Re-Sign: Re-Aligned End-to-End Sequence Modelling with Deep Recurrent CNN-HMMs

Oscar Koller, Sepehr Zargaran and Hermann Ney
Human Language Technology & Pattern Recognition Group
RWTH Aachen University, Germany
{koller,ney}@cs.rwth-aachen.de

Abstract

This work presents an iterative re-alignment approach applicable to visual sequence labelling tasks such as gesture recognition, activity recognition and continuous sign language recognition. Previous methods dealing with video data usually rely on given frame labels to train their classifiers. Looking at recent data sets, these labels often tend to be noisy which is commonly overseen. We propose an algorithm that treats the provided training labels as weak labels and refines the label-to-image alignment on-the-fly in a weakly supervised fashion. Given a series of frames and sequence-level labels, a deep recurrent CNN-BLSTM network is trained end-to-end. Embedded into an HMM, the resulting deep model corrects the frame labels and continuously improves its performance in several re-alignments. We evaluate on two challenging publicly available sign recognition benchmark data sets featuring over 1000 classes. We outperform the state-of-the-art by up to 10% absolute and 30% relative.

1. Introduction

Sequence data is difficult to annotate, when trying to attribute a label to each frame. Large amounts of continuous data usually contain labelling ambiguities and are not free of errors. The community needs to find ways how to deal with that. Sequence data annotations should be expected to vary in quality and weakly supervised approaches could cope with that. Evaluation metrics could judge on the sequence level rather than at exact frame matches, favouring those approaches that generalise over annotation imperfections. In this work we are proposing an iterative re-alignment algorithm to overcome such challenges. The presented approach has a direct impact on training classifiers for challenging sequence tasks like gesture or sign language recognition and relevant neighbouring fields.

Even though Long Short Term Memory (LSTM) models achieve outstanding results in speech recognition, hand writing recognition, machine translation, image captioning and image translation, until now they have not been successfully trained on real-life continuous gesture and sign language recognition tasks distinguishing a large number of classes. Our experimental evidence suggests that label to video re-alignments are needed to permit such successful training. Natural continuous sign language, as opposed to its artificial counterparts, poses a truly challenging large-scale classification task with inherent segmentation. The stream of continuous sign gestures constitutes overlapping context- and user-dependent interactions that make use of multi-modal channels which can often be observed in non-synchronous realisations. In this paper, we introduce new guidelines for outperforming the current state-of-the-art in the field of human gesture and sign language recognition. We propose a multilayer bi-directional LSTM that is trained end-to-end with a deep Convolutional Neural Network (CNN). The joint model is embedded into a Hidden-Markov-Model (HMM) for iterative refinement and final recognition outperforming the state of the art on two publicly available benchmark data sets by a large margin. As such we make several contributions addressing short comings in the current state-of-the-art:

1. We empirically validate the importance of re-alignments for continuous gesture and sign language recognition tasks and propose an iterative re-alignment algorithm based on a hybrid CNN-BLSTM embedded into a HMM.
2. To promote reproducibility on the selected corpora we will make the best alignments publicly available [1].
3. To the best of our knowledge, we are the first to successfully train end-to-end CNN-BLSTMs for continuous gesture and sign language tasks distinguishing over 1000 classes.

[1] http://www-i6.informatik.rwth-aachen.de/~koller/RWTH-PHOENIX/

4. We find that whole frame images outperform tracked hands.

This paper is organised as follows: after discussing the state-of-the-art and its short-comings in Section 2, we present the approach in Section 3. In Section 4, we show all empirical experiments to back our proposition. Finally, we end with conclusions and future work in Section 5

2. Related Work

This work introduces new guidelines for outperforming the current state-of-the-art in the field of human gesture and sign language recognition. Namely, our proposition relies on an iterative re-alignment algorithm. Iteratively refining the provided training labels allows to take full advantage of deep recurrent CNN-BLSTM models which until now have not been successfully applied to comparable tasks that features real-life, heavily co-articulated gesture and sign language data with a large number of classes.

Re-aligning the labels using the deprecated GMM-HMM approach has long time been a common procedure in speech recognition. In the recent speech literature some efforts can be observed performing re-alignments with GMM-free systems [33] purely based on deep neural networks, which is related to our presented approach. However, in gesture recognition and neighbouring fields not much work exists that exploits re-alignments. Most approaches simply rely on the provided frame labels or divide the input sequence length by the number of modelled states or classes performing a non-optimal flat segmentation, as in [41].

LSTMs [19] have been discovered nearly two decades ago. Since then, they have had large success in many human language related technologies *e.g.* as bidirectional LSTM based acoustic models [32, 44, 15] or language models [35] in speech recognition, in neural machine translation [36, 7] or handwriting recognition [16].

