nature of the data set and explains its comparative easiness.

Convolutional Neural Network Training. We base our CNN implementation on Jia et al (2014), which uses the NVIDIA CUDA Deep Neural Network GPU-accelerated library. If not stated otherwise in the respective experiments, we opted for the GoogLeNet Szegedy et al (2015) 22 layers deep CNN architecture with around 15 million parameters (for exact parameters refer to Table 3). GoogLeNet has shown many times in the past, most notably in the ImageNet 2014 (ILSVRC) Challenge, that it can be quite effective in combining impressive performance with minimal computational resources. Much of the improvements in this architecture compared to others' stems from the inception module which combines filters of different sizes after applying dimensionality reduction through a 1x1 Convolutional layer. The employed CNN architecture includes 3 classifying layers, meaning that besides the final classifier the network also includes two intermediary auxiliary classifiers. Those encourage discrimination in earlier layers of the network. The loss of these auxiliary classifiers is added to the total loss

with a weight of 0.3. All non-linearities are rectified linear units and each classifier layer is preceded by a dropout layer. We use a dropout rate of 0.7 for the auxiliary layers and 0.4 before the final classifier.

As mentioned in Section 3, the CNN training scheme requires an initial frame-state-alignment. This originates from a GMM-HMM recognition system that is trained to re-aligning the frame-to-state mapping (frame-level alignment). This is illustrated in Figure 8. If not stated

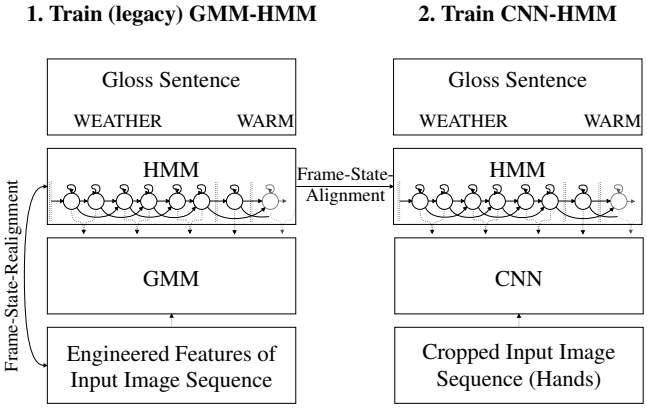

Fig. 8 Illustration of Training Scheme.

otherwise in the respective experiments, we use alignments from GMM-HMM systems reproducing the best published results on our chosen corpora. For SIGNUM and RWTH-PHOENIX-Weather 2014 we use the best results published in Koller et al (2016a) as alignment, whereas for RWTH-PHOENIX-Weather 2012 we use Koller et al (2015a). We split the frame-level alignment into a training ($\sim$ 90% of the data) and a validation set ($\sim$ 10% of the data) in order to be able to evaluate the per-frame accuracy of the CNN and stop the training when the validation accuracy deteriorates. However, we noticed that this seldom happened and in these experiments we always chose the last training iteration. We first train the network on the ImageNet data set with 1.2 million high-resolution images in 1000 classes and then exchange the final classification layers (on all three classifiers) and finetune the network on the sign language data for 80000 iterations with a mini-batch size of 32 images. We use stochastic gradient descent with an initial learning rate $\lambda_0 = 0.01$ for CNN networks. We employ a polynomial scheme to decrease the learning rate λ_i for iteration i as the training advances while reaching $\lambda_i = 0$ for the maximum number of iterations $i_{max} = 80$k in our experiments for 4 epochs on PHOENIX (2012 and 2014) and SIGNUM. Only the experiment in Subsection 6.4 that analyses the effect of the HMM structure does not use the training and validation splitting. Instead it uses all available training data for training the CNN. Therefore we train for 100k iterations here.

$$\lambda_i = \lambda_0 \cdot \left(1 - \frac{i}{i_{max}}\right)^{0.5} \tag{14}$$

CNN inference. Once the CNN training is finished, we consider all three classifiers (the main one and the two auxiliary ones) for estimating the best performing iteration. For the proposed hybrid CNN-HMM approach we add a softmax and use the resulting posteriors in our HMM as observation probabilities.

In the tandem CNN-HMM approach we employ the activations from the last layer before the softmax that yields the highest accuracy on the validation data. With RWTH-PHOENIX-Weather 2012, this is a fully connected layer of the first auxiliary classifier, possibly because the data set does not provide enough data for training an earlier softmax. For RWTH-PHOENIX-Weather 2014 and SIGNUM the pooling layer before the main classifier yields 1024 values. The tandem system requires feature extraction for both training and test sets, since a GMM-HMM system is retrained with them. After a global variance normalisation, we apply PCA to reduce the feature dimension to 200.

