Action Recognition: GORA Based Frame Selection as a Preprocess in Deep Learning

Chao Ni

September 2018

1 Frame Selection

Frame selection as a pre-process for video dataset has been proven an efficient approach in training model. We leverage this idea to illustrate that our global optimal reparameterization algorithm (GORA) can be utilized in frame selection. The GORA can be seen as a preprocess for frame selection in deep learning problems. It ignores frames with overlapped information and retains essential frames representing the action. In our verification experiment, the videos are firstly be extracted out into a sequence of frames. A certain number of frames are then be selected based on GORA and then be put into the deep learning architecture for training. Fig.1 shows the process of our verification for GORA on deep learning application.

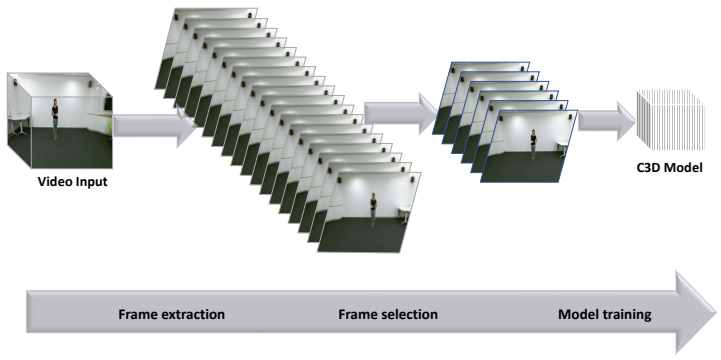


Figure 1: Frame selection as a preprocess in deep learning architecture training.

2 Datasets and neural network model

In our experiment, we utilize the NTU dataset [2], which is a large scale data set for human action recognition. We choose eight different actions with about 80 videos for each label as our data. Among them 64% are for training, 16% for validation and 20% for testing. The eight action labels are: put something inside pocket / take out something from pocket, hopping (one foot jumping), jump up, make a phone call/answer phone, playing with phone/tablet, typing on a keyboard, pointing to something with finger and taking a selfie. We demonstrate our frame selection algorithms for the 3D convolution architecture developed by Tran [3]. We want to show our frame selection method is not limited to certain models, but can be regarded as a general preprocess for other deep learning architectures.

3 Experimental Setup

We use the C3D model [3] pretrained on UCF-101 for our analysis on frame selection. We set the hyperparameters trained by UCF-101. The learning rate is 1e-4 and we crop each input frame into size of 112 by 112 around its center. We set the batch size as 10 and crop each video into 16 frames as a clip. For videos which has more than 16 frames, we randomly choose 16 continuous frames. This is a baseline and in order to verify GORA, we make a comparison between the randomly selection, uniformly selection and GORA.

4 Implementation

4.1 GORA reparameterization function formulation

The initial length of the video varies from about 80 frames to 150 frames. In the uniform selection case, the selection is simple as it is not relevant to the exact containing of the picture. we will present the concrete deduction of GORA based frame selection mapping function, and thus providing the corresponding frame index for time instances. In our experiment, each frame of the video have a size of (426, 240) with three channels. The mathematical deduction of the reparameterization can be found at [1]. The main result are as follows. The problem is to minimize the integral:

$$J = \int_0^1 ||X'(\tau)||^2 \dot{\tau}^2 dt \tag{1}$$

where J is the cost function, τ is the reparameterization mapping function, and X is a general value of the frame in our case. It has been proven that this optimization problem has a global optimal solution $\tau(t)$. The unique solution can be obtained by the inverse of:

$$F(x^*) = \frac{1}{c} \int_0^{x^*} ||X'(\sigma)|| d\sigma = t \tag{2}$$

the global solution is then $\tau = F^{-1}(t)$. When operated on the frames, we firstly convert the RGB images into grey images. Then the images are reshaped into a vector whose size is the product of the height and width of the image. Therefore, each frame is connected to a fixed-dimension vector $X(n)$, where n is the index of the frame. We interpolate the derivative by forward difference method.

$$X'(t_n) = \frac{||X(t_{n+1}) - X(t_n)||_2}{t_{n+1} - t_n} \tag{3}$$

Notice that in our dataset, the background of each action remains the same, thus we can express the derivative between frames by difference in pixel values. The difference between frames are minus if we take a close look at the total amount of pixels. The minimum value in the element of $X'(t)$ is x_{min}, we do a subtraction on each element to get a modified vector $X'_m(t)$.

$$X'_m(t) = X'(t) - x_{min} \tag{4}$$

We do this subtraction in order to eliminate the effect of background noises. Based on the reparameterization function (2), we can solve for the corresponding frame index for every time instances. An example for reparameterization mapping function is shown in Fig. 2. We project the video play duration into a unit time. In this unit time duration, we choose 16 time instances uniformly distributed and take the corresponding frame as the neural network input. For comparison, we plot two frame sequences as Fig. 3.