In related computer vision tasks, such as action or activity recognition, LSTMs seem to yield much less gain or are even outperformed by pooled multi-stream feed forward architectures [28]. We argue though, seconding Pigou *et al.* [29], that current general video classification data sets constitute challenges where the detection of specific objects in the scene is often sufficient for successful classification. However, when it comes to gesture and sign language recognition temporal sequence information, *e.g.* motion, is often critical. Looking at the state of the art in these fields, we note that in the last three years (particularly after 2015) several works successfully exploited different variants of LSTMs. But all prior work has some short comings preventing it from exploiting the full benefits of the architecture and transferring it to more challenging problems: Most works do not train the LSTMs jointly with CNNss in an end-to-end fashion [2, 26, 42, 43], their architecture is not deep [38] or they do not employ bidirectional LSTMs [10, 2, 42, 43, 28]. All previous work has been evaluated on a low number of classes [29, 38, 26, 12], sometimes with low input dimensionality not requiring image processing (the data sets provide tracked skeletons) [26, 12]. No work exists that models truly continuous data with overlapping classes as it is the case in natural gesturing and sign language recognition. To the best of our knowledge, in the fields of activity, action, gesture and sign language recognition, we are the first to successfully report the end-to-end training of CNN-LSTM networks for challenging continuous recognition tasks distinguishing over 1000 classes. Recently however, we learnt about works in lip-reading that employed LSTMs [8] and Gated Recurrent Units (GRUs) [1] to successfully distinguish a large number of classes as well. The latter paper employs Connectionist Temporal Classification (CTC) [16], which is related to the presented approach in this work, but differs in several points. CTC can be regarded as a special case of the hybrid full-sum HMM alignment, whereas we propose a viterbi best path alignment. Moreover, CTC has a specific HMM topology (1 state with no repetition followed by a tied blank state) and we follow the standard automatic speech recognition (ASR) bakis topology with 3 states and 2 repetitions. Furthermore, in CTC training the actual re-alignment is commonly applied for every mini-batch, whereas we realign every 4 epochs. A big advantage of our approach is that it doesn't require the LSTMs to cover a full input sequence, instead we can define the sequence and perform 'chopping' of the input. This allows to use much more complex visual models that have larger memory footprints. Refer to [5] for details on the comparison of the two approaches. In terms of sign language recognition, our work is related to [23] but differs in our proposed iterative label re-alignment strategy and the recurrent CNN-BLSTM models.

3. Iterative Re-Alignment of Labels and Video

Following a recent line of works [25, 41, 23] that make use of hybrid neural network and HMM modelling [6, 3] for gesture and sign language recognition, we also opt for a hybrid architecture. However, unlike the previous mentioned publications, we embed a CNN-BLSTM in a HMM and propose an iterative re-alignment algorithm in the following subsection. An overview of the presented algorithm can be seen in Figure 1.

3.1. Recognition Basics

The target in all sequence learning tasks is to predict a sequence of output symbols w_1^N, given an input sequence of images $x_1^T = x_1, \ldots, x_T$. To train the sequence classifier in a supervised setting either a direct frame-labelling is available or the target label sequence w is given and a monotonous occurrence of the corresponding events in the

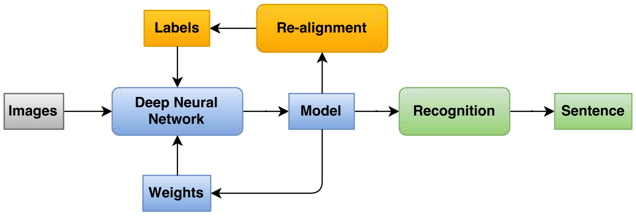

Figure 1. Overview of iterative re-alignment algorithm used to refine the training labels.

video material can be assumed. Our followed hybrid modelling approach makes use of the statistical paradigm with Bayes' decision rule, which has been successfully applied to speech recognition, hand writing recognition and statistical machine translation for decades now. The target objective is to maximise the true class posterior probability distribution $Pr(w_1^N|x_1^T)$ over the whole utterance. Decision theory allows to split up the class posterior probability in the class prior $Pr(w_1^N)$ and the class-conditional probability $Pr(x_1^T|w_1^N)$, which we can then model by different information sources. $p(w_1^N)$ will be modelled by a n-gram language model, whereas $p(x_1^T|w_1^N)$ will be modelled by a CNN-LSTM:

$$[w_1^N]_{\text{opt}} = \operatorname*{argmax}_{w_1^N} \left\{ p(w_1^N) \cdot p(x_1^T|w_1^N) \right\} \tag{1}$$

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

$$p(x_1^T|w_1^N) = \sum_{s_1^T} p(x_1^T, s_1^T|w_1^N) \tag{2}$$

$$= \sum_{s_1^T} \prod_{t=1}^{T} p(x_t, s_t|x_1^{t-1}, s_1^{t-1}, w_1^N) \tag{3}$$

$$= \sum_{s_1^T} \prod_{t=1}^{T} p(x_t|x_1^{t-1}, s_1^t, w_1^N) \cdot p(s_t|x_1^{t-1}, s_1^{t-1}, w_1^N) \tag{4}$$

$$= \sum_{s_1^T} \prod_{t=1}^{T} p(x_t|, s_t, w_1^N) \cdot p(s_t|s_{t-1}, w_1^N) \tag{5}$$

where the sum in Equation 2 expresses all viable paths that lead to the same output sequence w_1^N. Equation 3 and 4 constitute reformulations with help of the chain rule. Assuming s to be non-observable and a first order Markov process leads to Equation 5. After applying the viterbi approximation, which considers just the most likely path and plugging everything into Equation 1, we get:

$$\operatorname*{argmax}_{w_1^N} \left\{ p(w) \max_{s_1^T} \prod_{t=1}^{T} p(x_t|, s_t, w) p(s_t|s_{t-1}, w) \right\} \tag{6}$$

