Supplementary Material: Fingerspelling recognition in the wild with iterative visual attention

Bowen Shi[1], Aurora Martinez Del Rio[2], Jonathan Keane[2], Diane Brentari[2]
Greg Shakhnarovich[1], Karen Livescu[1]
[1]Toyota Technological Institute at Chicago, USA [2]University of Chicago, USA
{bshi,greg,klivescu}@ttic.edu {amartinezdelrio,jonkeane,dbrentari}@uchicago.edu

1. Pose estimation on fingerspelling data in the wild

To illustrate the difficulty of using pose estimation for our fingerspelling recognition task, we ran an off-the-shelf pose estimator, OpenPose (https://github.com/CMU-Perceptual-Computing-Lab/openpose), on our fingerspelling data. Example results are shown in Figure 1. OpenPose is a person keypoint detection library including a hand keypoint estimation module. Due to the visual challenges in our fingerspelling data, signing hands are not detected in some frames. Furthermore, the hand pose is often not correctly estimated even if the signing hand is detected successfully.

2. Face detector

The model and training data for the face detector we use have been described in the main paper. Here we provide additional detail on how we apply the face detector in the **face ROI** and **face scale** setups, in particular on how the ROI is extraced and scaled.

To save computation, the face detector is run on one in every five frames per sequence, interpolating the detections for the remaining 80% of the frames. If only one face is detected, we take the average of all bounding boxes for the whole sequence. In cases where multiple faces are detected, we first find a smooth "face tube" by successively taking the bounding box in the next frame that has the highest IoU with the face bounding box in the current frame. For every tube, a motionness score is defined as the average value of optical flow within a surrounding region ($3\times$ size of bounding box). Finally the tube with the highest score is selected and again the box is averaged over the whole sequence. In cases where face detection fails, we use the mean of all face bounding boxes detected in all images of the same size in the training set. We empirically observe that the failure case where no

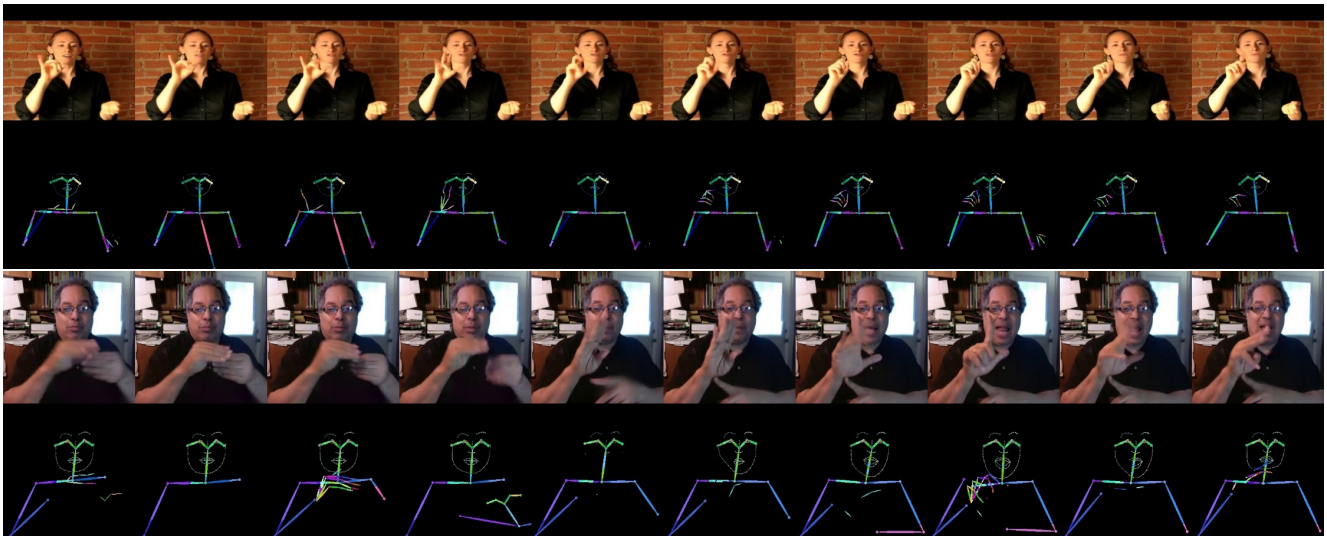


Figure 1: Examples of pose estimation failure on fingerspelling data from ChicagoFSWild.

face is detected is rare ($\sim 0.5\%$ of the training set).

