Recognizing American Sign Language Gestures from within Continuous Videos

Yuancheng Ye[1], Yingli Tian[1,2,*], Matt Huenerfauth[3], and Jingya Liu[2]
[1]The Graduate Center, City University of New York, NY, USA
[2]The City College, City University of New York, NY, USA
[3]Rochester Institute of Technology, Rochester, NY, USA
[*]Corresponding author: ytian@ccny.cuny.edu

Abstract

In this paper, we propose a novel hybrid model, 3D recurrent convolutional neural networks (3DRCNN), to recognize American Sign Language (ASL) gestures and localize their temporal boundaries within continuous videos, by fusing multi-modality features. Our proposed 3DRCNN model integrates 3D convolutional neural network (3DCNN) and enhanced fully connected recurrent neural network (FC-RNN), where 3DCNN learns multi-modality features from RGB, motion, and depth channels, and FC-RNN captures the temporal information among short video clips divided from the original video. Consecutive clips with the same semantic meaning are singled out by applying the sliding window approach to segment the clips on the entire video sequence. To evaluate our method, we collected a new ASL dataset which contains two types of videos: Sequence videos (in which a human performs a list of specific ASL words) and Sentence videos (in which a human performs ASL sentences, containing multiple ASL words). The dataset is fully annotated for each semantic region (i.e. the time duration of each word that the human signer performs) and contains multiple input channels. Our proposed method achieves 69.2% accuracy on the Sequence videos for 27 ASL words, which demonstrates its effectiveness of detecting ASL gestures from continuous videos.

1. Introduction

For many video analysis tasks such as action recognition and video captioning, the target video is associated with attributes that correspond to its entire duration. However, for other applications, it is important to know the characteristics of specific time intervals of a video. For instance, identifying a clip that contains a specific meaningful human action, from within a longer continuous video, is a common task in American Sign Language (ASL) recognition research. At times, researchers may wish to search videos for a performance of a specific ASL word, or for other applications, researchers may wish to segment and label all of the words appearing in a video. Finding and analyzing a specific movement in a long video requires a good understanding of each meaningful segment of content in the whole video, and existing methods may fail on this task.

Convolutional neural networks (CNNs) have enjoyed great success on many image-based computer vision tasks and have been extended to video domain. C3D [28] is a recently proposed model which constructs a 3D convolutional neural network to learn the spatiotemporal features from sliced video clips and then fuses those features to make the final classification. In addition to CNNs, recurrent neural networks (RNNs) have also been applied to many computer vision tasks [22].

In this paper, we propose a hybrid model, which applies C3D models to extract spatiotemporal features of sliced clips from different input channels and then adopts FC-RNN [31] to learn the sequential information among these clips. Our framework is similar to [2], which is the pioneering work to integrate CNNs with RNNs. However, their work only takes into account the sequential information of the frames. Our framework, which combines extracted spatiotemporal features with RNNs, can model both the spatiotemporal information of the sliced clips and the sequential characteristics of these clips.

As background, there are more than one hundred sign languages used around the world, and American Sign Language (ASL) is used throughout the U.S. and Canada, as well as other regions of the world, including West Africa and Southeast Asia. Within the U.S., there are approximately 500,000 people who use ASL as a primary language [21], and since there are significant linguistic differences between English and ASL, it is possible to fluent in one language but not the other. Due to a variety of educational factors and childhood language exposure, researchers have measured lower levels of English literacy among many deaf adults in the U.S. [29]. Technology to automatically recognize ASL signs from video input could enable new communication and accessibility technologies for people who

are deaf or hard-of-hearing (DHH), which may allow these users to input information into computing systems by performing sign language or may serve as a foundation for future research on machine translation technologies for sign languages. In addition, there has been recent work examining how technology to automatically identify words in videos of ASL could be used to create educational technologies for students who are learning sign language [9].

The main contributions of our paper are three-fold:

- We propose a novel hybrid model, 3DRCNN, by integrating C3D with RNNs to capture both spatiotemporal and sequential information of the sliced clips, which improves the discriminative power of the final feature representations. The proposed 3DRCNN is able to recognize and localize different semantic components in continuous videos. These semantic components may have various lengths.
- We are the first to design the multi-channel end-to-end network structure to detect and recognize the American Sign Language words from continuous videos.
- We create a new ASL dataset including multiple modalities (facial movements, hand gestures, and body pose) and multiple channels (RGB, Optical flow, and depth) by collaborating with ASL linguistic researchers; this dataset contains full annotation of the time durations when specific meaningful human actions occur, i.e. the time intervals that correspond to each word that the human is performing in ASL in a video. As to our knowledge, this is the first dataset to provide multiple modalities in ASL research.

2. Related Work

Recognizing actions and detecting their semantic temporal locations from within continuous videos is a challenging problem. One of the main challenges is that in actions in continuous videos, the temporal boundaries of a specific movement are not very clear. Another difficulty is that there exist quite few annotated datasets for this task. In this paper, we attempt to solve this new problem, i.e. recognizing ASL gestures and detecting their temporal locations from within continuous videos, by proposing a novel method which is based on recent achievements on deep learning research and by collecting an ASL dataset which has been annotated with the time-intervals for each ASL word. Our proposed method contains two primary components: 1) recognizing the contents of a sequence of frames; 2) locating the temporal boundaries of the targeted movement.