To be able to use a strong vision model we replace the emission probability $p(x_t|s_t, w_1^N)$ of the HMM by an embedded discriminative CNN-LSTM. Its outputs constitute posterior probabilities. Therefore, to keep the approach fully Bayesian, a conversion of the posteriors to class-conditional likelihoods following Bayes' rule is needed:

$$p(x_t|s_t, w_1^N) = p(x_t) \cdot \frac{p(s_t, w_1^N|x_t)}{p(s_t, w_1^N)} \tag{7}$$

where $p(s_t, w_1^N)$ can be approximated by the state label counts in our frame-state-alignment used to train the CNN-LSTM. In the implementation, we add several hyperparameters allowing to control the impact of the language model (γ) and the state prior (α). Neglecting the constant frame prior $p(x_t)$, we finally optimise the following equation to find the best output sequence:

$$\operatorname*{argmax}_{w} \left\{ p(w)^{\gamma} \max_{s_1^T} \prod_{t} \frac{p(s_t, w|x_t)}{p(s_t, w)^{\alpha}} p(s_t|s_{t-1}, w) \right\} \tag{8}$$

$p(s_t|s_{t-1})$ constitutes the state transition model, which we model pooled across all output classes. Only the ergodic garbage class has separate transition probabilities, such that it can always account for frames in between the output symbols. The employed HMM is in bakis structure, which is a left-to-right structure with forward, loop and skip transitions across at most one state. Furthermore, we implement each gesture class with six states, where two subsequent states share the same class probabilities. Additional to forward, loop and skip we model an exit penalty, which is added whenever a full symbol (gesture class) is emitted. These penalties, jointly with above mentioned γ and α represent the hyper-parameters in this approach, which are optimised on an independent development set using a grid search. All HMM experiments are conducted through RASR [31], a freely available and open-sourced speech recognition framework. We employ histogram and threshold pruning of the search space for better performance and memory consumption. All experiments are evaluated on word error rate (WER) which measures the number of necessary insertion, substitution and deletions to turn the recognised sentence to the reference sentence.

$$\text{WER} = \frac{\#\text{deletions} + \#\text{insertions} + \#\text{substitutions}}{\#\text{reference observations}} \tag{9}$$

3.2. Iterative EM Re-Alignment

CNNs have shown an incredible improvement in gesture and sign language processing [23]. But in these tasks motion seems to play a very important role and only relying on the generative HMM state sequence to capture temporal change may not be enough. Recurrent networks such as LSTMs have access to the whole or at least a sub-sequence and therefore may remedy that shortcoming. However, our experiments shifting feed forward to recurrent networks quickly revealed that it is not easy to benefit from the added modelling complexity. The fixed frame-state alignment being good for feed forward CNNs has proven non-optimal to train the LSTMs.

In this work, we propose an iterative re-alignment algorithm that helps to overcome these problems. The basic idea relies on Expectation Maximisation (EM) [9]. We initialise the algorithm with a provided frame labelling or a frame-state-alignment generated by standard CNN training. We then iteratively perform first a maximisation step, which corresponds to fitting our CNN-LSTM model to the data and then an expectation step, in which the previously trained model is embedded in a hybrid HMM recognition as described in the previous section. However, not a full recognition is performed, but rather a forced alignment: We force the word sequence w_1^N to match the given transcription in our training data and search for the most likely state sequence s_1^T. As depicted in Figure 1, after each successful re-alignment the following iteration of CNN-LSTM training benefits from the new frame-state labels and it also uses the previous iteration's model weight for initialisation. After each iteration we perform a recognition of the development data. Here, we optimise the mentioned hyper-parameters to obtain the best results. The same hyper-parameters are then used to re-align the next iteration.

3.3. Recurrent CNN-LSTM

In this work, we deal with the recognition of challenging real-life gesture and sign language video data. We therefore aim at joining a powerful and deep CNN with several bi-directional LSTM layers [17, 27]. In order to train the full network end-to-end, the CNN architecture of choice should have a low memory footprint, while still being very deep. After comparing different CNN architectures [34, 24, 37], we opted for the 22 layer deep GoogleNet [37] architecture, which we initially pre-train on the 1.4M images from the ILSVRC-2012 [30]. The main building blocks of this architecture are Inception modules which are fusion of multiple convolutional layers with different receptive fields applied to the output of a 1x1 convolution layer which serves as a dimensionality reduction tool. Finally, in addition to the last classifier, GoogLeNet also makes use of two auxiliary classifiers at in lower layers which are added to the final loss with a weight of 0.3. The pre-trained standalone CNN achieves a top-1 accuracy of 68.7% and a top-5 accuracy of 88.9% in the ILSVRC. The network uses ReLUs as non-linearity in its convolutional layers and 70% dropout ratio is set to prevent over-fitting.

