Yipeng Lu Research Report

Yipeng Lu

Dec 2020

1 Introduction

Detecting human actions from RGB camera is a key component of many smart home applications. The populations of old people in many countries are increasing fast as the improvement of life quality. Because old people usually suffer from several diseases, and is more easily injured than young people, it is quite dangerous when they are living alone, which is quite a common situation. If an algorithm can detect severe illnesses of a people through RGB camera, and report to the nearest hospital immediately, it could save the person's life.

However, in the meantime, human action recognition is one of the most challenging tasks in the field of computer vision. Occlusions, changes in view point or background, essential body parts missing, similarity in actions to be classified and a lot more factors could easily influence the performance of the detection tasks. Also, as we are using only one single RGB camera, the input data tends to lack 3D structure which is also one critical attribute of human actions.

The objective of this report is: First, train a neural network that can classify human actions that indicate severe illnesses of the person on a existing dataset. Second, find moments that a person is performing chest pain actions in the YouTube videos, and collect a large dataset of such moments.

2 Dataset

The dataset used in this report is NTU RGB+D dataset[8]. It contains 56,880 video samples which are recorded by 106 subjects under 17 setups using 3 cameras concurrently. There are mainly three types of actions in the dataset: daily actions, mutual actions, and medical conditions actions. For each sample, RGB videos, depth map sequences, 3D skeletal data, and infrared (IR) videos are provided. There are two types of standard evaluation for the classification performances, which are cross-subject evaluation and cross-setup evaluation.

In this report, RGB videos of class 1 to class 49 are used, and cross-subject evaluation is adopted to split the data and evaluate the models. Videos are split into train set, validation set and test set. Train set contains 22834 videos which have subject ids of 9, 13, 14, 15, 16, 17, 18, 19, 25, 27, 28, 31, 34, 35, 38.

Validation set contains 7830 videos which have subject ids of 1, 2, 4, 5, 8. Test set contains 13346 videos which have subject ids of remaining subjects.

3 Methodology

3.1 RNN

3.1.1 Obtain skeleton sequences

Skeleton data is a list of positions in coordinate format that depict the location of several important human body part, such as arms, head, and legs, etc. Skeleton sequence is a list of skeleton data, with each skeleton data from one frame of a video. Recently, skeleton based human representations have been intensively studied and kept attracting an increasing attention, due to their robustness to variations of viewpoint, human body scale and motion speed as well as the realtime, online performance[3]. Although structured-light cameras enable us to retrieve the 3D human skeletal information in real time[3], and NTU RGB+D dataset does provide 3D skeleton data, it was not used as the goal of this report is to predict action based on solely RGB videos. Therefore, the skeleton sequences were extracted from openpose[1]. Openpose uses a non-parametric representation, which is referred to as Part Affinity Fields (PAFs), to learn to associate body parts with individuals in the image, and can detect the pose of multiple person in videos, and output the location information of the 25 body parts of each person with decent accuracy.

Although openpose can achieve high accuracy in detecting body parts, and the video samples from NTU RGB+D dataset are clear and contain full body of one subject(the person who perform the target action), there are still cases that multiple persons are detected. This is caused by either the person not performing the target action is also detected, or a person-like non-person object is recognized as person by openpose. To solve this problem, a simple logic was designed so that the most centered person is regarded as subject if multiple persons are detected. Also, a probability threshold of 0.5 is set on average probability of successfully detected body part(body part with probability greater than 0). After performing these two steps, it is more likely that the correct person will be picked when multiple persons are detected by openpose.

3.1.2 Generate input tensor with normalized coordinate

In order to feed the skeleton sequences into a neural network, the shape of the data is fixed by sampling 24 frames from each video with different frame number and duration. For each frame, the positions of 25 body parts are padded to a 50-dimension vector and normalized by:

$$p_{normalized} = (p - p_{min})/(p_{max} - p_{min})$$

Where p is the original coordinate value, p_{max} is the maximum value of the coordinate value over the entire video, and p_{min} is the minimum value of the

coordinate value over the entire video. After padding and normalization, a tensor of shape (24,50) is generated for each video sample.

