Particle Filter based Probabilistic Forced Alignment for Continuous Gesture Recognition

Necati Cihan Camgoz
University of Surrey
Guildford, UK
n.camgoz@surrey.ac.uk

Simon Hadfield
University of Surrey
Guildford, UK
s.hadfield@surrey.ac.uk

Richard Bowden
University of Surrey
Guildford, UK
r.bowden@surrey.ac.uk

Abstract

In this paper, we propose a novel particle filter based probabilistic forced alignment approach for training spatio-temporal deep neural networks using weak border level annotations.

The proposed method jointly learns to localize and recognize isolated instances in continuous streams. This is done by drawing training volumes from a prior distribution of likely regions and training a discriminative 3D-CNN from this data. The classifier is then used to calculate the posterior distribution by scoring the training examples and using this as the prior for the next sampling stage.

We apply the proposed approach to the challenging task of large-scale user-independent continuous gesture recognition. We evaluate the performance on the popular ChaLearn 2016 Continuous Gesture Recognition (ConGD) dataset. Our method surpasses state-of-the-art results by obtaining 0.3646 *and* 0.3744 *Mean Jaccard Index Score on the validation and test sets of ConGD, respectively. Furthermore, we participated in the ChaLearn 2017 Continuous Gesture Recognition Challenge and was ranked 3rd. It should be noted that our method is learner independent, it can be easily combined with other approaches.*

1. Introduction

In recent years, Deep Learning [17] methods have obtained state-of-the-art performance for various spatio-temporal computer vision tasks, such as, Gesture Recognition [30], Action Recognition [5], Video Captioning [29], Lip Reading [9], and Sign Language Recognition [25]. However, due to their hierarchical structure and high number of parameters, deep neural networks are prone to over-fitting [33]. To be able to generalize, they need vast amounts of data. Furthermore, classical network architectures and loss functions require strong *Frame Level Annotations* [26], or in other words a label for each time-step. Unfortunately, annotating spatio-temporal data is a laborious task and most

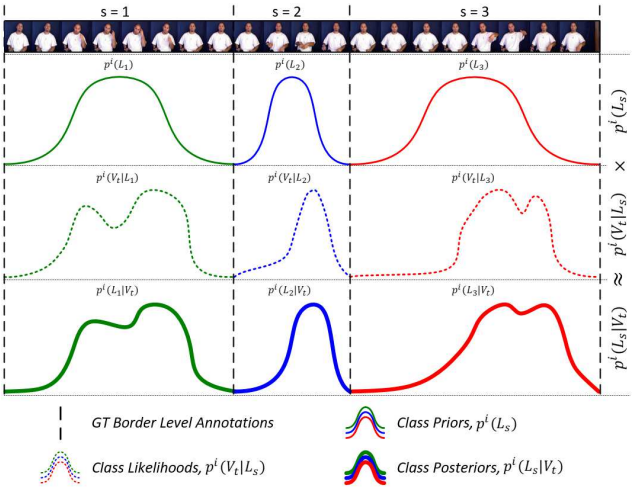

Figure 1. Posterior estimation using our prior assumptions and class likelihoods extracted from a 3D-CNN.

spatio-temporal datasets lack these strong annotations. Although some large datasets have frame level annotations [14], the majority provide a variety of weak annotations. To overcome these challenges and to train deep spatio-temporal neural networks using weakly annotated data, researchers have proposed several approaches.

For isolated recognition tasks such as Isolated Gesture [10] and Action Recognition [23], most datasets provide *Instance Level Annotations* that is a single label for each video clip which does not contain any temporal localisation. To train deep networks using instance level annotations, researchers [11, 21, 27, 28] frequently assign the provided instance labels to all time steps and train neural networks using Cross Entropy Loss [17]. However, identifying every part of a sequence with the same label can cause class ambiguity as different stages of a sequence can have different spatio-temporal features.