LSTMs are RNN variants that were invented to overcome the vanishing gradient problem [4] and as such can learn long time dependencies much better than vanilla RNNs. As the gradients are fully differentiable, we can train the recurrent network with Back Propagation Through Time (BPTT) [40]. We use stochastic gradient descent with an initial learning rate $\lambda_0 = 0.001$ for CNN-LSTM architectures and $\lambda_0 = 0.01$ 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 $i_{max} = 100$k (being roughly 4 epochs) in our experiments.

$$\lambda_i = \lambda_0 \cdot \left(1 - \frac{i}{i_{max}}\right)^{0.5} \tag{10}$$

Our CNN-LSTM implementation is based on [20]. The employed bi-directional CNN-LSTM-HMM architecture is depicted in Figure 2. All images are directly feed as inputs to a deep CNN architecture. All hand patches in our experiments are tracked in a similar fashion as the dynamic programming based approach of [11]. Furthermore all hand related experiments use the right hand which is the signers dominant hand. Both hand and full frame image inputs to the CNN are of the size 256x256 pixels. Each input is normalised by subtraction of the pixel-wise mean of all images in the training set. The resulting image is then cropped at a random position to a new size of 224x224 pixels.

4. Experiments

We conduct our experiments on the RWTH-PHOENIX-Weather 2014 [14] data set with more than a million frames and a vocabulary size of 1,081 unique words. This data set is recorded from a public television broadcast and contains sentences performed by 9 different signers. In this data set, in addition to the training set, two independent evaluation sets are provided each amounting to almost 10% of the training set in size. It is important to note that the these sets are not signer-independent, meaning all signers appear in all 3 sets. A 4-gram language model with a perplexity of 46.9 is trained on and used for experiments on this data set. Additionally, we created a signer independent subset of PHOENIX 2014 and a signer-independent 4-gram language model with a perplexity of 60.4 measured on the dev set. Due to a sensible amount of data, we chose to leave out signer 5 for signer independent experiments.

4.1. Uni-modal vs. Multi-modal

Given the success of previous work [22, 23] with manual (hand) features, we start our experiments using right

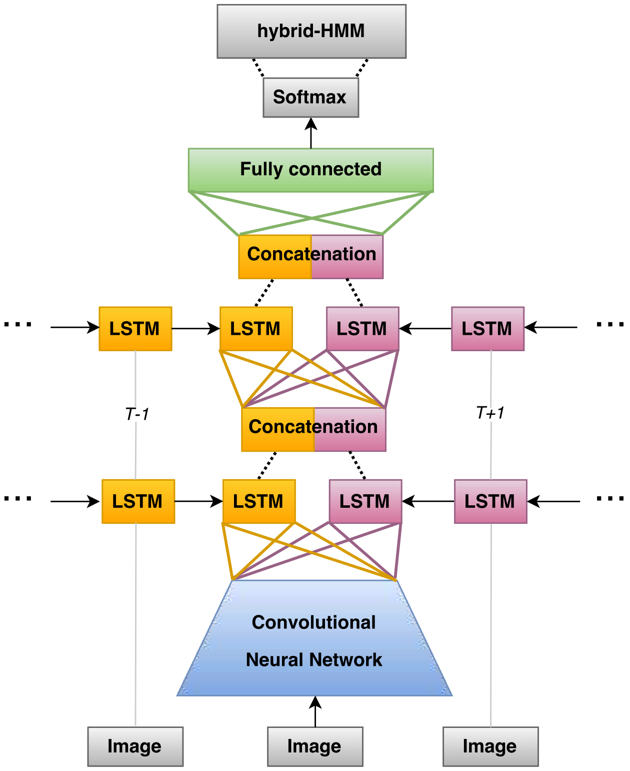

Figure 2. End-to-end CNN-LSTM architectures with two BLSTM layers.

	Train	Dev	Test
Signers	9	9	9
Duration [hours]	8.88	0.84	0.99
Frames	799,006	75,186	89,472
Sentences	5,672	540	629
Running glosses	65,227	5,540	6,504
Vocabulary	1,081	467	500

Table 1. Statistics of Phoenix-2014 data set.

	Train	Dev	Test
Signers	8	1	1
Duration [hours]	6.80	0.18	0.30
Frames	612,027	16,460	26,891
Sentences	4,376	111	180
Running glosses	49,966	1,167	1,901
Vocabulary	1,081	239	294

Table 2. Statistics of Phoenix-2014 Signer Independent SI5 data set.

hand patches. As mentioned earlier, we opt for GoogLeNet. which is initialised by weights learned on the ImageNet data set and trained on tracked right hand patches using an alignment (labels) generated from the approach of [22].

