CONTINUOUS SIGN LANGUAGE RECOGNITION VIA REINFORCEMENT LEARNING

Zhihao Zhang, Junfu Pu, Liansheng Zhuang, Wengang Zhou, Houqiang Li

CAS Key Laboratory of Technology in Geo-spatial Information Processing and Application System
EEIS Department, University of Science and Technology of China
{zzh3, pjh}@mail.ustc.edu.cn, {lszhuang, zhwg, lihq}@ustc.edu.cn

ABSTRACT

In this paper, we propose an approach to apply the Transformer with reinforcement learning (RL) for continuous sign language recognition (CSLR) task. The Transformer has an encoder-decoder structure, where the encoder network encodes the sign video into the context vector representation, while the decoder network generates the target sentence word by word based on the context vector. To avoid the intrinsic defects of supervised learning (SL) in our task, *e.g.*, the exposure bias and non-differentiable task metrics issues, we propose to train the Transformer directly on non-differentiable metrics, *i.e.*, word error rate (WER), through RL. Moreover, a policy gradient algorithm with baseline, which we call Self-critic REINFORCE, is employed to reduce variance while training. Experimental results on RWTH-PHOENIX-Weather benchmark verify the effectiveness of our method and demonstrate that our method achieves the comparable performance.

Index Terms— sign language recognition, reinforcement learning, self-critic

1. INTRODUCTION

Millions of hearing-impaired people routinely use some variants of sign languages to communicate, however, it's difficult to understand sign language for the hearing society. As a result, there is a huge communication disorder between the deaf-mute and the hearing people, which makes the automatic translation of sign language meaningful and important.

Continuous sign language recognition (CSLR) aims at translating sign videos into text sentences. Though significant progress [1, 2, 3, 4] has been made, CSLR is still a very challenging task. It requires a fine-grained understanding of gestures, hand motions or even facial expressions in a video. Meanwhile, there exist semantic gaps between videos and

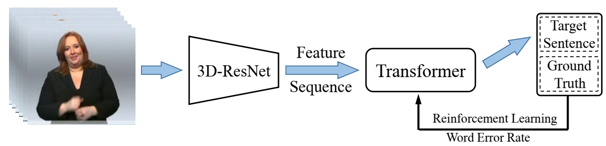

Fig. 1. An overview of the proposed method. We extract sign language features from video using 3D-ResNet. The Transformer translates the feature sequence to the target sentence. We train the Transformer directly on word error rate (WER) through RL.

sentences, as well as the difficulty of frame or word level alignment. To solve these challenges, we propose our CSLR model, as shown in Fig. 1. First, we adopt a 3D convolutional neural network to extract visual features from sign videos. Recently, residual network (ResNet) [5] and 3D convolutional neural network (3D CNN [6, 7, 8]) have shown outstanding performance in image and video representation, respectively. Inspired by the superiorities of ResNet and 3D CNN, we employ a combined 3D residual convolutional neural network (3D-ResNet) for feature extraction following CNN-DCN [9]. Second, we utilize a powerful neural machine translation (NMT) model to translate sign videos into text sentences. Recently, the Transformer [10], the first sequence transduction model based entirely on attention, achieves state-of-the-art performance on the English-German and English-French translation tasks. Considering the similarity between NMT and CLSR, we adopt the Transformer to bridge the semantic gap between sign videos and text sentences.

However, the Transformer [10] (or LSTM [11], *etc.*) for sequence transduction is typically trained to maximize the likelihood of the next ground-truth word given the previous ground-truth word using error back-propagation. This approach suffers a mismatch between training and testing since at test-time the model uses the previously generated words from the model distribution to predict the next word. This *exposure bias* [12] results in error accumulation during generation at the test time, since the model has never been exposed to its own predictions. Besides, there exist deviation between optimization objectives, *i.e.*, the cross-entropy loss, during training and the non-differentiable evaluation metrics during testing. Recently, it has been shown that both the exposure bias and non-differentiable task metrics issues can be addressed through reinforcement learning (RL). Motivated

This work was supported in part to Dr. Houqiang Li by the 973 Program under Contract No. 2015CB351803 and NSFC under contract No. 61836011, in part to Dr. Liansheng Zhuang by NSFC under contract No. 61472379, and in part to Dr. Wengang Zhou by NSFC under contract No. 61632019 and Young Elite Scientists Sponsorship Program By CAST (2016QNRC001).

by these works [13, 14, 15], we employ REINFORCE [16] to train our CSLR model. Moreover, we append a baseline to REINFORCE to form a self-critic architecture because of the high variance of REINFORCE.