For continuous recognition tasks, datasets with weak annotation commonly provide *Sequence Level Annotations*

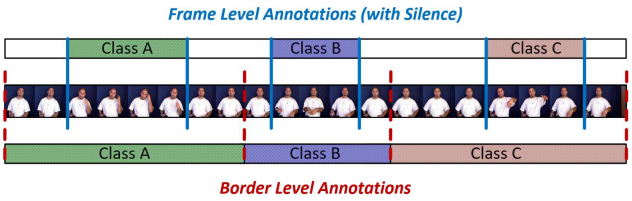

Figure 2. Difference between Strong Frame Level Annotations and Border Level Annotations.

that have a sequence of labels for each video that has the temporal ordering information [15]; or *Border Level Annotations* which are the boundaries between instances in the time domain [31].

Sequence Level Annotations are frequently used for tasks where the order of labels provides additional or complementary information. Such applications include Continuous Sign Language Recognition [4] and Video Captioning [7]. To be able to train spatio-temporal deep networks using sequence level annotations, researchers adopted sequence-to-sequence learning methods from other fields, namely Connectionist Temporal Classification [18] from Speech Recognition [19] and Encoder-Decoder Networks [8] from the field of Neural Machine Translations [1].

Compared to the Frame Level Annotations, *Border Level Annotations* are more weakly supervised but easier to annotate (See Figure 2). However, using these weak frame level annotations can cause the aforementioned class ambiguity problem. Furthermore, this ambiguity is exacerbated as the annotated regions include the silence and the transition between instances. To localize instances in the time domain and to remove silence and transition regions from training samples, researchers have proposed to use forced alignment that has been widely used in speech recognition [12].

Forced alignment has been applied to various vision applications. In [24], Koller et al. propose a CNN-HMM hybrid that learns to localize and recognize hand shapes. They first train a CNN using weak frame level annotations. Then using the trained network they reassign, or in other words align, their frame labels. In a more recent study [25], they extend their method and apply it to Continuous Sign Language recognition to train CNN-HMM hybrids from sequence level annotations.

Inspired by the success of forced alignment approaches, we propose a novel probabilistic forced alignment method to address the weakly supervised training of spatio-temporal deep neural networks. Our method learns to probabilistically localize and recognize isolated instances in continuous streams using a particle filter based approach. We use weak border level annotations to build distributions for each isolated instance that represent our prior assumption of importance over the time domain. We then train a 3D-CNN weighted by these distributions. We evaluate our prior distributions by extracting likelihoods of training samples using the trained network and update our prior assumption. We repeat this iterative process until the continuous recognition performance of our network has converged. Compared with a classical forced alignment method, our approach is more robust against errors in early stages due to its probabilistic nature.

We apply the proposed method to the continuous gesture recognition problem and evaluate its performance on the challenging large-scale user-independent ChaLearn 2016 Continuous Gesture Recognition (ConGD) dataset [31]. Our method was able to surpass continuous recognition performance of the state-of-the-art [3]. We also participated in the ChaLearn 2017 Continuous Gesture Recognition Challenge, ranking 3rd place [30].

The rest of the paper is structured as following: In Section 2 we describe each step of the proposed method in detail. In Section 3 we share our experimental setup and implementation details. Then we report our results on ConGD and compare the performance of our approach with the state-of-the art methods. Finally, we conclude our paper in Section 4 by discussing our findings and the future work.

2. Methodology

In this section we present a novel particle filter based probabilistic forced alignment technique that jointly learns to localize and recognize isolated instances in continuous streams using border level annotations. Given a video that contains multiple isolated occurrences, our method probabilistically divides the video into segments and classifies each segment using a 3D Convolutional Neural Network (3D-CNN).

While training we use border level annotations of continuous streams and build probability distributions of labels over frames for each isolated instance. To be able to segment isolated samples, or in other words to localize the silences between occurrences in a continuous stream, we introduce a silence class for which we also build probability distributions for each segment. We then train a 3D-CNN weighted by these distributions that learns to distinguish these classes. When the network is trained we use its output to update the probability distributions for the next iteration.