4.2 Training Performance and Comparison

On the experiment setup described above, we trained the model from the scratch for 25 epochs each in three frame selection cases: Randomly selected continuous 16 frames; uniformly selected frames; GORA based selected frames. Fig. 4 shows the performance difference among these three different frame selection methods. The GORA based selected frames achieves faster learning speed and higher recognition accuracy, the continuous random frames loses most information and has weakest training performance.

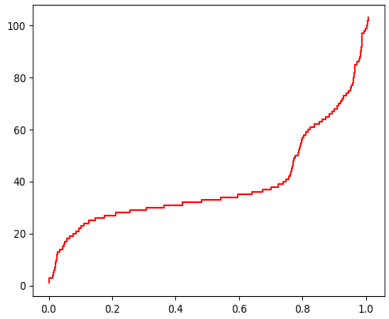
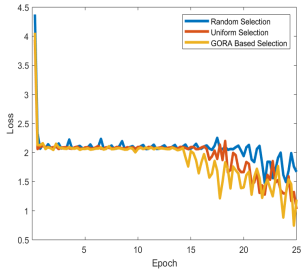
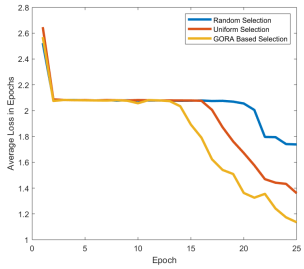


Figure 2: An example of reparameterization mapping function. The x axis represents the time horizon, we project the whole action into a unit time, the y axis represents the corresponding frame index. The flat part of the line shows that during this period, the difference between frames changes fast, the frames during this period contain more information compared to others in steep part of the line.

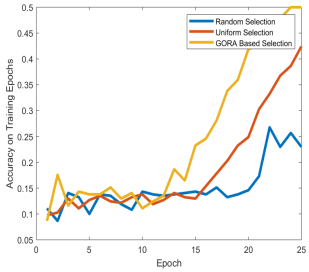
Figure 3: The frame sequence on the top represents the result input of the C3D architecture under the uniform selection method, the sequence on the bottom is the result of the reparameterization mapping. This figure shows the difference of the selected frames.



(a) The loss figure along the training process in 25 epochs.



(b) The average loss along the training process in 25 epochs.



(c) The tendency of training accuracy as the epoch increases.

Figure 4: Loss and accuracy tendency for three frame selection methods. We train the model for 25 epochs with batch size of 10. There are four bathes in every epoch. The experiment is reproduced several times and the result (tendency) is similar. Our GORA based selection achieves faster learning speed and higher recognition accuracy

Fig. 4(c) shows the performance difference even more clearly. The GORA based frame selection method achieve better performance in training the model compared the other two. We then continue to train the model until the loss arrives its global minimum. we test the model separately and the results are shown in Tab. 3. In this example, we can see that the final gap between GORA based selection and uniform selection is smaller than in epoch 25, which gives some ideas that these two methods extract out similar information. This happens when the initial video is nearly at its uniform temporal distribution and resulting GORA based selected frame are similar to those from the uniform selection. However, the GORA based frame selection still achieves faster learning speed and best training performance among three because of its smoother distribution.

	Acc% (Epoch 25)	Acc% (Epoch 100)
Random Selection	18.9	41.4
Uniform Selection	38.1	70.0
GORA Based Selection	44.9	70.1

Table 1: Accuracy on the testing set with different frame selection methods.

4.3 Background subtraction

Background subtraction is also another efficient and broadly used method in dealing with video data. In our case, we apply our GORA preprocess mainly on the videos whose background remains still in the video periods, which means most of the information the frame contains are overlapped. We can then subtract the background, thus eliminating the redundant information and speeding up the training process. We use the Gaussian Mixture-based Background Segmentation Algorithm [5] [4]. The frame sequence after the subtraction is shown in Fig. 5. It is shown in Fig.6 the difference among three selection methods after

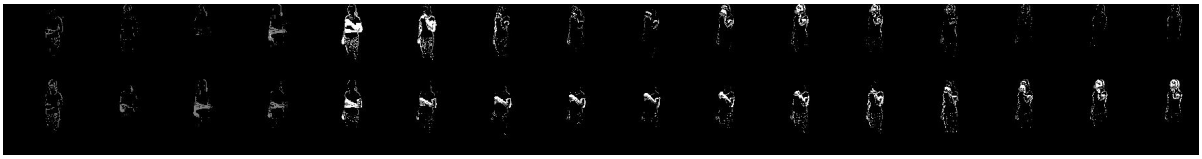


Figure 5: The frame sequence after the background subtraction.