In summary, our major contributions are listed as follows:

- We propose a novel framework based on 3D-ResNet and the Transformer for continuous sign language recognition (CSLR). To the best of our knowledge, we are the first to deploy the Transformer for sequence learning in CSLR.

- We introduce an RL-based optimization strategy for our CSLR model. Experiments on the RWTH-PHOENIX-Weather demonstrate the effectiveness of our approach.

2. RELATED WORK

In this section, we briefly review some continuous sign language recognition (CSLR) methods, and compactly introduce some sequence generation tasks which are closely related to our work.

CNN-LSTM based methods [17, 18] are very popular for continuous sign language recognition (CSLR). Recently, some works such as [19, 20] have employed a CNN-LSTM network with connectionist temporal classification (CTC) [21] for CSLR, since CSLR task lacks supervision on accurate temporal segmentation for sign words. In addition, there are some approaches for CSLR which are based on other sequential models. Re-Sign [22] embeds a hidden markov model (HMM) into a deep recurrent CNN-BLSTM network with an iterative re-alignment approach for CSLR. CNN-DCN [9] proposes a deep neural architecture composed by 3D-ResNet and dilated convolutional network [23] with CTC loss for CSLR. Similarly, we employ the Transformer [10] as the sequential model instead of LSTM to solve the CSLR task.

Neural machine translation (NMT) is a typical sequence learning task and has drawn much attention. Recently, the Transformer [10] has achieved the state-of-the-art results on both WMT2014 English-German and English-French translation tasks. Moreover, BR-CSGAN [24] proposes an approach for applying GANs to NMT with the Transformer through policy gradient methods. Actually, reinforcement learning (RL) algorithms are widely used among sequence learning tasks. MIXER [12] adopts the REINFORCE [16] algorithm for text generation applications. Since REINFORCE suffers from high variance, it requires a proper baseline. Therefore, Bahdanau *et al.* [14] train another critic network to predict the value of an output token. SCST [13] utilizes the output of its own test-time inference algorithm to normalize the rewards it experiences. Inspired by these works, we utilize REINFORCE with a baseline to train our model.

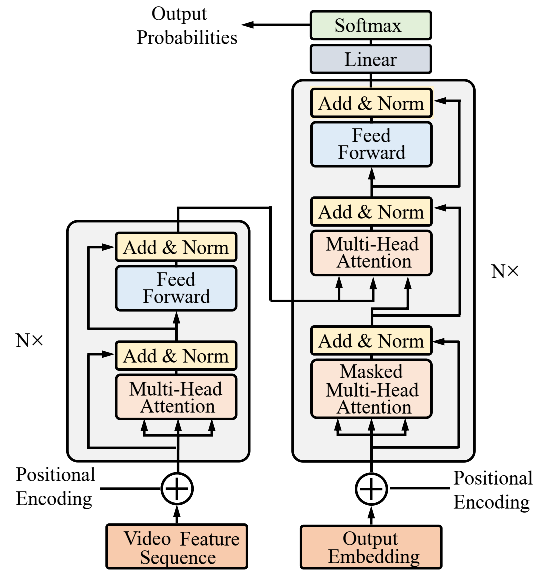

Fig. 2. The architecture of the Transformer [10].

3. OUR METHODS

In this section, we first propose a novel architecture based on the Transformer for continuous sign language recognition (CSLR). Then, we introduce our Self-critic REINFORCE for network training in our CSLR model.

3.1. Model Architecture

Our CSLR model consists of two components: 3D-ResNet for extracting video clip feature, and the Transformer which translates the visual feature sequences into sentences.