This process of updating distributions and retraining a 3D CNN is repeated until our network's recognition performance has converged. This iterative update process gives the proposed method the ability to localize the most representative regions in a weakly supervised manner. An overview of our method can be seen in Figure 3.

2.1. Building Initial Distributions

To create the initial approximate probability distributions we use the border level annotations and divide the training

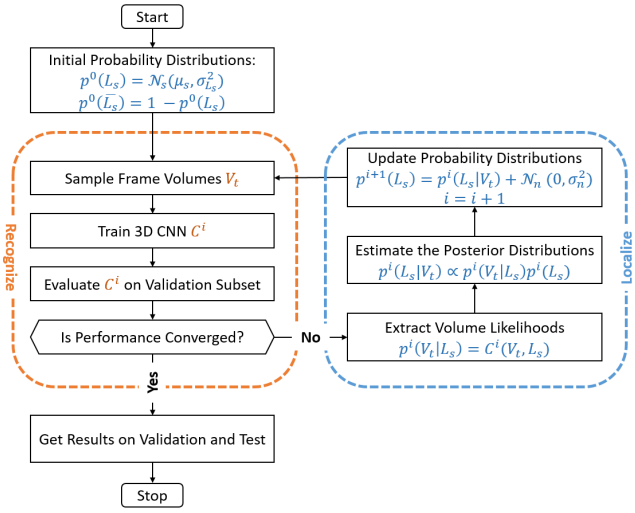

Figure 3. An overview of our iterative alignment procedure.

set into isolated instance segments. We calculate the standard deviation of segment lengths, σ_l, for each class l. We then propose an initial normal distribution, $p^0(L_s)$, for each segment s as:

$$p^0(L_s) = \mathcal{N}_s(\mu_s, (\sigma_l/2)^2) \tag{1}$$

where μ_s and L_s are the centre frame and the class label l of segment s, respectively. $p^0(L_s)$ is our prior assumption of a frame being discriminative for the present class, L_s, in a given segment s.

We also propose the inverse distribution, $p^0(\overline{L_s})$, for each segment as:

$$p^0(\overline{L_s}) = 1 - p^0(L_s) \tag{2}$$

which is the prior assumption of a frame being in the silence or transition region between instances of other classes. In other words being discriminative for the silence class. As an example, given a continuous stream with three isolated instances, our suggested prior distributions can be seen in Figure 4.

2.2. Training 3D Convolutional Neural Networks

Using our prior assumptions, $p^i(L_s)$ and $p^i(\overline{L_s})$, we now train our recognition system. We stochastically draw frames from each segment s according to these distributions. For each frame, F_t, we create a frame volume, V_t, by concatenating neighbouring frames where F_t is in the centre. Therefore, V_t represents the spatio-temporal changes around F_t. We then train a 3D Convolutional Neural Network (3D-CNN), C^i, using these volumes, which learns discriminative spatio-temporal features to distinguish between the target classes and the silence class.

We used the 3D CNN architecture proposed by Tran et al. [28], which has eight 3D Convolution, five 3D Pooling and

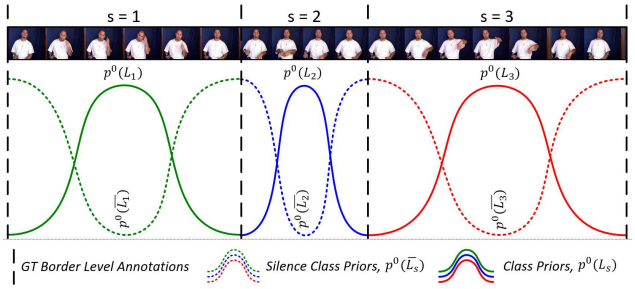

Figure 4. Initial Prior Distributions over an example video with three segments.

three Fully Connected layers. Our network architecture can be seen in Figure 5. We re-initialize the last fully connected layer $fc8$ which has $|l| + 1$ output units that corresponds to each class and the silence class.