Since the work [16] which makes use of the powerful computation ability of GPUs, deep neural networks (DNNs) have enjoyed a renaissance in various areas of computer vision, such as image classification [3, 27], object detection [7, 8], image description [2, 11], and others. Many endeavors have been made to extend CNNs to the video domain [4]. However, this task is difficult mainly due to three major problems: 1) Video data is much larger in size than images; therefore, efficiently handling the video data in the limited GPU memory is not tractable. 2) Compared to two-dimensional image data, videos have an additional temporal dimension. While it is important to take into account this temporal information, most CNN techniques are only image-based. 3) Videos may contain different numbers of frames; unlike images, for which a simple resizing process can transform them into a uniform size, video interpolation or extrapolation process may result in loss of important temporal information. An intuitive way to extend image-based CNN structures to the video domain is to perform the finetuning and classification process on each frame independently, and then conduct a later fusion, such as average scoring, to predict the action class of the video. Despite of its simple implementation, this method achieves comparable results to many carefully designed algorithms. To incorporate temporal information in the video, [26] introduced a two-stream framework. One stream is based on RGB images, and the other is based on stacked optical flows. Although that work proposed an innovative way to learn temporal information using a CNN structure, in essence it is still image-based since the third dimension of stacked optical flows collapsed immediately after the first convolutional layer. 3D CNN structure [10] can learn temporal information from the 3D convolutions with both spatial and temporal stride. The C3D model only operates on the sliced clips with fixed length, and later fusion methods are performed to obtain the final category of the entire video. This compromise implementation trick may obviate the sequential dependencies of those isolated clips.

To model the sequential information of different extracted features, [32] and [2] proposed to input features into RNN structures, and they achieved good results for an action recognition task. The former emphasizes pooling strategies and how to fuse different features, while the latter focused on how to train an end-to-end DNN structure that integrates CNNs with RNNs. The function of RNNs can be viewed as embedding sequential information into the input sequence. Therefore, the intention of both [32] and [2] was to explore the sequential information of the independent frames. This approach neglected the spatiotemporal attributes within the video, and to offset this shortcoming, both studies applied optical flow information as an additional channel and performed a later fusion with the RGB channel. As demonstrated in both studies, by fusing the optical flow information, the accuracy of action recognition can be significantly improved. This demonstrates the discriminative power of feature representations of videos can be improved by combining sequential and spatiotemporal information. Inspired by this, we propose a C3D-RNN hy-

brid framework to recognize the contents of sliced clips.

To the best of our knowledge, there are only a few methods that consider the multiple channels (RGB and Depth) for sign language recognition. The work of Koller and his colleagues [1, 14, 15] only employ RGB channel and only perform the recognition task on sign language videos. Although some other literatures attempted to employ both RGB and Depth channels [25, 17, 24], they are not deep learning based and the features of RGB and Depth channels are extracted independently first and then fused later. Moreover, those methods focus on the recognition task from videos only contain individual sign language words. Instead, our proposed approach attempts to recognize specific sign language words as well as detect their temporal locations in continuous sentences.

As discussed above, technology to recognize ASL signs from videos could enable new assistive technologies for people who are DHH, and there has been significant research on sign language of recognition, e.g. [19, 6]. However, a limiting factor for much of this research has been the scarcity of video recordings of sign language that have been annotated with time interval labels of the words that the human has performed in the video: For ASL, there have been some annotated video-based datasets [23] or collections of motion capture recordings of humans wearing special sensors [20]. While most existing datasets [5, 13] only contain general ASL vocabularies from RGB videos and a few with RGBD channels, our dataset is the first to focus on ASL grammar and fluent as well as to provide RGB, depth, skeleton, and HDface information. In this paper, we make use of a new dataset that we have collected in collaboration with ASL computational linguistic researchers (details in section 5), using a Kinect depth sensor to collect RGB and depth information.

3. Proposed 3DRCNN Hybrid Model

3.1. Spatiotemporal Feature Extraction by C3D

The C3D slices the whole video into fixed length clips, and then conducts the finetuning and classification processes on these clips independently.

In the 2D CNNs, the dimension of each feature map is $n \times h \times w$, where n stands for the number of filters in the corresponding convolutional layer, h and w represents the height and width of the feature map. The spatial size of the filter in the 2D convolutional process is defined manually, while the third dimension is automatically set to the value of the first dimension of the feature maps in the previous layer, which is also the number of filters in that layer.

In the 3D CNNs, the dimension of the feature maps produced by each convolutional layer is $n \times l \times h \times w$, where the additional parameter l stands for the number of frames. For the 3D filters in the 3D convolutional process, in addition to the spatial size, temporal length should also be set manually. The fourth dimension of the 3D filter is also automatically set to the first dimension of the feature maps in the previous convolutional layer. Moreover, instead of only pooling on the spatial domain, the pooling processes in the 3D CNNs pool the features in a small cuboid. Please note that if the temporal lengths of all 3D pooling layers are set to 2, then the number of 3D pooling layers in the 3D CNN structure should be $log_2 n$, where n is the length of sliced clips. After all the convolutional processes, a feature vector is generated and then fed into the fully connected layers. The network structure in C3D has 8 convolutional layers, 5 max-pooling layers, and 2 fully connected layers. The sizes of all 3D convolutional kernels are $3 \times 3 \times 3$, and the stride of these kernels are all 1 in both spatial and temporal domain. The sizes of all pooling kernels are $2 \times 2 \times 2$, except for the first one, which is $1 \times 2 \times 2$.