3D-ResNet. It is a great challenge to extract sematic information of sign language, which is contained in those elements of gestures, hand motions or even facial expressions in videos, for CSLR. Fortunately, 3D CNN has shown strong capability for video representation based on spatio-temporal information, since it considers the sequential relationship by temporal connections across frames. Following CNN-DCN [9], we adopt the 18-layers 3D-ResNet, which only replaces the 2D convolutional filters with 3D convolutional filters, for feature extraction. Furthermore, the training method for 3D-ResNet is the same as that introduced in CNN-DCN [9] as well.

The Transformer. The Transformer [10], as shown in Fig. 2, with an encoder-decoder architecture, has shown strong capability in neural machine translation (NMT). The encoder of the Transformer is composed of a stack of N identical layers. Each layer consists of a multi-head self-attention and a simple position-wise fully connected feed-forward network. The decoder is also composed of a stack of N identical layers. In addition to the two sub-layers in each encoder layer, the decoder inserts a third sub-layer, which performs multi-head attention over the output of the encoder stack. In the CSLR task, the Transformer regards the sign videos as source language and translates the source language to the target language (*i.e.*, text sentences). The input of the encoder is the video clip feature sequence extracted by 3D-ResNet. The input of the decoder is the learned embedding of a sentence or the beginning token of a sentence, while the decoder outputs the predicted next-token probabilities.

3.2. Self-critic REINFORCE

CSLR as a RL problem. Generally, the Transformer is trained using the cross-entropy loss. However, there exists deviation between the cross-entropy loss and the non-differentiable evaluation metrics, *i.e.*, WER. To directly optimize the WER metric, we cast our models in the reinforcement rearning (RL) terminology. The Transformer, which can be viewed as an "agent", defines a policy p_θ. The "agent" consistently produces the "action", *i.e.*, the prediction of the next token. We define the immediate reward $r = 0$ until the end-of-sequence (EOS) token generates. And "$1 - WER$" is received as the terminal reward denoted by R. To minimize the negative expected cumulative reward with the discount rate $\lambda = 1$, we formulate the goal as follows,

$$L(\theta) = -\mathbb{E}_{\omega^s \sim p_\theta}[R(\omega^s)], \tag{1}$$

where $\omega^s = (\omega_1^s, \cdots, \omega_T^s)$ and ω_t^s is the word sampled from the model at the time step t.

REINFORCE with baseline. According to [25], we compute the gradient $\nabla L(\theta)$,

$$\nabla L(\theta) = -\mathbb{E}_{\omega^s \sim p_\theta}[R(\omega^s)\nabla_\theta \log p_\theta(\omega^s)]. \tag{2}$$

We use samples of the expectation to instantiate our generic stochastic gradient ascent algorithm,

$$\nabla L(\theta) \approx -R(\omega^s)\nabla_\theta \log p_\theta(\omega^s). \tag{3}$$

This algorithm is called REINFORCE [16]. In addition, the policy gradient given by REINFORCE can be generalized to include a comparison of the action value $R(\omega^s)$ to an arbitrary baseline b as long as it does not depend on the "action" ω^s,

$$\nabla L(\theta) = -\mathbb{E}_{\omega^s \sim p_\theta}[(R(\omega^s) - b)\nabla_\theta \log p_\theta(\omega^s)]. \tag{4}$$

For each training case, we again approximate the expected gradient with a single sample $\omega^s \sim p_\theta$,

$$\nabla L(\theta) \approx -(R(\omega^s) - b)\nabla_\theta \log p_\theta(\omega^s). \tag{5}$$

Self-Critic REINFORCE. The central idea of the Self-critic REINFORCE is to take the reward, which is obtained by the Transformer under the inference algorithm used at the test time, as the baseline for the REINFORCE. As shown in Fig. 3, $R(\omega^s)$ is the reward which represents the sentence (*i.e.*, $(\omega_1^s, \cdots, \omega_T^s)$) generated through sampling based on its probabilities. Similarly, $R(\hat{\omega})$ is the reward which evaluates