As shown in Table 3 after the first re-alignment, we see an improvement of 1.7 and 0.9 percentage points on the dev and test sets, respectively. However, a next re-alignment iteration seems to have a smaller impact with an improvement of 0.4 percentage point in WER on the test set. This initial significant improvements over the state of the art suggest using re-alignments within a hybrid DNN-HMM framework can be quite beneficial.

Considering the multimodal nature of sign language, we know that using the right hand alone will not lead to the best possible result. Hence, similar experiments are conducted using full frames which contain visual information of all sign language modalities (e.g. Hands, face and etc.). Once again, Table 3 shows the results of our experiments using full frames. We can see that already in the first iteration full frame images outperform the state of the art uni-modal hand model by 3.9 and 4.7 percentage points on the dev and test sets, respectively. It is important to note, that when using right hand patches as features, there is the need for an additional tracking step. This is in contrast to the full frame experiment, in which the CNN is not only able to distinguish the hands on its own, but also recognise other modalities which leads to better results.

Furthermore, following the same pattern as the right hand experiments, re-alignments using full frames lead to an initial improvement in the second iteration and a stabilisation in the third iteration. This confirms that re-alignments lead to improved performance, but also that the gain is limited for a few iterations. Considering both the relative simplicity and increased performance of full frames, they should be the feature of choice for sign language recognition in the hybrid DNN-HMM approach.

LSTM layers	**Input**	**Re-Alignment Iteration**		
		1	**2**	**3**
0	Right hand	38.3	36.6	36.9
0	Full frame	33.7	30.7	29.0

Table 3. Recognition results in WER [%] (the lower the better) with different numbers of re-alignments using GoogleNet structure on PHOENIX 2014 Dev for full frame & tracked right hand.

4.2. Temporal context through LSTMs

LSTMs are powerful units at capturing sequential and temporal information within deep neural networks. In the context of computer vision, the use of such units is most often done in isolation and on top of features learned by a separately trained CNN. In this work however, we opt for an end-to-end training of a deep CNN-LSTM network. This results in a network consisting of many convolutional layers followed by one or more LSTMs layers at the top. In an initial experiment, a single LSTMs layer is stacked on top of the last pooling layer of GoogLeNet and followed by a final softmax classifier.

4.3. Pre-training and LSTM initialization

As mentioned before, we initialise the network with weight learned on the ImageNet data set. However, there is no sequential information on that data set, making it impossible to pre-train LSTMs units on it. The impact of this can be seen in Table 4 where the CNN-LSTM architecture is initialised with weights of a CNN (GoogLeNet) only training on ImageNet. Without any re-alignments, there is an almost 11 percentage point deterioration in WER. However, after only two re-alignment iterations, the WER of the CNN-BLSTM network reaches that of a CNN-only network.

LSTM layers	Pre-Training	Re-Alignment Iteration		
		1	2	3
0	ImageNet	33.7	30.7	29.0
1	ImageNet	44.2	36.9	33.8
1	Phoenix-2014	33.8	29.4	29.5

Table 4. CNN-LSTM GoogleNet structure with pre-training on ImageNet or on Phoenix-2014 (CNN-only) across several iterations of re-alignment. Recognition results in WER [%] (the lower the better) on PHOENIX 2014 Dev using full frame images.

Despite the improvement from using re-alignments, we want to keep the number of necessary re-alignments as low as possible. One way of addressing this would be to use the weights of training a CNN-only network on the same data set. So initially, a CNN-only GoogLeNet is initialised by weights learned on the ImageNet data set and trained on PHOENIX 2014. The resulting weights are then used to initialise the CNN-LSTM model. Table 4, shows that this results in WERs on par with the CNN-only model. After one re-alignment step the CNN-LSTM model outperforms the same iteration of CNN-only model by 1.3 percentage points in WER. Despite this, the CNN-only model reaches a better WER by the third iteration.

4.4. LSTM vs. BLSTM vs. 2BLSTM

The initial end-to-end CNN-LSTM experiment shows that there is gain to be made from using LSTM layers, however there is a need for further investigation of the exact LSTM configuration. A LSTM unit has access to information from the current sequence position as well as the preceding features. In contrast, a bi-directional LSTM (BLSTM) unit provides access to the upcoming sequence information too. This is possible by fusion of two LSTM units, one of which processes the sequence from the beginning and towards the end, while the other does the same from the end and towards the beginning. This way at each time-step the BLSTM unit has access to both preceding and upcoming data. We compare the CNN-LSTM architecture with both CNN-BLSTM (a single BLSTM layer) and CNN-2BLSTM (two consecutive BLSTM layers). The CNN-BLSTM has the worse performance compared to CNN-only and other LSTM based experiments. The CNN-2BLSTM method, on the other hand, outperforms all other approaches on all iterations, bringing the WER down to 27.1 on the dev set of the PHOENIX 2014 data set.

LSTM layers	Bi-Direct.	Re-Alignment Iteration			
		1	2	3	4
0	-	33.7	30.7	29.0	29.1
1	no	33.8	29.4	29.5	29.7
1	yes	34.4	30.2	30.2	30.0
2	yes	32.7	29.5	27.1	27.2

Table 5. CNN-LSTM GoogleNet structure pre-trained on Phoenix-2014 (CNN-only) with varying LSTM or BLSTM layers across several iterations of re-alignment. Recognition results in WER [%] (the lower the better) on PHOENIX 2014 Dev with full frames.