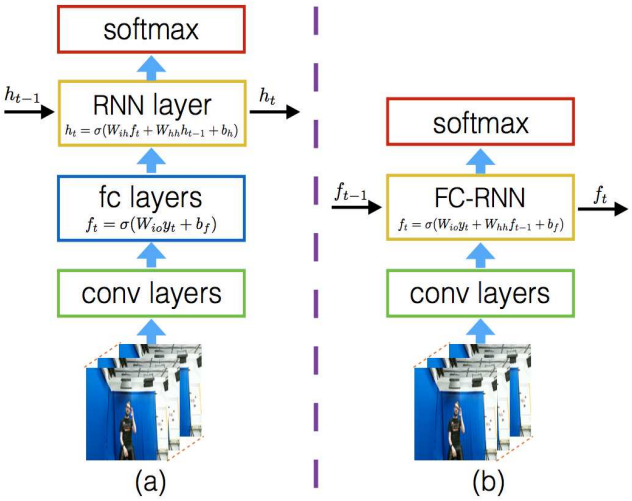

Figure 1: Comparison of (a) standard RNN and (b) our proposed FC-RNN.

Due to the limitation of GPU computing resources, the C3D model only operates on video clips of fixed length. Although by conducting later fusion on these clips, comparable results can be achieved, important sequential information between sliced clips is neglected.

3.2. Temporal Information Extraction by FC-RNN

Recurrent neural networks have succeeded in speech recognition, machine translation, computer vision, and other tasks; a notable advantage is their handling inputs with variant lengths [30].

FC-RNNs capture the temporal progress through a hidden state $\boldsymbol{h}_t$ at timestamp t whose activation is dependent on that of the previous timestamp:

$$\boldsymbol{h}_t = \sigma(\boldsymbol{W}_{ih}\boldsymbol{y}_t + \boldsymbol{W}_{hh}\boldsymbol{h}_{t-1} + b_h), \quad (1)$$

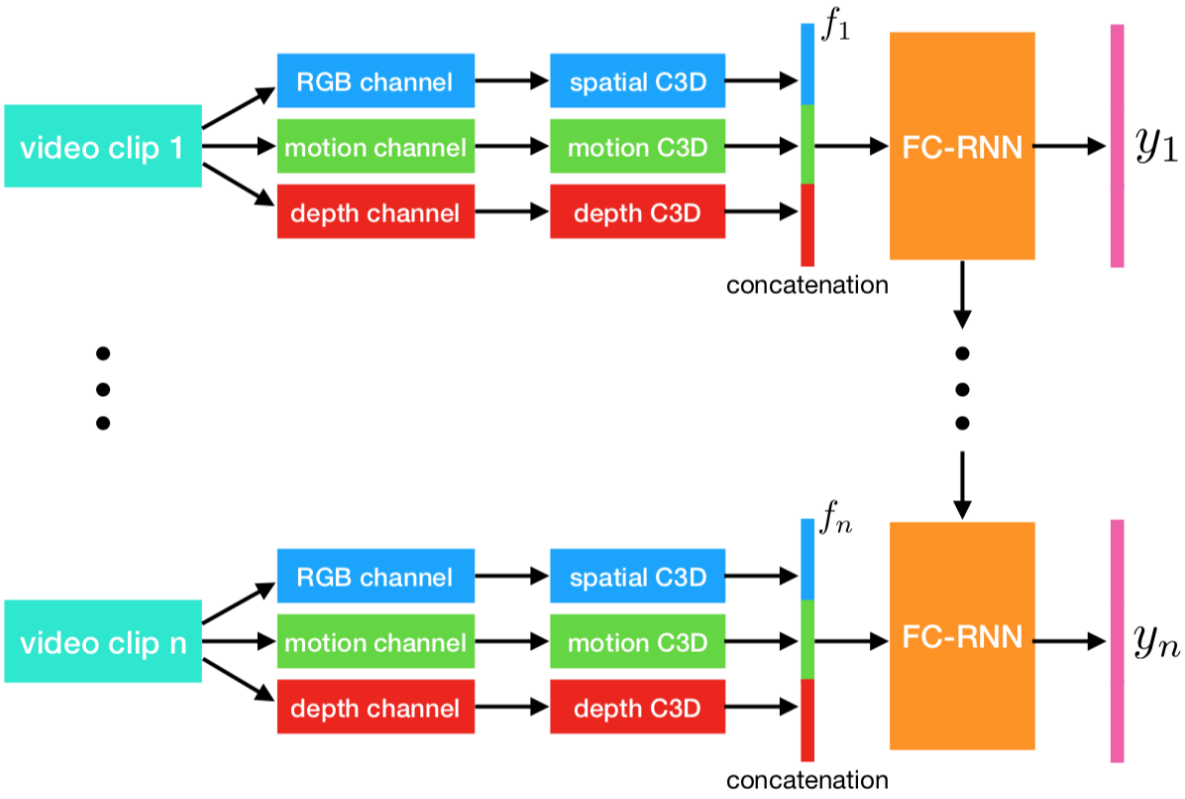

Figure 2: Illustration of the proposed 3DRCNN framework.