Continuous Sign Language Recognition. We base the HMM part of this work on the freely available state-of-the-art open source speech recognition system RASR Rybach et al (2011). Following the hybrid approach we use the posterior probabilities from the CNN, as well as the corresponding class priors. In the following experiments the prior-scaling-factor β is set to 0.3 if not stated otherwise. The LM is estimated as n-gram using the SRILM toolkit by Stolcke (2002). The HMM is employed in bakis structure (Bakis 1976). This is a standard left-to-right structure with forwards, loops and skips across at most one state. Additionally, two subsequent states share the same class probabilities. The transition model $p(s_t|s_{t-1}, w_1^N)$ is pooled across all sign-words. As we actually perform the search in log space we call the transition model TDPs. The TDPs define the transition penalties that account for state changes in the HMM. The garbage class is modelled as an ergodic state with separate transition penalties to add flexibility, such that it can always be inserted between sequences of sign-words. As for RWTH-PHOENIX-Weather 2014 and SIGNUM, we model each sign-word with three hidden states. However, in RWTH-PHOENIX-Weather 2012 we employ a length modelling scheme where sign-words are represented by more or fewer states depending on their average alignment length. For details on the employed length modelling consult Koller et al (2015a). In agreement to most sign language recognition literature, we measure the system performance in WER. WER is based on the Levenshtein alignment between reference and hypothesis sentence and it measures the required numbers of deletion, insertion and

Table 2 Showing the pruning values for each of the data sets. No pruning is necessary with PHOENIX 2012.

Type of pruning	PHOENIX 2012	PHOENIX 2014	SIGNUM
visual threshold	none	2000	2000
visual histogram	none	20000	20000
LM threshold	none	4000	4000
LM histogram	none	10000	10000

substitution operations to transform the recognised hypothesis into the reference sequence.

$$\text{WER} = \frac{\#\text{deletions} + \#\text{insertions} + \#\text{substitutions}}{\#\text{reference observations}} \tag{15}$$

For recognition, we perform a grid search over possible hyper parameters for γ, β and the TDPs. As such, the forward, loop, skip and exit transition penalties are optimised on the dev set (or if not available on the test set) in order to minimise the WER. RASR provides an efficient implementation of the word conditioned tree search which is based on the concepts described in Ney and Ortmanns (2000), which is used for this work. In brief, for each time step the search expands all possible state hypotheses and maintains them in memory. The current score of a hypothesis is composed of the visual score $-log(\frac{p(x_t|\alpha)}{p(\alpha)^\beta})$ and the transition penalty $-log(p(s_t|s_{t-1}))$. Whenever a sign-word ends (which manifests itself in leaving the last state of the HMM), the language model score $-log(p(w)^\gamma)$ and the exit penalty are also added (refer to Section 3, specifically Equation 13 for the exact composition of the search formula). The maximum-approximation (*c.f.* Section 3) allows recombination of state hypotheses that have reached the same state at the same time with the same sign-word history. This significantly limits the combinatorial explosion of the number of search hypotheses. Furthermore, the search space is pruned in order to boost performance and reduce memory consumption. We perform histogram and threshold pruning. The latter acts like a beam search. At each time step, only sign-word hypotheses with scores relatively close to the best hypothesis are allowed. All others are discontinued and therefore removed from memory. This maximum distance from the best hypothesis is represented by the visual threshold pruning value (in *log* domain). After adding the language model score at the word end the LM threshold pruning is applied in the same way. The histogram pruning uses a histogram to limit the amount of hypotheses to the given value. The visual histogram pruning is applied at every state, whereas the LM histogram pruning is only applied after the language model score has been added to each hypothesis at sign-word end states. Table 2 summarises the respective pruning settings for each of the data sets. The exact hyper parameter values for the transition probabilities are given for each experimental description, as they vary from experiment to experiment.

Computational requirements. Using a GeForce GTX 980 GPU with 4GB memory, training on the PHOENIX 2012 data set is done at the speed of $\sim$ 150 frames per second (fps) and inference at a rate of $\sim$ 450 fps. Using the same hardware on PHOENIX 2014 data set yields $\sim$ 35 fps for training and $\sim$ 350 fps for inference. SIGNUM runs at $\sim$ 10 fps during training and $\sim$ 56 fps for inference. HMM recognition is done at $\sim$ 2 fps for PHOENIX 2012 and due to the tighter pruning $\sim$ 25 fps for PHOENIX 2014, while SIGNUM runs at $\sim$ 8. The HMM parameter optimisation took a total of $\sim$ 38 hours for PHOENIX 2012, $\sim$ 130 hours for PHOENIX 2014 and $\sim$ 65 hours for SIGNUM using a single core machine with 2GB RAM.

The training and recognition pipelines have not been optimised for speed. We load individual image files from a file server, which acts as a significant bottleneck. We have experimented with a leveldb database, which is able to double the speed roughly.