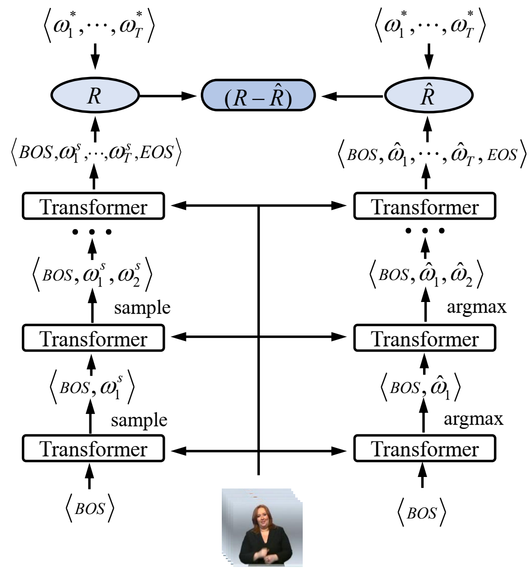

Fig. 3. The illustration of Self-critic REINFORCE. The difference between the reward for the sampled sentence and the reward obtained by the estimated sentence under the test-time inference procedure is used to update the weight of the sampled sentence.

the sentence (*i.e.*, $(\hat{\omega}_1, \cdots, \hat{\omega}_T)$) obtained by the Transformer under the inference algorithm used at test time, *i.e.*,

$$\hat{\omega}_t = \arg\max_{\omega_t} p(\omega_t). \tag{6}$$

For each training case, the Self-critic REINFORCE gives:

$$\nabla L(\theta) \approx -(R(\omega^s) - R(\hat{\omega}))\nabla_\theta \log p_\theta(\omega^s). \tag{7}$$

Self-critic REINFORCE inherits all the advantage of REINFORCE, as it not only directly optimizes the true, sequence-level evaluation metric but also avoids the usual scenario of having to learn a context-dependent estimate of expected future rewards as a baseline. Since the Self-critic REINFORCE baseline is based on the test-time estimation under the current model, Self-critic REINFORCE is forced to improve the performance of the model under the inference algorithm used at the test time. Besides, Self-critic REINFORCE avoids all the inherent training difficulties associated with actor-critic methods, where a second "critic" network must be trained to estimate value functions, and the actor must be trained on estimated value functions rather than actual rewards.

4. EXPERIMENTS

In this section, we first introduce the experiment setup. Besides, we discuss the comparison results as well as the ablation study.

Table 1. Summary of RWTH-PHOENIX-Weather dataset

	Train	Test	Dev
#Sentences	5672	629	540
#Vocabulary	1231	497	461
#Words	65227	6530	5564

4.1. Experiment Setup

Dataset. We conduct our experiments on the German sign language dataset, *i.e.*, RWTH-PHOENIX-Weather [26]. The dataset contains 7K weather forecast sentences from 9 signers. All videos are of 25 frames per second (FPS) and at resolution of 210×260. Following [26], 5,672 instances are used for training, 540 for validation, and 629 for testing. The statistic details of this dataset are available in Table 1.

Evaluation Metrics. Predicted sentence may suffer from errors including word substitution, insertion and deletion error. Following [18, 27, 28, 29], we measure the performance with word error rate (WER),

$$WER = \frac{S + I + D}{N} \times 100\%, \tag{8}$$

where S, I and D denote the minimum number of substitution, insertion and deletion operations needed to transform a hypothesized sentence to the ground truth. N is the number of words in ground truth.

Implementation Details. In our experiments, videos are divided into 8-frame clips with 50% overlap, with frames cropped and resized to 224×224. The output of our 3D-ResNet is a 512-dimensional vector, which represents the clip in sign video. We employ the Adam [30] optimization algorithm for the neural network training. In the Transformer, we apply dropout [31] to the output of each sub-layer, before it is added to the sub-layer input and normalized. Moreover, we apply dropout to the sums of the embeddings and the positional encodings in both the encoder and the decoder stacks. The dropout rate is set to 0.3. In addition, we employ label smoothing [32] with value $\epsilon_{ls} = 0.2$. This hurts perplexity, as the model learns to be more unsure, but improves the performance apparently.