In the **face ROI** setting, a large region centered on the detected face bounding box is cropped and resized to serve as input. This is because the signing hand(s) are spatially close to the face during fingerspelling. Specifically we crop a region centered on the bounding box which is 3 times larger. The ROI is resized with a ratio of $\frac{224}{max(w_{roi},h_{roi})}$ and then padded on the short side to make a squared target image of size 224×224.

In the **face scale** setting, we only scale the original frame based on the size of the face bounding box to avoid artifacts arising from cropping. The purpose of scaling is to make the scale of hands in different input sequences roughly uniform. As our data are from videos with a large variety of viewpoints and resolutions, the scale of the hands varies over a wide range. For instance the proportion of the hand in an image from a webcam video can be several times larger than that in an image from a third-person view. Specifically we pre-set a base size b (36 in our experiments) for the face bounding box. Input images of original size $W_I \times H_I$ with a bounding box of size $w_I \times h_I$ are rescaled with a ratio of $\frac{b}{max(w_I,h_I)}$. If the image area is larger than 224×224 after rescaling, we further rescale by a ratio of α to ensure the resulting image has at most 224×224 resolution due to memory constraints. α is multiplied in the iterative zooming-in for that input sequence.

3. Signing hand detector

We adopt the signing hand detector used in [4], made available by the authors. Unlike a general hand detector, the objective here is to detect the signing hand.[1] The detector is based on Faster R-CNN [3] and takes both the RGB image frame and corresponding optical flow as input. VGG-16 [5] is used as the backbone architecture. Unlike a general object detector, only the first 9 layers of VGG-16 are preserved and the stride of the network is reduced to 4. This is done so as to capture more fine details, since the signing hand tends to be small relative to the frame size. To enforce sequence-level smoothness, framewise bounding boxes are linked to a "signing tube". The linking process takes into account the IoU between bounding boxes in consecutive frames. More details on the hand detector can be found in [4].

Apart from the original hand detector used in [4], we also experimented with variants including using all convolutional layers of VGG-16 and concatenating feature maps in different layers to make it multi-scale as in [2, 1]. We did not observe any improvement from these variants, which may be because those more complex networks suffer from overfitting due to the limited amount of hand annotations. In addition, we notice that the majority of errors made by the hand detector consist of confusion between signing and non-signing hands instead of between hands and background objects. Typical errors can be seen in Figure 2. Thus it is difficult to mitigate the issue of data scarcity by simply augmenting our training data with external hand datasets from other domains.

[1] A large proportion of the video frames collected in the wild contain more than one hand.

4. Experiments on zooming vs. enlarging, prior vs. no prior

We ran the following experiment to show the benefits of distraction removal obtained by the zooming employed in iterative attention, in addition to the increase in resolution. In particular, we compare the accuracy of zooming at ratio R and enlarging the input images by $\frac{1}{R}$ in the **face ROI** setting. For this experiment, R is set to 0.9^3, corresponding to the zooming ratio we use in the first iteration. Comparison on smaller ratios is not feasible due to GPU memory constraints (12GB in our case). For both zooming and enlarging, the resolution of the signing hand is the same. As can be seen from Table 1, zooming outperforms enlarging. When the prior map is used, the gap between the two approaches is small. This is mainly because distracting portions can be filtered via the motion-based prior in our model. The gain of zooming becomes much larger when we do not use optical flow as a complementary prior, demonstrating the benefit of distraction removal in our approach. Additionally, the motion-based prior has a negligible effect on the accuracy of our approach in this setting.

$R = 0.9^3$	Zooming	Enlarging
with prior	**39.6**	39.3
without prior	**39.8**	38.1

Table 1: Accuracy comparison between zooming and enlarging in the **face ROI** setting.

5. Experiments on robustness to face detection errors

A face detector is used in two experimental setups: **face ROI** and **face scale**. To see how robust the model is to face detection errors, we add noise to the bounding box output by the face detector. Specifically, two types of noise were separately added: size noise and position noise. For size noise, we perturb the actual face detection boxes by multiplying the width and height of the box by factors each drawn from $\mathcal{N}(1, \sigma_s^2)$. For position noise, we add values drawn from $\mathcal{N}(0, \sigma_p^2)$ to the center coordinates of the face detection boxes. Note that position noise only affects the **face ROI** experiments. We vary σ_s, σ_p and show results in Table 2, 3. Overall we find that position noise has a smaller impact on accuracy compared to size noise. The **face scale**

setup, where no cropping is done in pre-processing, is more robust to size noise than the **face ROI** setup is. Adding size noise brings a small improvement in this setting, which provides evidence that the face detector we use is not perfect.