6 Experiments

In this section we present experimental evaluation to help estimate key factors influencing the performance of a CNN-HMM hybrid system on the task of sign language recognition. In the next subsection we first analyse the effect of the CNN structure on the final recognition performance. Then, in Subsection 6.2 we evaluate the effect of additional out-of-domain training data. In Subsection 6.3 we compare the hybrid and the tandem approach, before we analyse the effect of the HMM structure in Subsection 6.4. In Subsection 6.6 we provide a general overview comparison against the state-of-the-art, while in Subsection 6.5 we assess model ensembles.

6.1 Effect of CNN Structure

A crucial research question is to estimate the effect of the CNN architecture on a specific task. This subsection aims to provide an answer to this question by applying different CNN structures to the task of sign language recognition, while all remaining hyper parameters are fixed (we adjust the transition probabilities for each experiment). As such, we compare three well-known CNN architectures. All three have, at some point in time, received much attention by the community for outperforming the state-of-the-art largely on different classification tasks. LeNet (full name is LeNet-5), introduced by LeCun et al (1998), was the first successful CNN having 4 non-linear layers. Its application was character recognition of the MNIST digits (LeCun et al 1998) with a size of 32x32 pixel. In this work, we employ a

version which deviates from the original implementation by the number and kind of non-linearities. For simplicity we chose the version distributed jointly with the caffe framework (Jia et al 2014). The other two popular architectures analysed in the scope of this work were winners of the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2012 an 2014. AlexNet by Krizhevsky et al (2012) was the first deeper CNN with 8 layers (5 convolutional and 3 fully connected layers). It won the object classification competition with a top-5 error of 15.4% across the targeted 1000 classes. Two years later Szegedy et al (2015) won the competition with GoogLeNet. A 22-layer deep CNN that achieved a top-5 error of 6.67% on the task. In order to facilitate the comparison of the three mentioned architectures, we have compiled their key characteristics in Table 3. Note that the number of parameters for GoogLeNet includes the parameters used for the two additional auxiliary softmax classifiers. Table 3 shows the input size, the number of non-linear layers and the number of parameters of the whole network (whose last output layers have been adjusted to each of the three data sets analysed in this work). It also shows the number of parameters of the last fully connected classification layer, which often makes up the largest part in the network and varies from task to task. The last layer's size is due to the large amount of sign-labels α (*c.f.* Section 3 for details). α represents the labels belonging to the three hidden states that model each of the sign classes (over 1000) from our vocabulary (for PHOENIX 2014).

Discussion of results: Table 4 summarises the experimental results comparing the different CNN architectures. We see that GoogleNet clearly outperforms the other architectures on both tasks with at least 4% relative improvement in WER. We further see that it is clearly not just the number of parameters that determines the model quality but rather the number of non-linear layers.

6.2 Effect of Finetuning

In this experiment we want to evaluate the effect of using out-of-domain data to train the networks prior to finetuning them on the actual in-domain task using specific but quite limited training data. We therefore make use of the 1.2 million labelled images from the ILSVRC to train the networks. After that we exchange the final fully-connected classification layer and fine-tune the network. In case of the GoogLeNet architecture we exchange the layers of both auxiliary classifiers as well. We perform the experiment with the AlexNet and the GoogLeNet architectures.

Discussion of results: Table 5 and Table 6 report the results for the AlexNet and GoogLeNet architecture, respectively. For both architectures out-of-domain training and subsequent finetuning yields clear gains. With AlexNet we see 30% relative improvement on PHOENIX 2012, 8% on PHOENIX 2014 and 20% on SIGNUM, while with GoogLeNet we see 13% relative improvement on PHOENIX 2012, over 10% on PHOENIX 2014 and again 20% on SIGNUM. We conclude that strongly supervised out-of-domain data has a consistently positive influence on learning hybrid sign language models - at least if the out-of-domain data is as diverse as ImageNet.

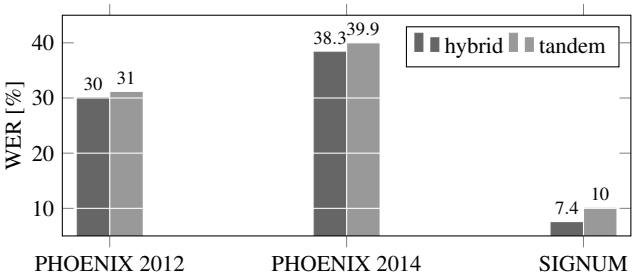

Fig. 9 The hybrid and the tandem approach side-by-side on all three data sets. Results in WER [%]: the lower the better.

6.3 Hybrid Compared to Tandem Modelling

In this subsection we want to explore the question of whether it is better to use the CNN's outputs as features and train a subsequent GMM-HMM system in the so-called tandem approach (Section 3.3) or to directly use the posteriors as observation probabilities as in the presented hybrid approach.