where σ is the activation function, $\boldsymbol{W}_{ih}$ is the input-to-hidden weight matrix, $\boldsymbol{W}_{hh}$ is the hidden-to-hidden weight matrix, $\boldsymbol{y}_t$ is the input feature, $\boldsymbol{h}_{t-1}$ is the hidden state from the previous timestamp, and b_h is the bias. RNN in most vision tasks is built upon CNNs that are pre-trained on large-scale datasets. However in traditional RNN, both $\boldsymbol{W}_{ih}$ and $\boldsymbol{W}_{hh}$ are randomly initialized. So, it would require the training of such a recurrent layer from scratch even if the pre-trained 3DCNN is used for feature extraction. We adopt the FC-RNN [31] which fuses the recurrent layer with the fully connected layer of a pre-trained CNN to preserve the pre-trained architecture as much as possible.

Assume the output of a fully connected layer of the 3DCNN at timestamp t is:

$$\boldsymbol{y}_t = \sigma(\boldsymbol{W}_{io}\boldsymbol{x}_t + b_y), \tag{2}$$

where $\boldsymbol{W}_{io}$ is the pre-trained input-to-output weight matrix, $\boldsymbol{x}_t$ is the output of previous feed-forward layer, and b_y is the bias. FC-RNN transforms it into a recurrent layer through:

$$\boldsymbol{y}_t = \sigma(\boldsymbol{W}_{io}\boldsymbol{x}_t + \boldsymbol{W}_{hh}\boldsymbol{y}_{t-1} + b_y). \tag{3}$$

This structure only introduces a single hidden-to-hidden weight matrix $\boldsymbol{W}_{hh}$ that is required to be trained from scratch, while other weight matrices have already been pre-trained and can be just fine-tuned. Figure 1 presents the comparison between the structures of RNN and FC-RNN. In the experiments, we observe that FC-RNN is able to effectively reduce over-fitting and expedite convergence.

3.3. Proposed 3DRCNN Hybrid Model

Based on the above discussions, we propose the 3DRCNN model to capture both spatiotemporal and sequential information of sliced video clips. The illustration of the 3DRCNN framework is demonstrated in the Figure 2.

In our proposed framework, the video is sliced into fixed length clips, from which spatiotemporal features are extracted. Afterward, these features are fed into the RNN model to embed the sequential information to the final representation, and then the labels of sliced clips are predicted independently. To stitch those discrete clips with the same labels in the video, we propose a linking method which greedily integrates the clips with highest confidence scores.

Suppose that the video $\mathbb{V}$ is sliced with a certain overlap into clips $\langle c_1, c_2, \ldots, c_n \rangle$. These sliced video clips are represented by the spatiotemporal features extracted from them:

$$\langle c_1, c_2, \ldots, c_n \rangle \mapsto \langle f_1, f_2, \ldots, f_n \rangle, \tag{4}$$

and then the sequence of these features, in the same order as the video clips, is fed into the sequential information encoder:

$$\langle y_1, y_2, \ldots, y_n \rangle = RNN(\langle f_1, f_2, \ldots, f_n \rangle), \quad (5)$$

In addition to training a C3D network on the RGB frames of videos, we also train a motion C3D network on the optical flow and also a depth C3D network on depth images. The optical flow images are generated by stacking the x-component, the y-component, and the magnitude of the flow on three channels respectively. Each element in the image is then multiplied by 16 and converted to the closest integer between 0 and 255. This practice has yielded good performance in many other studies [2, 32]. To apply the pre-trained CNN model to one-channel depth images, we sum the convolutional kernels for the three channels of the first layer to obtain one kernel. As observed in the experimental results, by fusing all the features generated by RGB, motion, and depth C3D models, the performance can be improved, which indicates that complementary information can be provided by training deep neural networks on different channels.

The features extracted by RGB, motion, and depth C3D networks are concatenated to form the feature representations of the clips. Before feeding these feature representations into the FC-RNN model, a fully connected layer is also added to smooth the transition process.

To stitch the discrete clips into the continuous region, we propose a linking method, which greedily groups the clips with highest scores. Suppose that, for a sequence of clips $\langle c_1, c_2, \ldots, c_n \rangle$, the clip with highest confidence score for a specific target is c_i. If the clip with next highest score is c_j, then we calculate the average score of the temporal region $[c_i, c_j]$. If this average score is larger than a threshold, then we group all the clips between c_i and c_j, otherwise c_j is discarded and deleted from the pool of clips. If the clips between c_i and c_j are grouped, the average score of these clips is then considered in the following steps. This procedure is continued until all clips classified with the same label have been processed.