3.1.3 Generate input tensor with pose feature and motion feature

Inspired by the idea in [7], the normalized coordinate was substituted by pose feature and motion feature as input tensor.

Pose feature consists of joint-joint distances and joint-joint orientations. Joint-joint distances are the distances of two body parts within one frame. There are 25 body parts and a total of 300 pairs of body parts, therefore the dimension of joint-joint distances in pose feature is 300. To normalize, each joint-joint distance within one frame is divided by the maximum value among 300 joint-joint distances within that frame. Joint-joint orientations are the unit vectors of the directions of two body parts within one frame. Each unit vector has x coordinate value and y coordinate value, therefore the dimension of joint-joint orientations in pose feature is 600. By concatenating joint-joint distances and joint-joint orientations, the 900-dimension pose feature is obtained.

Motion feature consists of joint distances and joint orientations. Joint distances are the distances of the same body part in two adjacent frames. There are 25 body parts in total, therefore the dimension of joint distances is 25. To normalize, each joint distance is divided by the maximum value of joint distances within one video. Coordinate distances are the unit vectors of the directions of the same body part in two adjacent frames, and have a dimension of 50.

The input tensor has a shape of (23,975), which is formed by concatenating the pose feature and motion feature frame by frame. Note that the value of any distances or orientations are set to 0 if anyone of the involved body part has (0,0) position, which indicate that openpose cannot detect that body part in that frame.

3.1.4 Architecture of the RNN model

Recurrent neural networks can handle sequence information with varied lengths of time steps. This transforms the input to a internal hidden state ht at each time step. The network passes the state along with the next input to the neuron, time step after time step[11], which makes it especially effective for processing sequence data. Compared with LSTM, GRU has a simpler structure and can be computed faster. Bidirectional GRU looks at a sequence both ways, thus sometimes could perform better than simple GRU. Stacking recurrent layers is useful as it can increase the representational power of a network. The overall structure of the RNN model is obtained from [11], as it shows promising performances on 2D video action recognition tasks. The model consists of two stacked bidirectional GRU layers, followed by a batch normalization layer to reduce the internal covariate shift and speed up the training process, a dropout layer to reduce over-fitting, and two densely connected layers as the final classifier. The architecture of the RNN model is shown in figure 1.

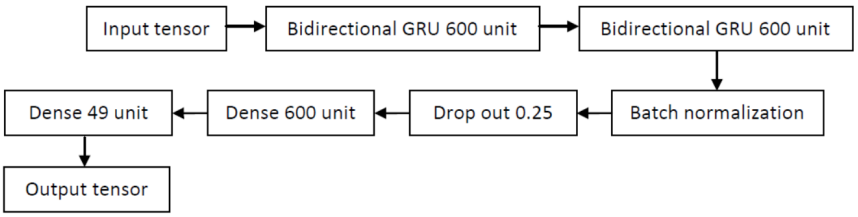

figure 1

The training parameters for the RNN model are: batch size to be 128, optimizer to be ADAM, learning rate to be 1e-4.

3.1.5 Performance of input tensor with normalized coordinate

During training, the epoch with the best validation loss is saved. After training for 100 epochs, the RNN model with normalized coordinate input tensor achieved 50.04% validation accuracy over the 49 action classes. As a better validation accuracy was obtained by using input tensor with pose feature and motion feature, the plots of accuracy and losses will not be shown here.

3.1.6 Performance of input tensor with pose feature and motion feature

After training for 100 epochs, the RNN model with pose feature and motion feature input tensor achieved 73.44% validation accuracy over the 49 action classes, which is much better compared with using the normalized coordinate input tensor. The plots of training accuracy vs. validation accuracy, and training loss vs. validation loss are shown in figure 2, and figure 3.