4.2. Comparison with the State-of-the-art

In the subsection, we evaluate the performance of our method by comparing it to some existing algorithms on the RWTH-PHOENIX-Weather dataset. The results are summarized in Table 2. In this table, “ins” and “del” mean the average operations of “insertion” and “deletion” that transform the generated sentences into the sentences of ground-truth.

Compared with other methods, our SL-based model achieves a competitive performance with a lower value of “ins” and “del”, which means that the “sub” (“substitution”) is higher. This is due to the fact that the encoder component of the Transformer has the capacity to distinguish sign language signal from a sequence of video clip features exactly since the Transformer is based solely on attention mechanisms. However, the translation results do not exhibit a higher level of translation. It means that the decoder component of the Transformer does not complete the translation task perfectly regardless of the limitations of the Transformer or the accuracy of the features of video clips.

Table 2. Performance of the proposed method and some existing algorithms on RWTH-PHOENIX-Weather. “SL” represents that the model is trained by supervised learning. “RL” represents that we fine-tune the model by reinforcement learning

Methods	Dev(%)		Test(%)	
	del / ins	WER	del / ins	WER
Deep Hand [33]	16.3 / 4.6	47.1	15.2 / 4.6	45.1
SubUNet [19]	14.6 / 4.0	40.8	14.3 / 4.0	40.7
Deep Sign [34]	12.6 / 5.1	38.3	11.1 / 5.7	38.8
Recurrent CNN [20]	13.7 / 7.3	39.4	12.2 / 7.5	38.7
CNN-DCN [9]	8.3 / 4.8	38.0	7.6 / 4.8	37.3
LS-HAN [18]	-	-	-	38.3
Ours (SL)	**5.7** / 6.8	39.7	**5.8** / 6.8	40.0
Ours (RL)	7.3 / **5.2**	**38.0**	7.0 / **5.7**	**38.3**

As revealed from the results, our RL-based method achieves comparable performance. It’s worth mentioning that our model achieves the best performance on the metrics of “del” and “ins”. The attention mechanism of the Transformer is of great benefit to distinguish effective sign language signal from a sequence of video clip features.

4.3. Ablation Study

In this subsection, we analyze the experimental results of our proposed methods based on supervised learning (SL) and reinforcement learning (RL), respectively, to verify the effectiveness of RL.

To solve the exposure bias and the deviation between the optimization objective and the non-differentiable evaluation metrics using SL, we fine-tune our model through RL. As shown in Table 2, the performance of WER decreases by approximately 1.7% on both the Dev and the Test set. Besides, the RL-based model achieves a lower “ins” but a higher “del”. The “sub” decreases from 27.2% to 25.6% on Dev set and from 27.4% to 25.6% on Test set in the experiment, respectively. It means that our RL-based model can capture the sign language signal more accurately.

5. CONCLUSION

In this paper, we propose a deep learning framework composed of 3D-ResNet and the Transformer for continuous sign language recognition (CSLR). Besides, a policy-gradient reinforcement learning (RL) method, which is equipped with a baseline to reduce variance, is utilized to train our end-to-end system directly on the non-differentiable metrics, *i.e.*, word error rate (WER), and leads to performance gain on RWTH-PHOENIX-Weather dataset.

6. REFERENCES

[1] Chao Sun, Tianzhu Zhang, Bing-Kun Bao, and Changsheng Xu, "Latent support vector machine for sign language recognition with kinect," in *ICIP*, 2013.

[2] Binyam Gebrekidan Gebre, Peter Wittenburg, and Tom Heskes, "Automatic sign language identification," in *ICIP*, 2013.

[3] Marc Martínez-Camarena, MJ Oramas, and Tinne Tuytelaars, "Towards sign language recognition based on body parts relations," in *ICIP*, 2015.