4. Dataset

A new dataset has been collected for this research in collaboration with ASL computational linguistic researchers. To facilitate this collection process, we have designed some recording software based on the Kinect 2.0 sensor. For recording, we asked the participants to perform two types of videos: (1) Sequence videos: the human performing a sequence of 99 ASL signs, and (2) Sentence videos: the human performing continuous signing consisting of short (1-5 sentences) responses to some prompt (details below). During the recording session, a native ASL signer met the participant and conducted the session: Prior research in ASL computational linguistics has emphasized the importance of having only native signers present when recording ASL videos so that the signer does not produce English-influenced signing [20]. Details about the videos appear below:

Figure 3: Two examples of videos from the Sequence and Sentence video datasets. Faces are blurred to preserve participant privacy. (a) A screenshot taken from a Sequence video collected for this study, showing a fluent signer performing the ASL word "ME," in which she makes a pointing movement to her torso. (b) A screenshot taken from a Sentence video collected for this study, showing an ASL student performing the ASL word "LIKE," in which the hand moves away from the torso while moving the thumb and third finger together until they touch.

Sequence videos: A total of 27 videos were collected from 15 people, who are fluent ASL signers. Participants responded affirmatively to the following screening question: Did you use ASL at home growing up, or did you attend a school as a very young child where you used ASL? Participants were provided with a slideshow presentation that asked them to perform a sequence of 99 individual ASL signs, without lowering their hands between words. The 99 signs were selected based on their use in the curriculum of an introductory ASL course. (This dataset was collected as part of a larger research project to investigate educational tools for students in such courses.)

Sentence videos: A total of 100 videos were collected from 14 people, which included two groups: students and fluent signs. The students consisted of 6 individuals who were currently enrolled (or had just completed) their second semester of a university-level course in ASL. The fluent signers consisted of 8 individuals who responded affirmatively to the screening questions (above) that asked about their early use of ASL. For these videos, the participant was presented with a mock homework assignment for an ASL course, which asked them to create short videos (between 1 and 5 sentences in length) in response to prompts (e.g. Can you tell us about any pets you have owned? Can you tell us

about where you live?).

All the video sequences were annotated by a team of ASL linguists, who produced a timeline of the words in the video. The linguists used a coding scheme in which an English identifier label was used to correspond to each of the ASL words used in the videos, in a consistent manner across the videos. For example, all of the time spans in the videos when the human performed the ASL word "NOT" were labeled with the English string "NOT" in our linguistic annotation. Total 27 ASL words are recognized in our experiments including "YOU", "NOT", "NOW", "WHERE", "WHICH", "NONE", "NO", "QMWG", "TIME", "TODAY", "I-ME","EVERY_DAY", "TONIGHT", "QUESTION", "WEEK" "ALWAYS", "TUESDAY", "LAST_YEAR", "DON'T_CARE", "DODO1", "WEDNESDAY", "DODO2", "IX_HE_SHE_IT", "WAVE_NO", "EVERY_THURSDAYY", "SOON2", and "MONTH". The ratio of training and testing instances for each ASL word is $7:3$. The lists of training and testing files used in our experiment will be released at the conclusion of our project, along with the associated dataset. Two example frames from the dataset are demonstrated in the Figure 3.

5. Experiments

5.1. Implementation Details

The C3D network is pre-trained on the Sports1M dataset [12]. The training parameters are the same as in the paper [28]. Three C3D networks are trained in our proposed framework: spatial-C3D on RGB frames, motion-C3D on optical flow images and depth-C3D on depth images. The dimensions of the features extracted by spatial-C3D, motion-C3D and depth-C3D are all 4096. The final spatiotemporal features of sliced video clips are the concatenations of these features. Therefore, the final dimension of the extracted spatiotemporal features is $3 \times 4096 = 12288$. To integrate the RNN model with the extracted spatiotemporal features, a fully connected layer is added ahead of the RNN network.

Our proposed model is trained in Theano with cuDNN3 on two Titan X GPUs. We fine-tune the C3D models for 16 epochs with an initial learning rate of $\lambda = 3\times10^{-3}$, reduced by a factor of 10 after every 4 epochs. To increase variability in the training examples, the following data augmentation steps are applied to each video in addition to cropping: random spatial rotation and scaling, temporal scaling, and jittering. The parameters for each augmentation step are drawn from a uniform distribution with a specified range.

Since recurrent connections can learn the specific order of actions in the training set, the training videos are randomly permuted for each training epoch. To apply the pre-trained CNNs on three-channel RGB images to one-channel depth images, we sum the convolutional kernels for the three channels of the first layer to obtain one kernel. For the 3D-CNN, the video is split into non-overlapping clips of 16 frames. We analyze our proposed approach on the newly collected ASL dataset. Our code will be released upon the acceptance of this paper.

Channels	Fusions					
RGB	√			√	√	√
Motion		√		√		√
Depth			√		√	√
mAP	61.8%	52.3%	48.7%	66.4%	65.2%	69.2%

Table 1: Evaluation of fusions of channels for the person-dependent model (some subjects in testing set may appear in the training set.)

Channels	Fusions					
RGB	√			√	√	√
Motion		√		√		√
Depth			√		√	√
mAP	55.6%	47.3%	40.6%	60.8%	55.1%	65.8%

Table 2: Evaluation of fusion of channels (person-independent, subjects in testing do not appear in training set.)