During training we feed the stochastically selected volumes, V_t, from all the segments to the network and fine-tune the network, C^i, using Cross Entropy Loss [17] and the segment labels, L_s. We run the optimization for I iterations and take snapshots of the weights every K steps. We then select the best performing model from these snapshots by evaluating its continuous gesture recognition performance on a validation set, which we will be describing in detail in Section 2.4.

2.3. Particle Filter based Probabilistic Forced Alignment

We continue our alignment procedure by including new evidence from recognition. By feeding a frame volume, V_t, to the network, C^i, we estimate its likelihood of being discriminative for the class L_s of their respective segment s. We obtain the likelihood, $p^i(V_t|L_s)$, by taking the softmax of the corresponding output unit as:

$$p^i(V_t|L_s) = C^i(V_t, L_s) = \frac{e^{f^i(V_t, L_s)}}{\sum_l e^{f^i(V_t, l)}} \tag{3}$$

where function $f^i(V, l)$ is the value produced by the l^{th}output unit, which corresponds to class l, in the last fully connected layer, $fc8$, of the 3D-CNN ,C^i, for the input V.

Using the estimated likelihood distributions, $p^i(V_t|L_s)$, we update our prior assumptions $p^{i+1}(L_s)$ of each segment s. First, we calculate the posterior distribution $p^i(L_s|V_t)$ as:

$$p^i(L_s|V_t) \propto p^i(V_t|L_s)p^i(L_s) \tag{4}$$

$$\propto C^i(V_t, L_s)p^i(L_s) \tag{5}$$

where $p^i(V_t|L_s)$ and $p^i(L_s)$ are the estimated likelihood and our prior assumption for a segment s, respectively. A visualization of our posterior estimation can be seen in Figure 1.

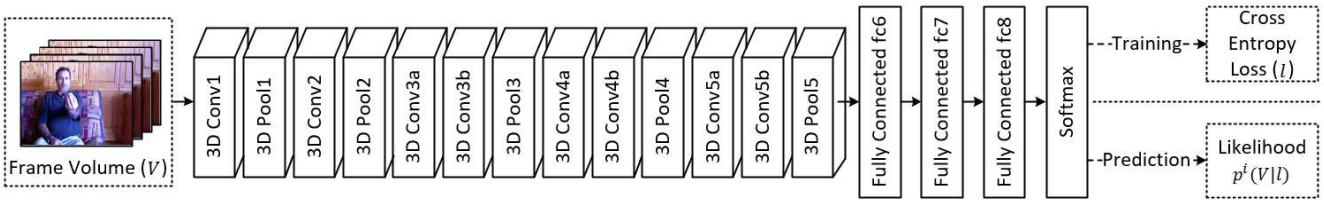

Figure 5. Our 3D Convolutional Neural Network Architecture

Then we update our prior assumption for the next iteration as:

$$p^{i+1}(L_s) = p^i(L_s|V_t) + \mathcal{N}_n(0, \sigma_n^2) \quad (6)$$

where N_n is Gaussian noise with σ_n standard deviation. Using the updated prior distributions, $p^{i+1}(L_s)$, we retrain a new 3D-CNN, C^{i+1}, initializing from the previous weights, C^i:

$$C^{i+1} = \text{ReTrain}(C^i, p^{i+1}(L_s)) \quad (7)$$

This alignment (Section 2.3) and retraining a 3D-CNN (Section 2.2) procedure is repeated until the network's continuous recognition performance (Section 2.4) converges.

2.4. Prediction: Segmentation and Classification

We propose a two stage prediction approach for recognizing isolated instances in continuous streams. Our method first splits a given video into multiple segments (which do not necessarily correspond to the ground truth segmentations) and then classifies each segment using the likelihoods extracted from a 3D CNN. This re-segmentation allows our method to handle unlabelled instances.