Discussion of results: Figure 9 compares the hybrid CNN-HMM modelling against the tandem modelling. We can see that the hybrid approach slightly outperforms the tandem approach on all three data sets. This is consistent with the literature as found by Golik et al (2013) in speech and handwriting recognition. However, especially in terms of training complexity, the hybrid approach is clearly favourable as the subsequent GMM training is not necessary.

6.4 Effect of Hidden States

Until this point, we have seen experiments estimating the effect of several components on the overall sign recognition pipeline. However, the question remains, how much the HMM impacts the final WER. It is clear that the HMM is the key element to allow the mapping from an input sequence of specific length to an output sequence of different length. But does the hidden state topology influence the final result in a similar way as the CNN structure or the CNN training?

In this subsection we analyse the effect of the HMM structure. More specifically, we want to know if multiple hidden states help the deep CNN to perform better or if

Table 3 Showing number of parameters (weights+biases) in millions of different CNN structures adapted to our tasks: PHOENIX 2012:1443 outputs PHOENIX 2014: 3694 outputs SIGNUM: 1366 outputs.

NN-structure	Input size [px]	#layers	PHOENIX 2012 #params (last fc) [10^6]		PHOENIX 2014 #params (last fc) [10^6]		SIGNUM #params (last fc) [10^6]	
LeNet	227x227	4	73.6	(0.7)	74.7	(1.8)	73.6	(0.6)
AlexNet	227x227	8	62.7	(5.9)	72.0	(15.1)	62.4	(5.5)
GoogLeNet	224x224	22	14.7	(1.4)	21.6	(3.7)	14.5	(1.4)

Table 4 Comparing different CNN structures. Results in WER [%]: the lower the better.

CNN Structure	PHOENIX 2012 Test	PHOENIX 2014 Dev	PHOENIX 2014 Test	SIGNUM Test
LeNet (227x227 input)	47.8	69.5	68.4	17.9
AlexNet	51.5	45.5	44.5	10.6
GoogLeNet	34.1	43.1	42.7	8.9

Table 5 Comparing the effect of pretraining CNN structures on out-of-task data: ILSVRC 2014. The first line represents training from scratch using the AlexNet structure, whereas the second corresponds to fine-tuning weights learnt on Imagenet. Results in WER [%]: the lower the better.

AlexNet	PHOENIX 2012 Test	PHOENIX 2014 Dev	PHOENIX 2014 Test	SIGNUM Test
Randomly initialised	51.5	45.5	44.5	10.6
Fine-tuned	39.2	42.2	41.1	8.7

Table 6 Comparing the effect of pretraining CNN structures on out-of-task data: ILSVRC 2014. The first line represents training from scratch using the GoogLeNet structure, whereas the second corresponds to finetuning weights learnt on Imagenet. Results in WER [%]: the lower the better.

GoogLeNet	PHOENIX 2012 Test	PHOENIX 2014 Dev	PHOENIX 2014 Test	SIGNUM Test
Randomly initialised	34.1	43.1	42.7	8.9
Fine-tuned	30.0	38.3	38.8	7.4

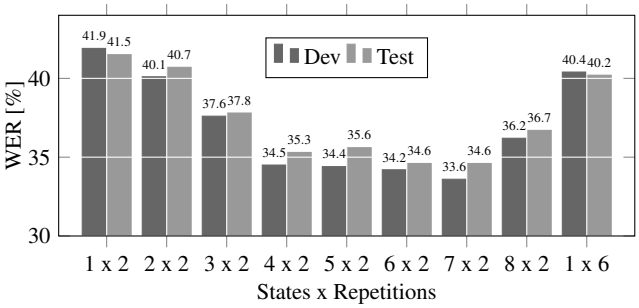

Fig. 10 Showing the best achieved WERs in [%] (the lower the better) on PHOENIX 2014 for different numbers of states and repetitions.

they are a relic of the GMM-HMM architecture that strong CNNs make redundant. Therefore, we perform experiments on the PHOENIX 2014 data set altering the HMM topology w.r.t. the number of hidden states. The baseline system corresponds to a HMM architecture that models each sign-word with 3 hidden states which are each repeated twice (sharing the same probabilities). Thus, this topology has 6 states, but only 3 probability distributions need to be estimated by the CNN. This standard bakis topology ensures that we can compensate for variation in signing speed by skipping states. By definition, we can skip at most one state. The repetitions therefore ensure that all emission probabilities have to be visited at least a single time. In order to allow for valid conclusions, we need to make sure that all systems have the chance to find a good alignment w.r.t. their HMM architecture. In opposition to all other experiments presented in the scope of this work, we therefore perform multiple iterations of re-alignment, where we re-estimate the viterbi path. We start from a flat segmentation, where the available frames are equally distributed across all states of a sentence. The re-alignment then iteritively updates the frame labelling and therefore affects the subsequent CNN training. Thus, after each re-alignment we perform a fine-tuning of the previous iteration's model for 100k iterations (~ 4 epochs). Each iteration takes about 6 hours for CNN training and 20 minutes for viterbi alignment. We perform 10 re-alignment iterations for all different HMM topologies and report the best result among all iterations.