5.2. Influence of Different Channels

In this section, we analyze the influence of each channel in our model: RGB, motion, and depth. Our model training/testing is conducted in two ways: 1) person-dependent analysis, clips are extracted from all videos and pooled together. Training samples and testing samples are divided by the ratio of $8:2$. 2) person-independent analysis, we randomly select one person as the testing target and then use videos of the remaining people as training data.

The person-dependent results are shown in the Table 1. For only RGB channel, the accuracy is 61.8%, while the results for motion and depth channels are 52.3% and 48.7% respectively. From this observation, RGB channels have better performance than the other two input sources. The reason may be that the RGB channels can capture more information than motion and depth channels in our model, since the temporal information is also embedded in the final feature representation. By combining the RGB and motion channels, the accuracy is improved from 61.8% to 66.4%, while the boost by fusing RGB and depth channels is 3.4% compared with only using RGB channel. By combining all three channels, i.e. RGB, motion and depth, the performance can be further boosted to 69.2%.

Table 2 shows the results of person-independent model. The accuracy of RGB, motion and depth channels are 55.6%, 47.3% and 40.6% respectively. By fusing the RGB and motion channels, the performance is boosted

Methods	Person-dependent	Person-independent
LRCN[2]	49.8%	44.5%
Two-stream[26]	52.3%	47.6%
C3D[28]	61.2%	55.4%
3DRCNN	**69.2%**	**65.8%**

Table 3: Comparison of different recognition approaches by the person-dependent and person-independent models.

to 60.8%, while the accuracy of fusing RGB and depth channels is 55.1%. By combining all three channels in the person-dependent model, the recognition accuracy is 65.8%. Similar to the person-dependent results above, all three channels, i.e. RGB, motion and depth, have complementary influence to each other.

5.3. Person Dependent and Independent

In this section, we analyze the ability of our program to be generalized to different people. For person-dependent, clips are extracted from all videos and pooled together. Comparing Table 1 and Table 2, we observe that the RGB channel for the person-dependent model outperforms the person-independent model by a margin of 6.2%. For motion and depth channels, the margins are 5.0% and 8.1% respectively. The reason that motion channel performs better in generalization ability may be that optical flow features focus on the motion information, while RGB and depth channels put more attention on other information, e.g. shape, color and distance from the camera. By fusing all three channels, the difference between person-dependent and person-independent categories is 3.4%. Therefore, the generalization ability of our model is improved by combining all three channels.

5.4. Evaluation of 3DRCNN

In this section, we compare 3DRCNN with Long-term Recurrent Convolutional Networks (LRCN) [2], Two-stream [26] and C3D [28], to demonstrate that by incorporating both spatiotemporal and sequential information, 3DRCNN obtains better performance.

Since our 3DRCNN are implemented on RGB, motion and depth channels, we provide the same channels of information to other methods to make the comparison more reasonable. For LRCN, like our 3DRCNN, three different features are extracted from RGB, motion and depth channels, and then concatenated before sending to the RNN structure. For Two-stream network, a depth stream is added and then concatenated with other two streams. For C3D model, features are extracted from spatial-C3D on RGB frames, motion-C3D on optical flow images and depth-C3D on depth images. These three features are concatenated and then sent to a SVM classifier.

As shown in the Table 3, our proposed 3DRCNN achieves the best performance compared with other methods. For the person-dependent category, 3DRCNN can achieve 69.2% accuracy, while the results for LRCN, Two-stream and C3D are 49.8%, 52.3%, and 61.2% respectively. For the person-independent category, the performance for 3DRCNN is 65.8%, while LRCN is 44.5%, Two-stream is 47.6% and C3D is 55.4%. We argue that LRCN only takes into account the sequential information of consecutive frames, while Two-stream and C3D only consider the spatiotemporal information. However, our proposed 3DRCNN incorporates both spatiotemporal and sequential information, which is the main contribution to the better results in the experiments.

In our experiments, in total 27 ASL words are tested. To understand our proposed approach more clearly, we display the confusion matrices of 16 selected ASL words with most instance clips on the person-dependent category for RGB, motion, depth and fusion of all three channels. For RGB channel, as demonstrated in Figure 4, the word with best performance is "QUESTION" and the words with worst performance are "WHICH" and "QMWG". The failures associated with "WHICH" and "QMWG" are distributed among several words, which demonstrates that these two words are very difficult to be distinguished from other words. For the motion channel, as shown in Figure 5, the word that has the best performance is "ALWAYS", whose accuracy is 82%. For the depth channel, as displayed in Figure 6, the best result resides in the "TONIGHT" entry, which the motion channel does not perform well. The words "YOU", "NOT" and "NO" also have relatively good results for the depth channel, while the word "QMWG" has the worst accuracy.