Given a video of T frames we initially create frame volumes, V_t, for each frame, F_t, by concatenating neighbouring frames. We then extract likelihoods of these volumes $p(V_{1:T}|l)$ for each class, l, using a 3D CNN, C^i. l can be either one of the classes or the silence class. Using the extracted likelihoods we probabilistically split the video into multiple isolated segments. This is done by localizing silence regions using the silence class likelihood $p^i(V_{1:T}|\overline{L_s})$ over frames. Assuming there are silence or transitions between instances, we segment the video clip where the silence class likelihood in a region is higher than any other class.

Once the video is divided into segments, we use the extracted class likelihoods to predict the most prominent class in each non-silence segment,L_s^{pred}, as:

$$L_s^{pred} = \underset{l}{\text{argmax}} \sum_{t}^{T_s} p^i(V_t|l) \quad (8)$$

where T_s is the number of frames in a given segment s and $p^i(V_t|l)$ is the likelihood of frame volumes V_t belonging to the class l. We then expand the segment level predictions into border level annotations and evaluate it against the ground truth using the Mean Jaccard Index Score. A visualization of our recognition procedure can be seen in Figure 6.

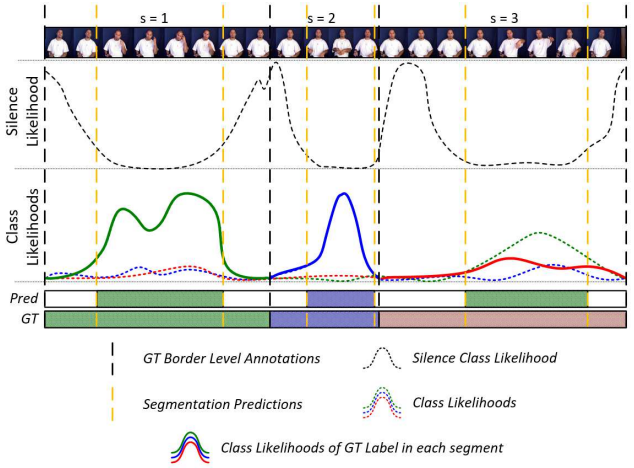

Figure 6. Recognition: Segmentation and Classification (GT = Ground Truth Labels, Pred = Our Predictions). In this example, ,we were able to successfully localize and recognize first two segments (s = 1 and s = 2) while failing to recognize the third segment (s = 3).

3. Evaluation

To evaluate the effectiveness of the particle filter based probabilistic forced alignment approach we conducted experiments on the popular ChaLearn Continuous Gesture Dataset (ConGD) [31], featured in the 2nd round of the ChaLearn 2017 Continuous Gesture Recognition Challenge. The dataset was formed by re-annotating the ChaLearn 2011 Gesture Dataset [20] to enable evaluation of user-independent recognition. ConGD contains 47,933 gesture samples belonging to 249 gesture classes performed by 21 subjects. A summary of the dataset can be seen in Table 1. It is currently the largest continuous user-independent gesture recognition dataset surpassing other large gesture recognition datasets, such as ChaLearn 2014 [14] (which has 13,858 samples of 20 gesture classes), both in the number of samples and the number of classes.

We implemented our method using the Caffe [22] distribution developed by Tran et al. [28] that supports the use of 3D Convolution and Pooling layers. The code to reproduce our results is publicly available[1].