σ_s	IoU	Face ROI	Face Scale
0.0	1.000	**45.6**	42.9
0.1	0.858	45.2	42.7
0.2	0.741	44.7	43.3
0.3	0.641	44.3	**44.0**
0.4	0.556	42.6	43.3

Table 2: Impact of size noise on letter accuracy for **face ROI** and **face scale** setups. IoU is measured between the perturbed and original bounding boxes.

σ_p	IoU	Face ROI
0.0	1.000	**45.6**
0.5	0.780	45.2
1.0	0.621	45.0
1.5	0.499	44.6
2.0	0.402	44.2

Table 3: Impact of position noise on letter accuracy for the **face ROI** setup. Note the **face scale** is not affected by position noise. IoU is measured between the perturbed and original bounding boxes.

6. Iterative attention vs. off-the-shelf signing hand detector

Iterative attention serves as an implicitly learned "detector" of signing hands. We compare the performance of this detector with a separately trained signing hand detector here. The signing hand detector is the one used in [4] and has been described in the previous section. We convert the iterative attention ROI to an explicit detector through the following steps: take the input image of the last iteration, backtrack to the original image frame to get its coordinates, and use these coordinates as the bounding box. We take a model trained in the **face ROI** setting and compare it with an off-the-shelf detector. Figure 2 shows example sequences from the ChicagoFSWild dev set, where our approach successfully finds signing sequences while the off-the-shelf detector fails. For quantitative evaluation, we take the dev set of hand annotation data in ChicagoFSWild, which includes 233 image frames from 19 sequences, and remove all frames with two signing hands. That amounts to 200 image frames in total. We compute average IoU and miss rate between the target bounding box and ground truth. The miss rate is defined as 1-intersection/ground-truth area. As the two detectors have different IoU's and miss rates, for ease of comparison we resize the bounding box of the off-the-shelf detector to keep its miss rate consistent with that of the iterative-attention detector. As is shown in Table 4, our detector almost doubles the average IoU of the off-the-shelf detector at the same miss rate. Though numerical differences between IoU's may be exaggerated due to the small amount of evaluation data, the effectiveness of our approach for localization of signing hands can also be inferred from improvements in recognition accuracy.

	Off-the-shelf [4]	Iterative-Attn
Avg IoU	0.213	**0.443**
Avg Miss Rate	0.158	0.158

Table 4: Comparison of IoU between an off-the-shelf signing hand detector and a detector produced by iterative attention.

References

[1] T. Hoang Ngan Le, Kha Gia Quach, Chenchen Zhu, Chi Nhan Duong, Khoa Luu, and Marios Savvides. Robust hand detection and classification in vehicles and in the wild. In *CVPR workshop*, 2017. 2

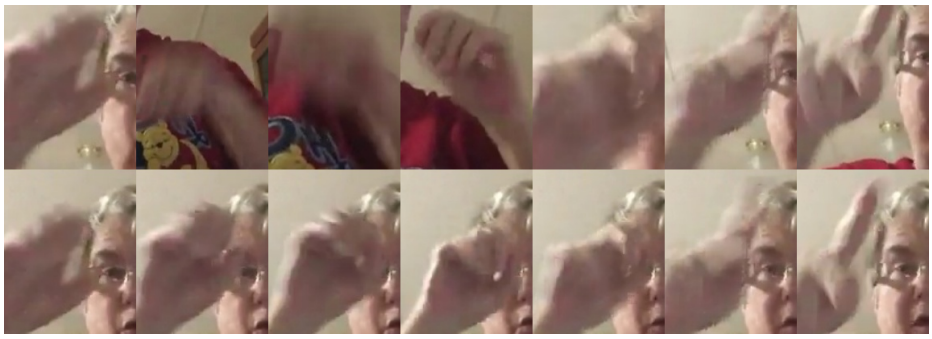
[2] T. Hoang Ngan Le, Chenchen Zhu, Yutong Zheng, Khoa Luu, and Marios Savvides. Robust hand detection in vehicles. In *ICPR*, 2016. 2