[4] Jie Huang, Wengang Zhou, Houqiang Li, and Weiping Li, "Attention based 3d-cnns for large-vocabulary sign language recognition," *TCSVT*, 2018.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[6] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu, "3d convolutional neural networks for human action recognition," *TPAMI*, vol. 35, no. 1, pp. 221–231, 2013.

[7] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *ICCV*, 2015.

[8] Zhaofan Qiu, Ting Yao, and Tao Mei, "Learning spatio-temporal representation with pseudo-3d residual networks," in *ICCV*, 2017.

[9] Junfu Pu, Wengang Zhou, and Houqiang Li, "Dilated convolutional network with iterative optimization for continuous sign language recognition.," in *IJCAI*, 2018.

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," in *NIPS*, 2017.

[11] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[12] Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba, "Sequence level training with recurrent neural networks," *arXiv preprint arXiv:1511.06732*, 2015.

[13] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel, "Self-critical sequence training for image captioning," in *CVPR*, 2017.

[14] Dzmitry Bahdanau, Philemon Brakel, Kelvin Xu, Anirudh Goyal, Ryan Lowe, Joelle Pineau, Aaron Courville, and Yoshua Bengio, "An actor-critic algorithm for sequence prediction," *arXiv preprint arXiv:1607.07086*, 2016.

[15] Zhou Ren, Xiaoyu Wang, Ning Zhang, Xutao Lv, and Li-Jia Li, "Deep reinforcement learning-based image captioning with embedding reward," in *CVPR*, 2017.

[16] Ronald J Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine Learning*, vol. 8, no. 3-4, pp. 229–256, 1992.

[17] Dan Guo, Wengang Zhou, Houqiang Li, and Meng Wang, "Hierarchical lstm for sign language translation," in *AAAI*, 2018.

[18] Jie Huang, Wengang Zhou, Qilin Zhang, Houqiang Li, and Weiping Li, "Video-based sign language recognition without temporal segmentation," in *AAAI*, 2018.

[19] Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, and Richard Bowden, "Subunets: End-to-end hand shape and continuous sign language recognition," in *CVPR*, 2017.

[20] Runpeng Cui, Hu Liu, and Changshui Zhang, "Recurrent convolutional neural networks for continuous sign language recognition by staged optimization," in *CVPR*, 2017.

[21] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *ICML*, 2006.

[22] Oscar Koller, Sepehr Zargaran, and Hermann Ney, "Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent cnn-hmms," in *CVPR*, 2017.

[23] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[24] Zhen Yang, Wei Chen, Feng Wang, and Bo Xu, "Improving neural machine translation with conditional sequence generative adversarial nets," *arXiv preprint arXiv:1703.04887*, 2017.

[25] Richard S Sutton and Andrew G Barto, *Reinforcement learning: An introduction*, MIT press, 2018.

[26] Oscar Koller, Jens Forster, and Hermann Ney, "Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers," *CVIU*, vol. 141, pp. 108–125, 2015.

[27] Thad Starner, Joshua Weaver, and Alex Pentland, "Real-time american sign language recognition using desk and wearable computer based video," *TPAMI*, vol. 20, no. 12, pp. 1371–1375, 1998.

[28] Jihai Zhang, Wengang Zhou, and Houqiang Li, "A threshold-based hmm-dtw approach for continuous sign language recognition," in *ICIMCS*, 2014.

[29] Junfu Pu, Wengang Zhou, and Houqiang Li, "Iterative alignment network for continuous sign language recognition.," in *CVPR*, 2019.

[30] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[31] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *JMLR*, vol. 15, no. 1, pp. 1929–1958, 2014.

[32] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna, "Rethinking the inception architecture for computer vision," in *CVPR*, 2016.

[33] Oscar Koller, Hermann Ney, and Richard Bowden, "Deep hand: How to train a cnn on 1 million hand images when your data is continuous and weakly labelled," in *CVPR*, 2016.

[34] Oscar Koller, O Zargaran, Hermann Ney, and Richard Bowden, "Deep sign: hybrid cnn-hmm for continuous sign language recognition," in *BMVC*, 2016.