[1]https://github.com/neccam/PaFiFA

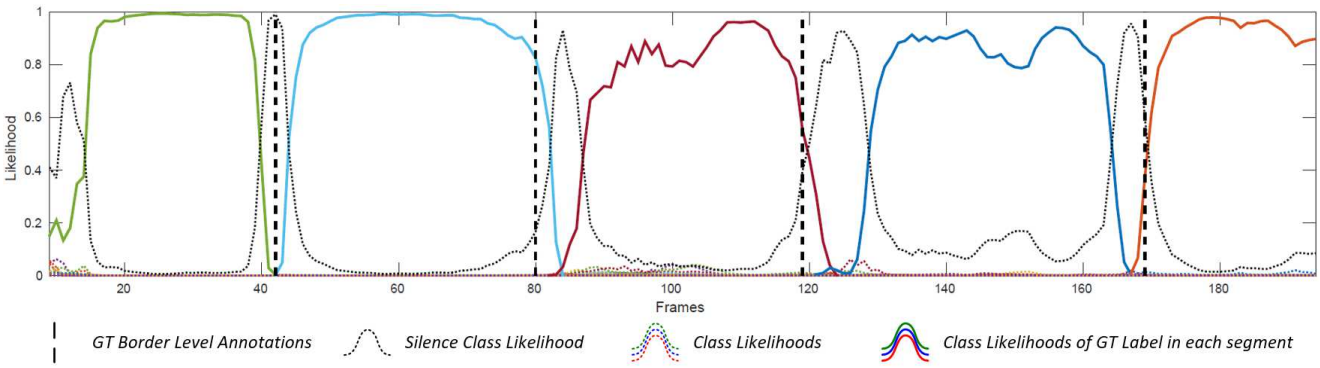

Figure 7. Recognition results on a validation sample (ConGD Validation Set Sample ID: 001/00012)

Partition Name	# of Samples	# of Sequences	# of Subjects
Train	30,442	14,134	17
Validation	8,889	4,179	2
Test	8,602	4,042	2
All	47,933	22,355	21

Table 1. Summary of ChaLearn 2016 ConGD Dataset

To train our 3D-CNNs we have used Stochastic Gradient Descent [2] with a learning rate of $lr = 10^{-3}$, a momentum of $m = 0.99$ and a batch size of 45. The networks were initialized with the pre-trained weights provided by Camgoz et al. [3] which was the previous state-of-the-art on the ConGD dataset. We re-initialized the last fully connected layer of our 3D-CNNs for each alignment iteration using the Xavier initialization method [16].

We trained our networks for 40,000 steps (roughly 3 epochs) for each alignment iteration. We take snapshots of 3D-CNN weights every 1,000 steps. We then evaluate the continuous gesture recognition performance as described in Section 2.4 to determine the best performing model. Due to large number of models (40 for each alignment iteration), we used a subset of the ConGD validation set to speed up the evaluation process. The same validation subset has been used for all alignment iterations. The best performing model is then used to perform continuous recognition on the validation and test sets of ConGD.

To approximate the prior distributions we stochastically chose twenty frames from each segment, ten for the class present in the segment and ten for the silence class. For each frame we create a frame volume by concatenating the neighbouring 16 RGB frames. After each iteration we keep half of the samples which have the highest posteriors and re-sample the remainder using stationary sampling. We used five frames as σ_n.

We iterated our method 3 times and evaluated its performance on the validation set. As can be seen in Table 2 our method's recognition performance generally improves over iterations. We believe the fluctuations over the iterations is related to the number of samples we have used to approximate the importance distributions. Using more samples would yield a lower variation and more smooth change over the iterations, while slowing the training and alignment steps of our method. Even though we selected the best performing model for each iteration by evaluating them on a validation subset, the change in the performance was also reflected on the full validation set, indicating good generalization. After three iterations, the recognition performance has not completely converged and further iterations may improve the performance.

Iteration	Validation Subset MJI	Validation MJI
0	0.3743	0.3544
1	0.3996	0.3646
2	0.3998	0.3634
3	0.4048	0.3806

Table 2. Performance evaluation of the proposed method over iterations. (MJI: Mean Jaccard Index)

The recognition performance of a ConGD validation sample using our best performing model can be seen in Figure 7. Given a sample with 5 isolated instances, our method was able to correctly recognize all of them. Furthermore, it managed to localize where the gestures are more accurately than the "ground truth" annotations. In addition, there was low inter-class ambiguity in gesture segments, indicating that the introduction of the silence class did not cause class confusion.