4.5. LSTM size

Additional experiments were also conducted to determine the influence of the number of LSTMs neurons. So far, all of our experiments used LSTM layers with 1024 neurons. Using more than 1024 is not feasible considering the memory consumption of the setup used for our experiments. However, we were able to repeat the CNN-2BLSTM setup using LSTM layers with 512 neurons, the result of which is shown in Table 5. As can be seen, even though the 512 neuron model manages to outperform CNN-only experiments, the lower number of neurons leads to inferior WERs on almost all iterations.

BLSTM layers	Number of Hidden Units	Re-Alignment Iteration			
		1	2	3	4
2	512	32.8	29.2	27.9	28.5
2	1024	32.7	29.5	27.1	27.2

Table 6. CNN-BLSTM GoogleNet structure pre-trained on Phoenix-2014 (CNN-only) with two layers and varying number of hidden units per layer across several iterations of re-alignment. Recognition results in WER [%] (the lower the better) on PHOENIX 2014 Dev with full frames.

4.6. Signer Independent Recognition

In the scope of this work, we are presenting signer independent experiments on PHOENIX 2014, where we test on a single individual which has not been seen during training. As there are no previous alignments available for this task, we start by linearly segmenting our data set. Fig. 3 shows the WER as a function of the training iterations. After 10 iterations, the algorithm has converged and we reach

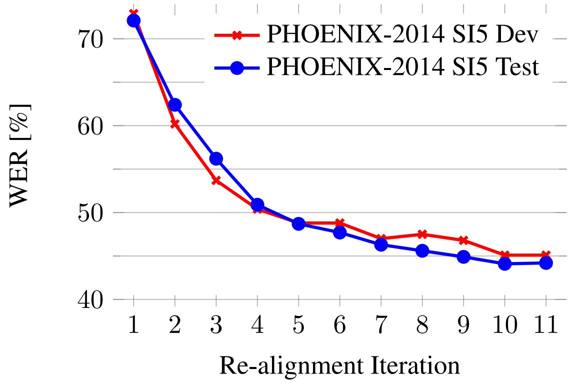

Figure 3. Showing signer independent recognition results in WER [%] (the lower the better) for signer 5 on the RWTH-PHOENIX-Weather 2014 SI5 corpus. The employed CNN-2BLSTM hybrid system is initialised by linearly segmenting the corpus data.

a WER of 45.1% on Dev and 44.1% on Test. Comparing these numbers to the best error rates on the PHOENIX 2014 multisigner, being 27.1% Dev and 26.8% Test, we note that the signer independent setting poses a much more difficult problem. Multisigner performance is nearly 20% absolute better than signer independent recognition.

4.7. Generalisability

Given the success of the presented approach, we conducted the same experiments on another sign language data set. SIGNUM [39] is a medium sized data set of sentences performed by a single signer in a controlled environment. Extra effort has been made by the signer to make all gestures and movements visually visible and easily understandable. This makes recognition of sentences in this data set easier compared to other data sets containing real-world data. A 3-gram language model with a perplexity of 97.6 is trained on and used for experiments on this data set. All sentences in this data set are pre-arranged, leading to relatively strong language model. Note, that the state-of-the-art WER on this is 7.4%.

	Train	**Test**
Signers	1	1
Duration [hours]	3.85	1.05
Frames	416,620	114,230
Sentences	1,809	531
Running glosses	11,109	2,805
Vocabulary	455	-

Table 7. Statistics on SIGNUM data set. OOV refers to words not found in the train sets vocabulary

Once again we can see that using full frame images leads to a 1.7 percentage point improvement in WER (compared to [23]). This is increased by applying re-alignments leading to a WER which is 2.4 percentage points better than the state of the art. Same as before, re-alignments lead to an improvement but the results stabilise after a few iterations. In the third iteration, the CNN-2BLSTM setup achieves an improvement of 0.5% WER absolute and nearly 10% relative over the CNN-only architecture.

LSTM layers	**Bi-Direct.**	**Re-Alignment Iteration**		
		1	**2**	**3**
0	-	5.7	5.0	5.3
2	yes	6.5	5.0	4.8

Table 8. CNN-only versus CNN-2BLSTM GoogleNet structure pre-trained on SIGNUM (CNN-only) across several iterations of re-alignment. Recognition results in WER [%] (the lower the better) on SIGNUM single singer using full frames.

4.8. Overview

Table 9 shows a comparison of our results to the state-of-the-art. As can be seen, the iterative use of re-alignments is instrumental to achieving the best possible WER on both CNN-only and CNN-2BLSTM methods. Furthermore, additional improvements are gained by incorporating BLSTM units leading to an end-to-end CNN-2BLSTM architecture. On the PHOENIX 2014 data set, our approach outperforms the state-of-the-art on both dev and test sets by up to 5.9 percentage points absolute or 15.2% relative without re-alignments and 12.0 percentage points absolute or 30.9% relative with re-alignments. Similarly, the state-of-the-art on SIGNUM is improved by 2.4 percentage points absolute or 32.4% relative. The results 'CNN' and 'CNN-2BLSTM' without any re-alignments constitute performances with legacy-style GMM-HMM alignments.