Discussion of results: Figure 10 shows the results in terms of WER on the PHOENIX 2014 dev and test partition. We first vary the amount of states per sign-word from 1 to 8, maintaining the 2 state repetitions. In this setting, the baseline of 3 states and 2 repetitions clearly outperforms topologies with less states. However, we see the best performance further increasing the numbers of states to 7. We note a WER difference between the weakest (1×2 states) and the strongest topology (7×2 states) of 8.5% absolute and over 20% relative. The 7 state architecture achieves 33.4% WER on the dev set and 34.4 on the test set. One could argue that it is the implied HMM length and not the division into hidden states that produces the improvements with longer HMMs. Therefore, we further perform an experiment with 1 state and 6 repetitions, which has the same length behaviour as the baseline. However, this model performs much worse than the baseline. As such, we can conclude that the HMM

Table 7 Showing how the HMM structure in terms of HMM states and repetitions affects the total number of HMM states and the neural network parameters (weights+biases) in millions.

HMM Structure States x Repetitions	PHOENIX 2014 Total States	Parameters [10^6]
1 x 2	1232	14.1
2 x 2	2463	17.9
3 x 2	3694	21.7
4 x 2	4925	25.4
5 x 2	6156	29.2
6 x 2	7387	33.0
7 x 2	8618	36.8
8 x 2	9849	40.6

architecture has a strong influence on the recognition performance. Nevertheless, in Table 7 we can see how the number of HMM states affects the overall model size. This significantly impacts runtime.

6.5 Effortless Ensemble of Models

Finally, we want to show that a log-linear combination of multiple CNN models can further improve performance. We therefore define the probability by the visual model to be the combined product of each single model i scaled by a factor δ_i as in

$$p(x_1^T|w_1^N) = \max_{s_1^T}\{\prod_{t=1}^{T}\prod_i [\frac{p_i(\alpha|x_t)}{p_i(\alpha)^\beta}]^{\delta_i} \cdot p\left(s_t|s_{t-1}, w_1^N\right)\} \tag{16}$$

In the scope of this work we combine $I = 2$ models. The fact that model ensembles increase performance is well known. However, typically the building of models that are sufficiently complementary to yield any improvements constitutes a large computational overhead. In this Section, we show that the process of re-aligning the models already adds sufficient discriminative information. Even models from successive re-alignment iterations yield strong gains when deployed as ensemble. This is remarkable as it means that with the proposed algorithm we get such models free of additional effort.

We choose two successive iterations of the best HMM architecture using 7 states and 2 repetitions, namely the 10^{th} iteration yielding 33.6 / 34.6 and the 9^{th} iteration yielding 33.8 / 34.6 on the development set and on the test set respectively. The log-linear combination with $\delta_1 = 0.87$ and $\delta_2 = 0.13$ yields a WER of 31.6% and 32.5% for dev and test respectively on PHOENIX 2014. This corresponds to a relative gain of around 6% compared to the single models.

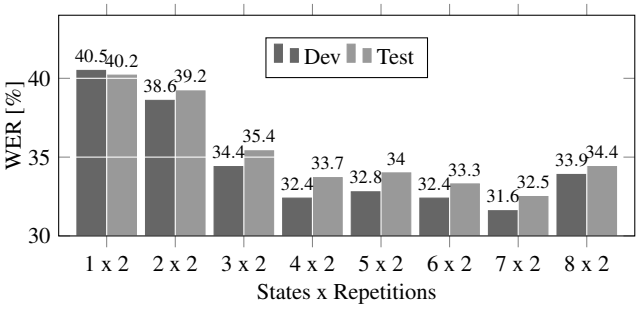

Fig. 11 Showing the best achieved WERs in [%] (the lower the better) on PHOENIX 2014 for log-linear model combination of the two best alignment iterations (while keeping the HMM architecture fixed).