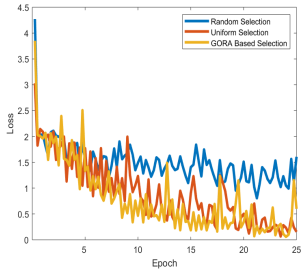
background subtraction. Compared with raw pictures without background subtraction, the learning is speed up in all three frame selection methods. But the performance difference remains the same as the raw frames. The GORA based frame selection achieves best performance among the three. Tab. 2 shows the quantitative comparison when operated with background subtraction. After 25 epochs, the GORA based selection method beats the rest two. We find the loss almost converges to its minimum after 25 epochs because of the limited sample size. Although the superiority over the uniform selection is minus, it achieves faster learning speed.

	Acc%
Random Selection	45.5
Uniform Selection	65.4
GORA Based Selection	70.0

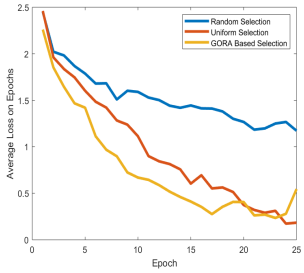
Table 2: Accuracy on the testing set with different frame selection methods after background subtraction.

4.4 More analysis on the data

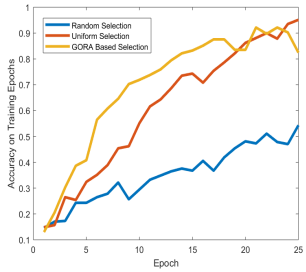
Apart from experiments we described above, we also change the data to see whether the GORA still achieve better performance when confronted with different kinds of actions and videos. We found that among



(a) The loss figure along the training process in 25 epochs after background subtraction.



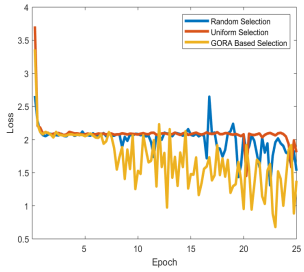
(b) The average loss along the training process on 25 epochs after background subtraction.



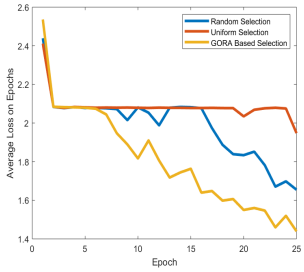
(c) The tendency of training accuracy as the epoch increases after background subtraction.

Figure 6: Loss and accuracy tendency for three frame selection methods after background subtraction. We train the model for 25 epochs with batch size of 10. There are four bathes in every epoch. The experiment is reproduced several times and the result (tendency) is similar. Our GORA based selection achieves faster learning speed and higher recognition accuracy

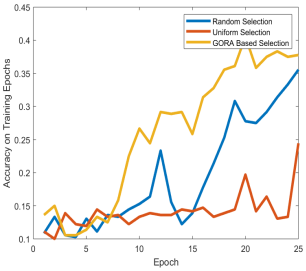
three frame selection methods, GORA always wins while the rest two sometimes swap their positions in the competition. Fig. 7 presents an example when we choose another eight different kinds of action.[1]



(a) The loss figure along the training process on 25 epochs.



(b) The average loss along the training process on 25 epochs.



(c) The tendency of training accuracy as the epoch increases.

Figure 7: Loss and accuracy tendency for three frame selection methods under different action labels and video data. In this case, the GORA based frame selection still outweighs among three, but random selection produces better result than uniform selection method.

	Acc% (Epoch 25)	Acc% (Epoch 100)
Random Selection	24.1	58.5
Uniform Selection	20.9	56.4
GORA Based Selection	31.8	64.5

Table 3: Another try: accuracy on the testing set when dataset changes. The GORA based frame selection method still wins while the random selection method beats uniform selection method.

5 Conclusion and Discussion

In our verification of GORA in frame selection application as a preprocess for deep learning training, we proof the advantage in learning speed and training performance of GORA. With GORA based selection method, we can extract out as more information of the video as possible in limited number of frames. We can then

[1]Drink water, sit down, stand up, kick something, put something inside pocket / take out something from pocket, point to something with finger, walk towards each other, walk apart from each other.

grab the essence of the action and the better performance in training is intuitive. We can also see some ideas from the experiments that the advantage of GORA is independent of the data. However, the actions we used in the experiment are still limited in kinds and the videos which can be applied with GORA are limited to those who are not encountered huge difference in its background.

References

[1] Gregory S Chirikjian. Signal classification in quotient spaces via globally optimal variational calculus. *arXiv preprint arXiv:1705.03744*, 2017.

[2] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+d: A large scale dataset for 3d human activity analysis. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[3] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[4] Zoran Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 2, pages 28–31. IEEE, 2004.

[5] Zoran Zivkovic and Ferdinand Van Der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern recognition letters*, 27(7):773–780, 2006.