5. Conclusion

This work presents an iterative re-alignment algorithm based on a hybrid CNN-BLSTM embedded into a HMM, which is applicable to visual sequence labelling tasks such as gesture recognition, activity recognition and continuous sign language recognition. In this work, we empirically validate the importance of such re-alignments for continuous gesture and sign language recognition tasks. Because of this, we are able to successfully train end-to-end CNN-BLSTMs for challenging real-life continuous gesture and sign language tasks distinguishing over 1000 classes. To the best of our knowledge, we are the first to achieve this in the context of large vocabulary sign language or gesture recognition. We evaluate on two challenging publicly available sign recognition benchmark data sets featuring over 1000 classes. We outperform the state-of-the-art by up to 10% absolute and 30% relative. Embedded into a HMM, the resulting deep model corrects the frame labels and continuously

	PHOENIX 2014		SIGNUM
	Dev	Test	Test
[39]	–	–	12.7
[18]	–	–	11.9
[13]	–	–	10.7
[21]	57.3	55.6	10.0
[22]	47.1	45.1	7.6
[23]	38.3	38.8	7.4
CNN	33.7	33.3	5.7
CNN re-aligned	29.0	29.4	5.0
CNN-2BLSTM	32.7	32.9	6.5
CNN-2BLSTM re-aligned	27.1	26.8	4.8

Table 9. Overview of presented approach against best published results. Continuous sign language recognition results in WER [%] (the lower the better) on PHOENIX 2014 Multisigner and SIGNUM. CNN-2BLSTM refers to a CNN jointly trained with 2 layers of bidirectional LSTMs.

improves its performance in several re-alignments. Further, we find that whole frame images outperform tracked hands, which used to be the method of choice until now. In terms of future work, it could be promising to compare the algorithm to connectionist temporal classification. Also, more work tackling signer independency is needed.

References

[1] Y. M. Assael, B. Shillingford, S. Whiteson, and N. de Freitas. LipNet: End-to-End Sentence-level Lipreading. *arXiv:1611.01599 [cs]*, Nov. 2016. 2

[2] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Sequential Deep Learning for Human Action Recognition. In *Proceedings of the Second International Conference on Human Behavior Unterstanding*, HBU'11, pages 29–39, Berlin, Heidelberg, 2011. Springer-Verlag. 2

[3] Y. Bengio. A connectionist approach to speech recognition. *International journal of pattern recognition and artificial intelligence*, 7(04):647–667, 1993. 2

[4] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994. 4

[5] T. Bluche, H. Ney, J. Louradour, and C. Kermorvant. Framewise and CTC training of Neural Networks for handwriting recognition. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 81–85, Aug. 2015. 2

[6] H. A. Bourlard and N. Morgan. *Connectionist Speech Recognition: A Hybrid Approach*, volume 247. Springer Science & Business Media, 1994. 2

[7] K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio. On the Properties of Neural Machine Translation: Encoder–Decoder Approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar, Oct. 2014. Association for Computational Linguistics. 2

[8] J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman. Lip Reading Sentences in the Wild. *arXiv:1611.05358 [cs]*, Nov. 2016. 2

[9] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977. 4

[10] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-Term Recurrent Convolutional Networks for Visual Recognition and Description. pages 2625–2634, 2015. 2

[11] P. Dreuw, T. Deselaers, D. Rybach, D. Keysers, and H. Ney. Tracking Using Dynamic Programming for Appearance-Based Sign Language Recognition. In *IEEE International Conference Automatic Face and Gesture Recognition*, pages 293–298, Southampton, UK, Apr. 2006. IEEE. 4

[12] Y. Du, W. Wang, and L. Wang. Hierarchical Recurrent Neural Network for Skeleton Based Action Recognition. pages 1110–1118, 2015. 2

[13] J. Forster, C. Oberdörfer, O. Koller, and H. Ney. Modality Combination Techniques for Continuous Sign Language Recognition. In *Iberian Conference on Pattern Recognition and Image Analysis*, Lecture Notes in Computer Science 7887, pages 89–99, Madeira, Portugal, June 2013. Springer. 8

[14] J. Forster, C. Schmidt, O. Koller, M. Bellgardt, and H. Ney. Extensions of the Sign Language Recognition and Translation Corpus RWTH-PHOENIX-Weather. In *Language Resources and Evaluation*, pages 1911–1916, Reykjavik, Island, May 2014. 4

[15] A. Graves and N. Jaitly. Towards End-To-End Speech Recognition with Recurrent Neural Networks. pages 1764–1772, 2014. 2

[16] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, and J. Schmidhuber. A Novel Connectionist System for Unconstrained Handwriting Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(5):855–868, May 2009. 2

[17] A. Graves and J. Schmidhuber. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5):602–610, 2005. 4