Figure 7 demonstrates the confusion matrix of the fusion of all three channels. We can observe that the words "ALWAYS" and "QUESTION" have better performance than other words. The word with worst performance is "QMWG", which is consistent in the confusion matrices for individual channels. The "QMWG" sign occurs optionally at the end of yes-no questions, and a question facial expression co-occurs with this sign when it appears. The sign is short in duration and there is variation among humans as to whether it has a single or a double wiggle movement. For a human perceiving this sign, it is possible that the facial expression is an important part of the performance of the sign from a perceptual standpoint. So since our system is not using any facial information then the brief duration of the sign and its variability of performance may explain the poor performance.

6. Conclusion

In this paper, we have proposed a novel multi-modality model, which learns complementary information and embeds the sequential information to the extracted spatiotem-

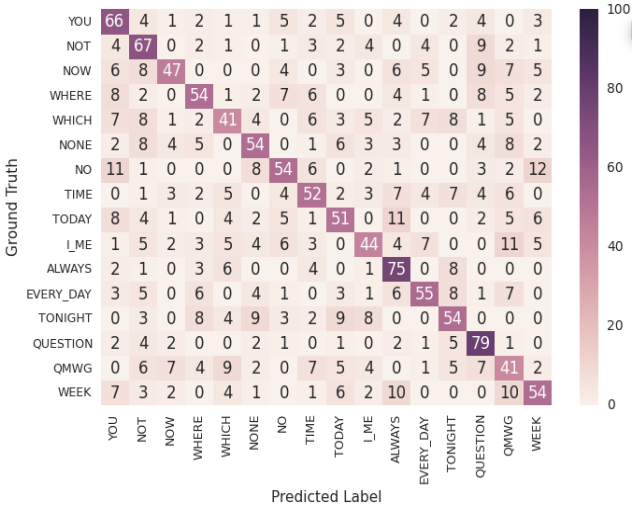

Figure 4: Confusion matrix of RGB channel for the person-dependent model.

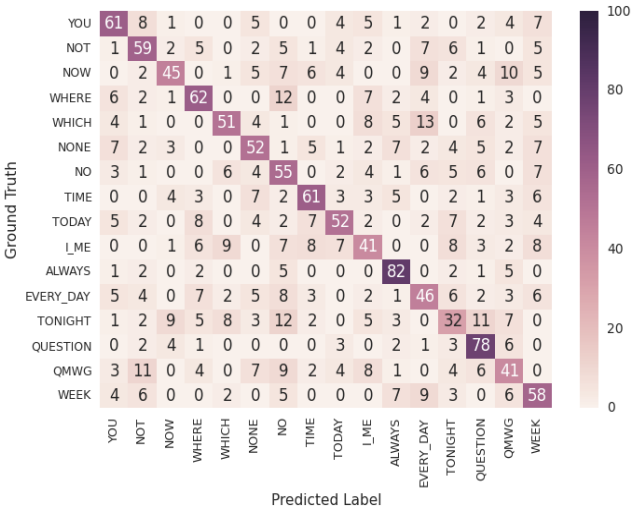

Figure 5: Confusion matrix of motion channel for the person-dependent model.

poral features to recognize actions and localize their temporal boundaries within continuous videos. To validate our proposed method, we collaborate with ASL experts to collect an ASL dataset with full annotation on the temporal domain. The experimental results demonstrate that by fusing multiple sources in our proposed framework, the ASL gestures and their temporal boundaries in continuous videos can be recognized. This technology for identifying the appearance of specific meaningful human actions (in this case, performances of specific ASL words) has valuable applications for technologies that can benefit people who are Deaf

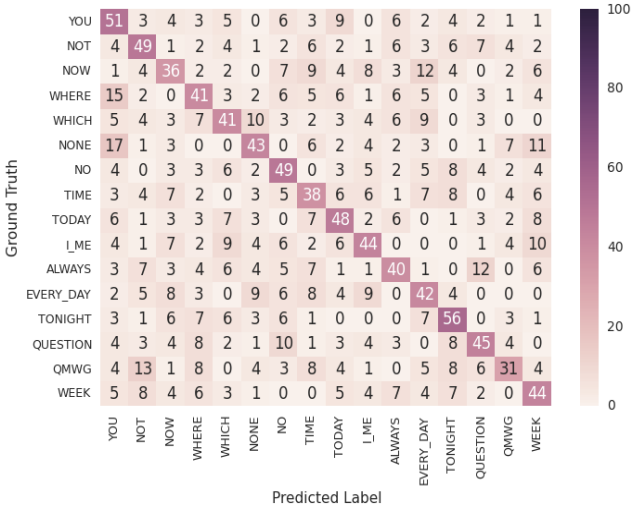

Figure 6: Confusion matrix of depth channel for the person-dependent model.

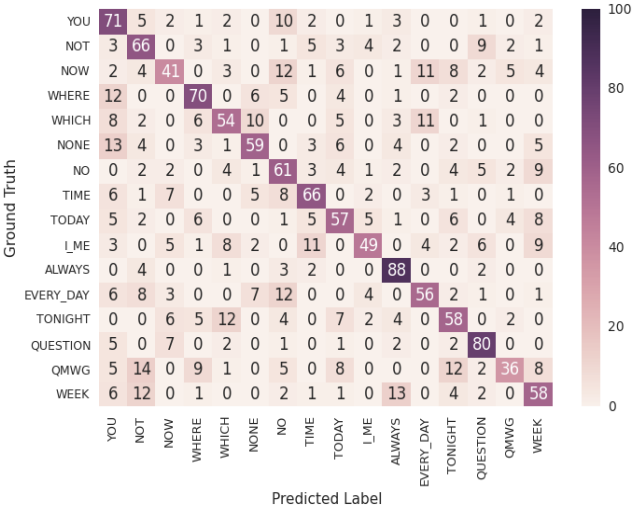

Figure 7: Confusion matrix of all three channels for the person-dependent model.

or Hard-of-Hearing (DHH) [1, 15, 14, 25, 17, 24, 18].