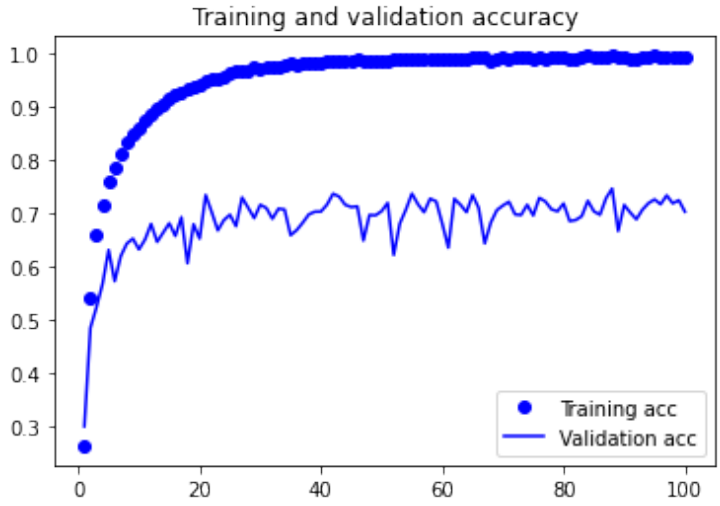

figure 2

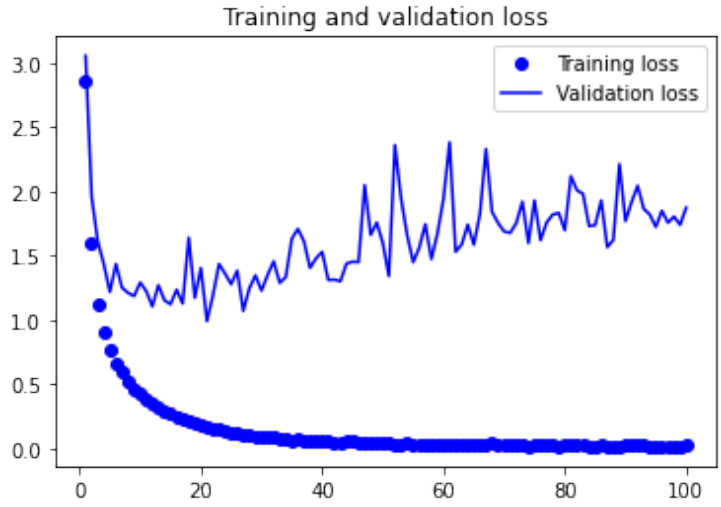

figure 3

3.2 I3D

3.2.1 Generate input tensor of RGB information

Although skeleton sequences are good for action recognition, it could not contain information other than positions of body parts, which may also be important in order to successfully classify the action, such as objects involved with the action. Therefore RGB information from a video should also be considered.

For many pretrained image classification neural networks, the standard image input size is (224,224). Considering that the videos from NTU RGB+D dataset are of resolution 1920*1080, and the subjects that performing the actions may not locate center in every frame, some image processing operations such as cropping and resizing is needed.

9 frames are sampled from each video. For every frame of a video, firstly, a square containing the entire body of the subject is cropped and resized to (249,249,3) according to the body parts locations generated by openpose. The width-height ratio does not change during the resizing operation to avoid distortion. Then, the square is randomly cropped to (224,224,3), and flip horizontally randomly to make the data more diverse and prevent overfitting. Note that the randomness only applies to distinct training videos in distinct epochs, the cropping of frames within the same training video in one specific epoch is the same, and videos in validation set or test set are just center cropped to (224,224,3). After that, the cropped image is pre-processed by subtracting the mean and dividing the standard deviation. Finally, the input tensor of shape (9,224,224,3) containing RGB information is generated.

3.2.2 Generate input tensor of flow information

Optical flow can capture the motion of a video, and is also very effective for action recognition on 2D videos. For many action recognition models, taking optical flow feature into account besides RGB information can usually increase the performance.

10 frames are sampled from each video. After performing the cropping and resizing operations to each frame similar to the previous section except for the last pre-processing step, the optical flow is calculated by TV-L1 algorithm[10]. Finally, the input tensor of shape (9,224,224,2) containing flow information is generated.