When compared with the previous state-of-the-art [13], our method with a single alignment step was able to surpass performance on both the validation and the test set by 6.3284% and 18.97044% relative Mean Jaccard Index score respectively (See Table 3). With two additional alignment steps our method gained 4% more relative Mean Jaccard Index score over the state-of-the-art on the validation set. However, we could not report our results on the test set as the labels are withheld. Furthermore, our method yielded more balanced performance between validation and test set, indicating better generalization. We also submitted our recognition results from the first iteration to the 2nd round of the ChaLearn 2017 Continuous Gesture Recognition Challenge

and were ranked third. As our method is learner independent, it could easily be integrated to other approaches from the challenge.

	Method	Validation MJI	Test MJI
Baseline [31]	MFSK	0.0902	0.1464
Wang et al. [32]	IDMM + CNN	0.2403	0.2655
Chai et al. [6]	CNN + RNN	0.2655	0.2869
Camgoz et al. [3]	3D-CNN	0.3429	0.3147
Ours (Challenge)	3D-CNN + Alignment	**0.3646**	**0.3744**
Ours (Best)	3D-CNN + Alignment	**0.3806**	N/A

Table 3. Comparison with the state-of-the-art methods (MJI: Mean Jaccard Index)

4. Conclusion

In this paper we proposed a particle filter based probabilistic forced alignment approach to address the weakly supervised training of deep spatio-temporal neural networks. Using the proposed method we trained 3D-CNNs to simultaneously localize and recognize isolated instances in continuous streams.

We applied our approach to the difficult task of continuous gesture recognition and evaluated its performance on the challenging ChaLearn 2016 Continuous Gesture Recognition dataset. Through our experiments, we have seen the effectiveness of the proposed probabilistic forced alignment approach as it has iteratively improved the recognition performance by 8.1485% relative Mean Jaccard Index Score compared to training with a naive prior distribution. Our method was able the surpass the previous state-of-the-art [3], which also used a 3D-CNN based learning, by obtaining 0.3806 Mean Jaccard Index Score on the validation set. We also participated in the ChaLearn 2017 Continuous Gesture Recognition challenge and were ranked third [30]. However, our method is independent of the learning algorithm and could easily be used with other approaches.

As future work we would like to apply our method to other spatio-temporal tasks such as Continuous Sign Language Recognition and extend our framework to incorporate multiple modalities. It would also be interesting to investigate parallel alignment of these modalities.

References

[1] D. Bahdanau, K. Cho, and Y. Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. In *International Conference on Learning Representations (ICLR)*, 2015.

[2] L. Bottou. Large-Scale Machine Learning with Stochastic Gradient Descent. In *International Conference on Computational Statistics (COMPSTAT)*. 2010.

[3] N. C. Camgoz, S. Hadfield, O. Koller, and R. Bowden. Using Convolutional 3D Neural Networks for User-Independent Continuous Gesture Recognition. In *International Conference on Pattern Recognition (ICPR) Workshops*, 2016.

[4] N. C. Camgöz, S. Hadfield, O. Koller, and R. Bowden. Subunets: End-to-end hand shape and continuous sign language recognition. *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[5] J. Carreira and A. Zisserman. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. *arXiv:1705.07750*, 2017.

[6] X. Chai, Z. Liu, F. Yin, Z. Liu, and X. Chen. Two Streams Recurrent Neural Networks for Large-Scale Continuous Gesture Recognition. In *International Conference on Pattern Recognition (ICPR) Workshops*, 2016.

[7] K. Cho, A. Courville, and Y. Bengio. Describing MultiMedia Content using Attention-based Encoder-Decoder Networks. *IEEE Transactions on Multimedia*, 17(11):1875–1886, 2015.

[8] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

[9] J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman. Lip Reading Sentences in the Wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[10] J. Duan, J. Wan, S. Zhou, X. Guo, and S. Li. A Unified Framework for Multi-Modal Isolated Gesture Recognition. In *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM),(under review, round 2)*, 2017.

[11] J. Duan, S. Zhou, J. Wan, X. Guo, and S. Z. Li. Multi-Modality Fusion based on Consensus-Voting and 3D Convolution for Isolated Gesture Recognition. *arXiv:1611.06689*, 2016.