[18] Y. Gweth, C. Plahl, and H. Ney. Enhanced Continuous Sign Language Recognition using PCA and Neural Network Features. In *CVPR 2012 Workshop on Gesture Recognition*, pages 55–60, Providence, Rhode Island, USA, June 2012. 8

[19] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 2

[20] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional Architecture for Fast Feature Embedding. *arXiv preprint arXiv:1408.5093*, 2014. 4

[21] O. Koller, J. Forster, and H. Ney. Continuous sign language recognition: Towards large vocabulary statistical recognition

systems handling multiple signers. *Computer Vision and Image Understanding*, 141:108–125, Dec. 2015. 8

[22] O. Koller, H. Ney, and R. Bowden. 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 2016. 4, 5, 8

[23] O. Koller, S. Zargaran, H. Ney, and R. Bowden. Deep Sign: Hybrid CNN-HMM for Continuous Sign Language Recognition. In *British Machine Vision Conference*, York, UK, Sept. 2016. 2, 4, 7, 8

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, pages 1106–1114, 2012. 4

[25] H.-S. Le, N.-Q. Pham, and D.-D. Nguyen. Neural Networks with Hidden Markov Models in Skeleton-Based Gesture Recognition. In V.-H. Nguyen, A.-C. Le, and V.-N. Huynh, editors, *Knowledge and Systems Engineering*, number 326 in Advances in Intelligent Systems and Computing, pages 299–311. Springer International Publishing, 2015. 2

[26] G. Lefebvre, S. Berlemont, F. Mamalet, and C. Garcia. BLSTM-RNN based 3D gesture classification. In *International Conference on Artificial Neural Networks*, pages 381–388. Springer, 2013. 2

[27] M. Liwicki, A. Graves, H. Bunke, and J. Schmidhuber. A novel approach to on-line handwriting recognition based on bidirectional long short-term memory networks. In *In Proceedings of the 9th International Conference on Document Analysis and Recognition, ICDAR 2007*, 2007. 4

[28] J. Y.-H. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond Short Snippets: Deep Networks for Video Classification. pages 4694–4702, 2015. 2

[29] L. Pigou, A. van den Oord, S. Dieleman, M. Van Herreweghe, and J. Dambre. Beyond Temporal Pooling: Recurrence and Temporal Convolutions for Gesture Recognition in Video. *arXiv:1506.01911 [cs, stat]*, June 2015. 2

[30] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252, Dec. 2015. 4

[31] D. Rybach, S. Hahn, P. Lehnen, D. Nolden, M. Sundermeyer, Z. Tüske, S. Wiesler, R. Schlüter, and H. Ney. RASR - The RWTH Aachen University Open Source Speech Recognition Toolkit. In *IEEE Automatic Speech Recognition and Understanding Workshop*, Waikoloa, HI, USA, Dec. 2011. 3

[32] H. Sak, A. W. Senior, and F. Beaufays. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *INTERSPEECH*, pages 338–342, 2014. 2

[33] A. Senior, G. Heigold, M. Bacchiani, and H. Liao. GMM-free DNN training. In *Proceedings of ICASSP*, pages 5639–5643, 2014. 2

[34] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 4

[35] M. Sundermeyer, R. Schlüter, and H. Ney. LSTM Neural Networks for Language Modeling. In *INTERSPEECH*, pages 194–197, 2012. 2

[36] I. Sutskever, O. Vinyals, and Q. Le. Sequence to Sequence Learning with Neural Networks. In *Advances in Neural Information Processing Systems*, pages 3104–3112, 2014. 2

[37] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going Deeper With Convolutions. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, Boston, Ma, USA, June 2015. 4

[38] E. Tsironi, P. Barros, and S. Wermter. Gesture Recognition with a Convolutional Long Short-Term Memory Recurrent Neural Network. Bruges, Belgium, 2016. 2

[39] U. von Agris, M. Knorr, and K.-F. Kraiss. The significance of facial features for automatic sign language recognition. In *Automatic Face & Gesture Recognition, 2008. FG'08. 8th IEEE International Conference On*, pages 1–6. IEEE, 2008. 7, 8

[40] R. J. Williams and D. Zipser. Gradient-based learning algorithms for recurrent networks and their computational complexity. *Back-propagation: Theory, architectures and applications*, pages 433–486, 1995. 4

[41] D. Wu, L. Pigou, P.-J. Kindermans, N. LE, L. Shao, J. Dambre, and J.-M. Odobez. Deep Dynamic Neural Networks for Multimodal Gesture Segmentation and Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP(99):15, 2016. 2

[42] J. Xu, T. Mei, T. Yao, and Y. Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016. 2

[43] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville. Describing Videos by Exploiting Temporal Structure. pages 4507–4515, 2015. 2

[44] A. Zeyer, P. Doetsch, P. Voigtlaender, R. Schlüter, and H. Ney. A Comprehensive Study of Deep Bidirectional LSTM RNNs for Acoustic Modeling in Speech Recognition. *arXiv:1606.06871 [cs]*, June 2016. 2