7. Acknowledgement

This work was supported in part by NSF grants EFRI-1137172 and IIS-1400802.

References

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

[2] 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. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2625–2634, 2015.

[3] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. *arXiv preprint arXiv:1310.1531*, 2013.

[4] B. Fernando, E. Gavves, J. Oramas, A. Ghodrati, and T. Tuytelaars. Rank pooling for action recognition. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 39(4):773–787, 2017.

[5] J. Forster, C. Schmidt, T. Hoyoux, O. Koller, U. Zelle, J. H. Piater, and H. Ney. Rwth-phoenix-weather: A large vocabulary sign language recognition and translation corpus. In *LREC*, pages 3785–3789, 2012.

[6] S. Gattupalli, A. Ghaderi, and V. Athitsos. Evaluation of deep learning based pose estimation for sign language recognition. In *Proceedings of the 9th ACM International Conference on Pervasive Technologies Related to Assistive Environments*, page 12. ACM, 2016.

[7] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 580–587. IEEE, 2014.

[8] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *Computer Vision–ECCV 2014*, pages 346–361. Springer, 2014.

[9] M. Huenerfauth, E. Gale, B. Penly, S. Pillutla, M. Willard, and D. Hariharan. Evaluation of language feedback methods for student videos of american sign language. *ACM Transactions on Accessible Computing (TACCESS)*, 10(1):2, 2017.

[10] S. Ji, W. Xu, M. Yang, and K. Yu. 3d convolutional neural networks for human action recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(1):221–231, 2013.

[11] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. *arXiv preprint arXiv:1412.2306*, 2014.

[12] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1725–1732. IEEE, 2014.

[13] 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, 2015.

[14] O. Koller, H. Ney, and R. Bowden. Deep learning of mouth shapes for sign language. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 85–91, 2015.

[15] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3793–3802, 2016.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.

[17] P. Kumar, P. P. Roy, and D. P. Dogra. Independent bayesian classifier combination based sign language recognition using facial expression. *Information Sciences*, 428:30–48, 2018.

[18] W. Liu, Y. Fan, Z. Li, and Z. Zhang. Rgbd video based human hand trajectory tracking and gesture recognition system. *Mathematical Problems in Engineering*, 2015, 2015.

[19] Z. Liu, F. Huang, G. W. L. Tang, F. Y. B. Sze, J. Qin, X. Wang, and Q. Xu. Real-time sign language recognition with guided deep convolutional neural networks. In *Proceedings of the 2016 Symposium on Spatial User Interaction*, pages 187–187. ACM, 2016.

[20] P. Lu and M. Huenerfauth. Cuny american sign language motion-capture corpus: first release. In *Proceedings of the 5th Workshop on the Representation and Processing of Sign Languages: Interactions between Corpus and Lexicon, The 8th International Conference on Language Resources and Evaluation (LREC 2012), Istanbul, Turkey*, 2012.

[21] R. E. Mitchell, T. A. Young, B. Bachleda, and M. A. Karchmer. How many people use asl in the united states? why estimates need updating. *Sign Language Studies*, 6(3):306–335, 2006.

[22] P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree, and J. Kautz. Online detection and classification of dynamic hand gestures with recurrent 3d convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4207–4215, 2016.

[23] C. Neidle and C. Vogler. A new web interface to facilitate access to corpora: Development of the asllrp data access interface (dai). In *Proc. 5th Workshop on the Representation and Processing of Sign Languages: Interactions between Corpus and Lexicon, LREC*, 2012.

[24] M. Palmeri, F. Vella, I. Infantino, and S. Gaglio. Sign languages recognition based on neural network architecture. In *International Conference on Intelligent Interactive Multimedia Systems and Services*, pages 109–118. Springer, 2017.

[25] L. Pigou, M. Van Herreweghe, and J. Dambre. Gesture and sign language recognition with temporal residual networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3086–3093, 2017.

[26] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems*, pages 568–576, 2014.

[27] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *arXiv preprint arXiv:1409.4842*, 2014.

[28] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4489–4497, 2015.

[29] C. B. Traxler. The stanford achievement test: National norming and performance standards for deaf and hard-of-hearing students. *Journal of deaf studies and deaf education*, 5(4):337–348, 2000.

[30] G. Varol, I. Laptev, and C. Schmid. Long-term temporal convolutions for action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[31] X. Yang, P. Molchanov, and J. Kautz. Multilayer and multimodal fusion of deep neural networks for video classification. In *Proceedings of the 2016 ACM on Multimedia Conference*, pages 978–987. ACM, 2016.

[32] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 4694–4702, 2015.