6.6 General Comparison to State-of-the-art

Table 8 shows a detailed comparison to the state-of-the art on the three employed benchmark corpora. Besides performance measures, it reports the method of choice by the respective publications. Note that the proposed hybrid approach currently exploits only a single cropped hand of the signer and yet achieves state-of-the-art performance. Sign language is highly multimodal and makes heavy use of manual components (hand shape, orientation, place of articulation, movement) and also non-manual components (facial expression, eyebrow height, mouth, head orientation, upper body orientation). Most of the competing approaches use these additional modalities in recognition, which is why we expect additional gain when including them in the proposed approach. The previously best hand only result mentioned in Koller et al (2016a) also relied on CNN models, but did not employ the hybrid approach end-to-end in recognition, loosing some performance due to this. It set the benchmark on PHOENIX 2014 Multisigner to 51.6% WER. However, our proposed CNN-HMM achieves a strong result of 33.6% and 34.6% on dev and test respectively with a single model and 31.6%/32.5% with model combination. This corresponds to about 20% absolute WER or over 38% relative improvement. On the single signer corpus RWTH-PHOENIX-Weather 2012 the proposed approach improved the best baseline from 35.5% to 30.0%, still being a relative improvement of over 15%. On SIGNUM we improve the best known word error rates from from 12.0% to 7.4%. As can be seen in Table 8, our hand-only hybrid CNN-HMM even outperforms multimodal approaches.

Nevertheless, the need to include more modalities than just the right hand is revealed by looking at the recognition errors. Qualitative examination of the top confusions on PHOENIX 2014 made by the hybrid approach highlight confused pairs such as “SNOW” with “RAIN” or “SHOWER” with “RAIN”. However, these signs share the same hand configurations, whereas only the mouth shape

Table 8 Comparison with state-of-the-art. Results in WER [%]: the lower the better. Best results of the proposed approach are single models. Model combination further improves the error on PHOENIX 2014 down to 34.4%.

Method		PHOENIX 2012 Test	PHOENIX 2014 Dev	PHOENIX 2014 Test	SIGNUM Test
von Agris et al (2008a)	GMM-HMM	–	–	–	12.7
Gweth et al (2012)	GMM-HMM (MLP feat.)	–	–	–	11.9
Forster et al (2013b)	GMM-HMM	41.9	–	–	10.7
Forster et al (2013a)	GMM-HMM	38.6	–	–	10.7
Koller et al (2015a)	GMM-HMM	34.3	57.3	55.6	10.0
Koller et al (2015a)	GMM-HMM (CMLLR)	–	55.0	53.0	–
Koller et al (2016a)	GMM-HMM (CNN feat.)	31.2	47.1	45.1	7.6
Koller et al (2016b)	tandem CNN-HMM	31.0	39.9	38.8	10.0
Camgoz et al (2017)	CNN-LSTM with CTC	–	40.8	40.7	–
Cui et al (2017)	CNN-LSTM with CTC	–	39.4	38.7	–
Proposed approach:	hybrid CNN-HMM	**30.0**	**31.6**	**32.5**	**7.4**

changes. Given the classification relies purely on the right hand, it is understandable that it cannot distinguish between these signs. The top 30 confusions all relate to this type of error.

7 Conclusion and Future Work

In this work, we introduced an end-to-end embedding of a CNN into a HMM, while interpreting the outputs of the CNN in a truly Bayesian framework and training the system as a hybrid CNN-HMM in an end-to-end fashion Most state-of-the-art approaches in gesture and sign language modelling use a sliding window approach or simply evaluate the output in terms of overlap with the ground truth. While this is sufficient for data sets that provide such training and evaluation characteristics, it is unsuitable for real world use. For the field to move forward more realistic scenarios, such as those imposed by challenging real-life sign language corpora, are required.

In this manuscript, we presented a hybrid CNN-HMM framework that combines the strong discriminative abilities of CNNs with the sequence modelling capabilities of HMMs, while abiding to Bayesian principles. This work represents the extended version of our previous work (Koller et al 2016b), where we were the first to present such an embedding in the context of sign language and gesture recognition. With the hybrid method we were able to achieve a large relative improvement of over 15% compared to the previous state-of-the-art on three challenging benchmark continuous sign language recognition data sets. On the two single signer data sets RWTH-PHOENIX-Weather 2012 and SIGNUM we improve the best known word error rates from 35.5% to 30.0% and from 12.0% to 7.4% respectively, while only employing basic hand-patches as input. On the difficult 9 signer $>$ 1000 vocab RWTH-PHOENIX-Weather 2014 Multisigner, we lower the error rates from 51.6% / 50.2% to 31.6% / 32.5% on dev / test.

In the scope of this extended manuscript, we significantly added to the theoretical explanation of the hybrid approach, with the aim of making its idea more accessible to newcomers to the field and presented much more extensive experiments: We analysed the effect of both CNN- and HMM-structure on the hybrid approach. We investigated the effect of using out-of-domain data to train the network prior to finetuning using in-domain data. Finally, we showed that the use of ensembles of hybrid CNN-HMMs is able to further boost performance.

In terms of future work, we would like to extend our approach to cover all relevant modalities. Moreover, techniques to overcome the necessary initial alignment, such as end-to-end training will also be investigated.