[12] S. Dupont and J. Luettin. Audio-Visual Speech Modeling for Continuous Speech Recognition. *IEEE Transactions on Multimedia*, 2(3):141–151, 2000.

[13] H. J. Escalante, V. Ponce-López, J. Wan, M. A. Riegler, B. Chen, A. Clapés, S. Escalera, I. Guyon, X. Baró, P. Halvorsen, et al. Chalearn Joint Contest on MultiMedia Challenges Beyond Visual Analysis: An Overview. In *IEEE International Conference on Pattern Recognition (ICPR) Workshops*, 2016.

[14] S. Escalera, X. Baró, J. Gonzalez, M. Á. Bautista, M. Madadi, M. Reyes, V. Ponce-López, H. J. Escalante, J. Shotton, and I. Guyon. ChaLearn Looking at People Challenge 2014: Dataset and Results. In *ECCV Workshops (1)*, pages 459–473, 2014.

[15] J. Forster, C. Schmidt, O. Koller, M. Bellgardt, and H. Ney. Extensions of the Sign Language Recognition and Translation Corpus RWTH-PHOENIX-Weather. In *International Conference on Language Resources and Evaluation (LREC)*, 2014.

[16] X. Glorot and Y. Bengio. Understanding the Difficulty of Training Deep Feedforward Neural Networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 249–256, 2010.

[17] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT press, 2016.

[18] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber. Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks. In *International Conference on Machine Learning (ICML)*, 2006.

[19] A. Graves, A. Mohamed, and G. Hinton. Speech Recognition with Deep Recurrent Neural Networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013.

[20] I. Guyon, V. Athitsos, P. Jangyodsuk, and H. J. Escalante. The ChaLearn Gesture Dataset (CGD 2011). *Machine Vision and Applications*, 25(8):1929–1951, 2014.

[21] S. Ji, W. Xu, M. Yang, and K. Yu. 3D Convolutional Neural Networks for Human Action Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 35(1):221–231, 2013.

[22] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional Architecture for Fast Feature Embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM, 2014.

[23] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, et al. The Kinetics Human Action Video Dataset. *arXiv:1705.06950*, 2017.

[24] 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 (CVPR)*, 2016.

[25] O. Koller, S. Zargaran, and H. Ney. Re-Sign: Re-Aligned End-to-End Sequence Modelling with Deep Recurrent CNN-HMMs. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[26] Y. LeCun, Y. Bengio, and G. Hinton. Deep Learning. *Nature*, 521(7553):436–444, 2015.

[27] K. Simonyan and A. Zisserman. Two-Stream Convolutional Networks for Action Recognition in Videos. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.

[28] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning Spatiotemporal Features with 3D Convolutional Networks. In *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[29] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko. Sequence to Sequence-Video to Text. In *Proceedings of the IEEE international Conference on Computer Vision (ICCV)*, 2015.

[30] J. Wan, S. Escalera, A. Gholamreza, H. J. Escalante, X. Baró, I. Guyon, M. Madadi, A. Juri, G. Jelena, L. Chi, and X. Yiliang. Results and Analysis of ChaLearn LAP Multi-modal Isolated and Continuous Gesture Recognition, and Real versus Fake Expressed Emotions Challenges. In *IEEE International Conference on Computer Vision (ICCV) Workshops*, 2017.

[31] J. Wan, Y. Zhao, S. Zhou, I. Guyon, S. Escalera, and S. Z. Li. Chalearn Looking at People: RGB-D Isolated and Continuous Datasets for Gesture Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2016.

[32] P. Wang, W. Li, S. Liu, Y. Zhang, Z. Gao, and P. Ogunbona. Large-scale Continuous Gesture Recognition Using Convolutional Neutral Networks. *International Conference on Pattern Recognition (ICPR) Workshops*, 2016.

[33] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding Deep Learning Requires Rethinking Generalization. 2017.