[3] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 2

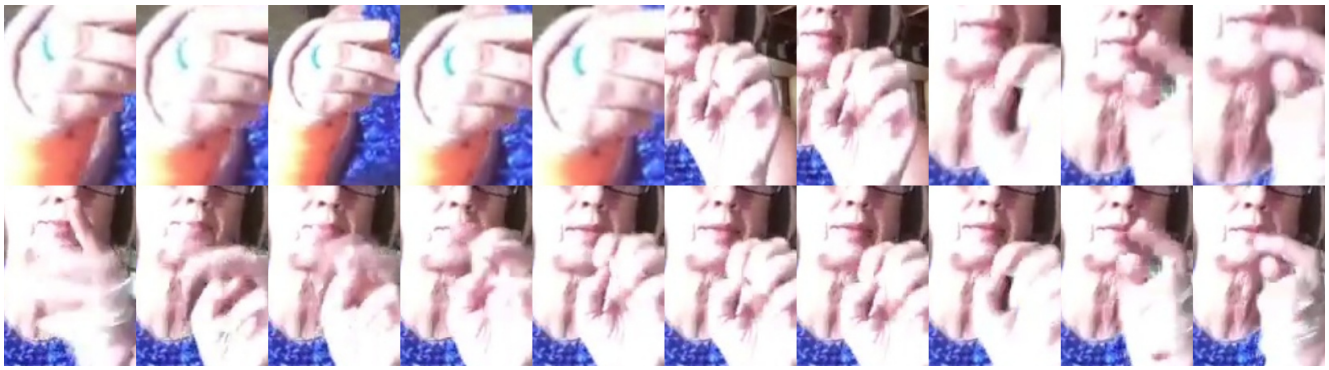
[4] Bowen Shi, Aurora Martinez Del Rio, Jonathan Keane, Jonathan Michaux, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. American Sign Language fingerspelling recognition in the wild. In *SLT*, 2018. 2, 3, 4

[5] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 2

(1).



(2).



(3).

(4).

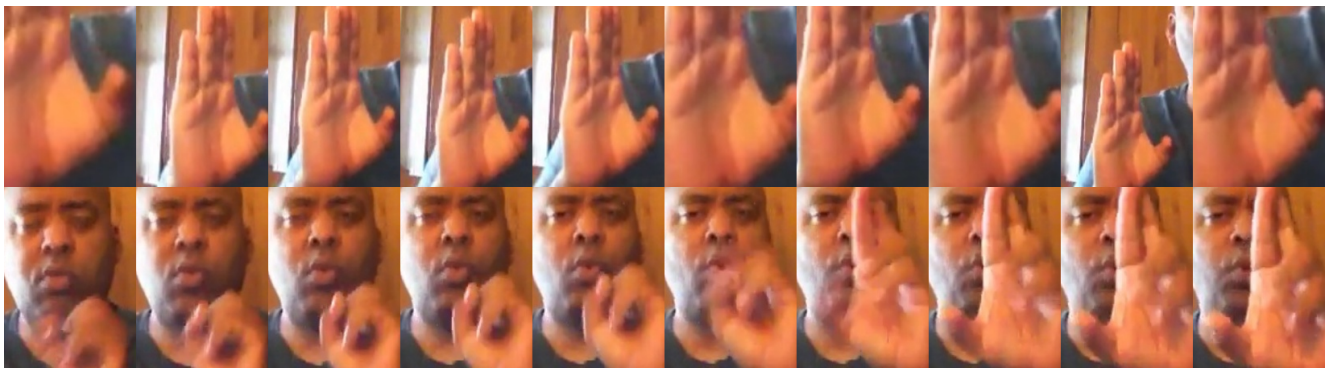


Figure 2: Signing hands detected by the iterative attention detector vs. the off-the-shelf signing hand detector [4], taken from the ChicagoFSWild dev set. In each example, the upper row is from off-the-shelf detector and the lower row is from iterative attention. Signing hands are successfully detected by iterative attention in all cases.
Errors made by the off-the-shelf detector: In (1) and (2), bounding boxes are switched between signing and non-signing hand; in (3), the detected signing hand is incomplete; in (4), the non-signing hand is mis-detected as the signing hand. Note that sequence-level smoothing has already been incorporated in the off-the-shelf detector.