References

Assael YM, Shillingford B, Whiteson S, de Freitas N (2016) LipNet: End-to-End Sentence-level Lipreading. arXiv:161101599 [cs] 1611.01599

Bahl LR, Jelinek F, Mercer RL (1983) A Maximum Likelihood Approach to Continuous Speech Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-5(2):179–190, DOI 10.1109/TPAMI.1983.4767370

Bakis R (1976) Continuous speech recognition via centisecond acoustic states. The Journal of the Acoustical Society of America 59(S1):S97–S97, DOI 10.1121/1.2003011

Bengio Y, Frasconi P (1996) Input-output HMMs for sequence processing. IEEE Transactions on Neural Networks 7(5):1231–1249

Bluche T, Ney H, Louradour J, Kermorvant C (2015) Framewise and CTC training of Neural Networks for handwriting recognition. In: 2015 13th International Conference

on Document Analysis and Recognition (ICDAR), pp 81–85, DOI 10.1109/ICDAR.2015.7333730

Bourlard H, Wellekens CJ (1989) Links Between Markov Models and Multilayer Perceptrons. In: Advances in Neural Information Processing Systems, pp 502–507

Bourlard HA, Morgan N (1993) Connectionist Speech Recognition: A Hybrid Approach. Kluwer Academic Publishers, Norwell, MA

Camgoz NC, Hadfield S, Koller O, Bowden R (2017) SubUNets: End-to-end Hand Shape and Continuous Sign Language Recognition. In: IEEE International Conference on Computer Vision, pp 22–27

Cui R, Liu H, Zhang C (2017) Recurrent Convolutional Neural Networks for Continuous Sign Language Recognition by Staged Optimization. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA

Dreuw P, Deselaers T, Rybach D, Keysers D, Ney H (2006) Tracking Using Dynamic Programming for Appearance-Based Sign Language Recognition. In: IEEE International Conference Automatic Face and Gesture Recognition, IEEE, Southampton, UK, pp 293–298

Escalera S, Baró X, Gonzalez J, Bautista MA, Madadi M, Reyes M, Ponce-López V, Escalante HJ, Shotton J, Guyon I (2014) Chalearn looking at people challenge 2014: Dataset and results. In: Computer Vision-ECCV 2014 Workshops, Springer, pp 459–473

Forster J, Schmidt C, Hoyoux T, Koller O, Zelle U, Piater J, Ney H (2012) RWTH-PHOENIX-Weather: A Large Vocabulary Sign Language Recognition and Translation Corpus. In: International Conference on Language Resources and Evaluation, Istanbul, Turkey, pp 3785–3789

Forster J, Koller O, Oberdörfer C, Gweth Y, Ney H (2013a) Improving Continuous Sign Language Recognition: Speech Recognition Techniques and System Design. In: Workshop on Speech and Language Processing for Assistive Technologies, Grenoble, France, pp 41–46, satellite Workshop of INTERSPEECH 2013

Forster J, Oberdörfer C, Koller O, Ney H (2013b) Modality Combination Techniques for Continuous Sign Language Recognition. In: Iberian Conference on Pattern Recognition and Image Analysis, Springer, Madeira, Portugal, Lecture Notes in Computer Science 7887, pp 89–99

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

Golik P, Doetsch P, Ney H (2013) Cross-entropy vs. squared error training: A theoretical and experimental comparison. In: INTERSPEECH, pp 1756–1760

Granger N, el Yacoubi MA (2017) Comparing Hybrid NN-HMM and RNN for Temporal Modeling in Gesture Recognition. In: Neural Information Processing, Springer, Cham, Lecture Notes in Computer Science, pp 147–156, DOI 10.1007/978-3-319-70096-0_16

Graves A, Schmidhuber J (2005) Framewise phoneme classification with bidirectional LSTM and other neural network architectures. Neural Networks 18(5):602–610

Gweth Y, Plahl C, Ney H (2012) Enhanced Continuous Sign Language Recognition using PCA and Neural Network Features. In: CVPR 2012 Workshop on Gesture Recognition, Providence, Rhode Island, USA, pp 55–60

Hermansky H, Ellis DW, Sharma S (2000) Tandem connectionist feature extraction for conventional HMM systems. In: Acoustics, Speech, and Signal Processing, 2000. ICASSP'00. Proceedings. 2000 IEEE International Conference On, IEEE, vol 3, pp 1635–1638

Huang J, Zhou W, Li H, Li W (2015) Sign Language Recognition using 3D convolutional neural networks. In: 2015 IEEE International Conference on Multimedia and Expo (ICME), pp 1–6, DOI 10.1109/ICME.2015.7177428

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

Koller O, Ney H, 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, Shanghai, PRC, pp 1–6

Koller O, Ney H, Bowden R (2014) Read My Lips: Continuous Signer Independent Weakly Supervised Viseme Recognition. In: Proceedings of the 13th European Conference on Computer Vision, Zurich, Switzerland, pp 281–296

Koller O, Forster J, Ney H (2015a) Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers. Computer Vision and Image Understanding 141:108–125, DOI 10.1016/j.cviu.2015.09.013

Koller O, Ney H, Bowden R (2015b) Deep Learning of Mouth Shapes for Sign Language. In: Third Workshop on Assistive Computer Vision and Robotics, ICCV, Santiago, Chile

Koller O, Ney H, Bowden R (2016a) 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, pp 3793–3802

Koller O, Zargaran S, Ney H, Bowden R (2016b) Deep Sign: Hybrid CNN-HMM for Continuous Sign Language Recognition. In: British Machine Vision Conference, York, UK

Koller O, Zargaran S, Ney H (2017) Re-Sign: Re-Aligned End-to-End Sequence Modelling with Deep Recurrent

CNN-HMMs. In: IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA

Krishnan R, Sarkar S (2015) Conditional distance based matching for one-shot gesture recognition. Pattern Recognition 48(4):1298–1310

Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet Classification with Deep Convolutional Neural Networks. In: Advances in Neural Information Processing Systems, pp 1106–1114

Le HS, Pham NQ, Nguyen DD (2015) Neural Networks with Hidden Markov Models in Skeleton-Based Gesture Recognition. In: Nguyen VH, Le AC, Huynh VN (eds) Knowledge and Systems Engineering, no. 326 in Advances in Intelligent Systems and Computing, Springer International Publishing, pp 299–311, DOI 10.1007/978-3-319-11680-8_24

LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proceedings of the IEEE 86(11):2278–2324

Marcel S, Bernier O, Viallet JE, Collobert D (2000) Hand gesture recognition using input-output hidden markov models. In: Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference On, IEEE, pp 456–461

Molchanov P, Gupta S, Kim K, Kautz J (2015) Hand Gesture Recognition with 3D Convolutional Neural Networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp 1–7

Neverova N, Wolf C, Taylor GW, Nebout F (2014) Multi-scale Deep Learning for Gesture Detection and Localization. In: Agapito L, Bronstein MM, Rother C (eds) Computer Vision - ECCV 2014 Workshops, Springer International Publishing, Lecture Notes in Computer Science, pp 474–490, DOI 10.1007/978-3-319-16178-5_33

Ney H, Ortmanns S (2000) Progress in dynamic programming search for LVCSR. Proceedings of the IEEE 88(8):1224–1240

Ong EJ, Koller O, Pugeault N, Bowden R (2014) Sign Spotting using Hierarchical Sequential Patterns with Temporal Intervals. In: IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, pp 1931–1938

Pigou L, van den Oord A, Dieleman S, Van Herreweghe M, Dambre J (2018) Beyond Temporal Pooling: Recurrence and Temporal Convolutions for Gesture Recognition in Video. International Journal of Computer Vision 126(2-4):430–439, 1506.01911

Rao Y, Lu J, Zhou J (2017) Attention-Aware Deep Reinforcement Learning for Video Face Recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 3931–3940

Richard A, Kuehne H, Gall J (2017) Weakly Supervised Action Learning with RNN based Fine-to-coarse Modeling. In: IEEE Conference on Computer Vision and Pattern Recognition, Hawaii, USA, 1703.08132

Richard MD, Lippmann RP (1991) Neural Network Classifiers Estimate Bayesian a posteriori Probabilities. Neural Computation 3(4):461–483

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

Schlüter R, Nussbaum-Thom M, Ney H (2012) Does the Cost Function Matter in Bayes Decision Rule? IEEE Transactions on Pattern Analysis and Machine Intelligence 34(2):292–301, DOI 10.1109/TPAMI.2011.163

Schober M, Adam A, Yair O, Mazor S, Nowozin S (2017) Dynamic Time-Of-Flight. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)

Stolcke A (2002) SRILM- an Extensible Language Modeling Toolkit. In: Proc. International Conference on Spoken Language Processing (ICSLP), Denver, Colorado

Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going Deeper With Convolutions. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, Ma, USA, pp 1–9

von Agris U, Knorr M, Kraiss KF (2008a) The significance of facial features for automatic sign language recognition. In: Automatic Face & Gesture Recognition, 2008. FG'08. 8th IEEE International Conference On, IEEE, pp 1–6

von Agris U, Zieren J, Canzler U, Bauer B, Kraiss KF (2008b) Recent developments in visual sign language recognition. Universal Access in the Information Society 6(4):323–362

Wu D, Shao L (2014) Deep dynamic neural networks for gesture segmentation and recognition. In: Computer Vision-ECCV 2014 Workshops, Springer, pp 552–571

Wu D, Pigou L, Kindermans PJ, LE N, Shao L, Dambre J, Odobez JM (2016) Deep Dynamic Neural Networks for Multimodal Gesture Segmentation and Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence PP(99):15