3.2.3 The architecture of I3D

The architecture of I3D is obtained through [2]. I3D is based on 2D ConvNet inflation: filters and pooling kernels of very deep image classification ConvNets are expanded into 3D, making it possible to learn seamless spatio-temporal feature extractors from video while leveraging successful ImageNet architecture designs and even their parameters[2]. It is especially effective on action recognition of 2D videos: after pre-training on Kinetics[5], I3D models considerably improve upon the state-of-the-art in action classification, reaching 80.9% on HMDB-51[6] and 98.0% on UCF-101[9]. All models but the C3D-like 3D ConvNet use ImageNetpretrained Inception-V1[4] as base network. The Inflated Inception-V1 architecture (left) and its detailed inception submodule (right) is shown in figure 4.

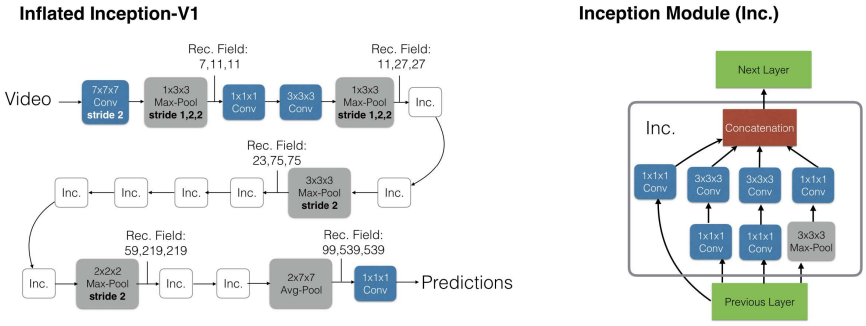

figure 4[2]

3.2.4 Fine tune I3D with input tensor of RGB information

The base model that were fine tuned is the I3D model that pretrained on RGB videos of kinetics dataset. After obtaining the 400-dimension logit outputted by I3D, a dropout layer with 0.36 dropout rate is added to prevent over-fitting, followed by a 49-neuron densely connected layer as the new classifier. The training parameters are: batch size to be 32, optimizer to be SGD,

momentum to be 0.9, initial learning rate to be 1e-3, and decreases every 10000 step. After 50 epochs of training, the model achieved 75.4% validation accuracy over the 49 action classes. The plot of training accuracy vs. validation accuracy is shown in figure 5.

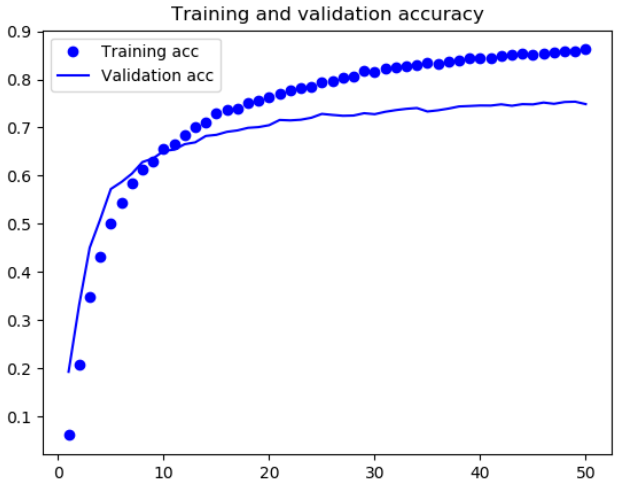

figure 5

3.2.5 Fine tune I3D with input tensor of flow information

The base model that were fine tuned is the I3D model that pretrained on the optical flows of videos in kinetics dataset. The architecture modification and training parameters are the same as the previous section, except that batch size is reduced to 16. After 22 epochs of training, the model achieved 75.9% validation accuracy over the 49 action classes. The plot of training accuracy vs. validation accuracy is shown in figure 6.

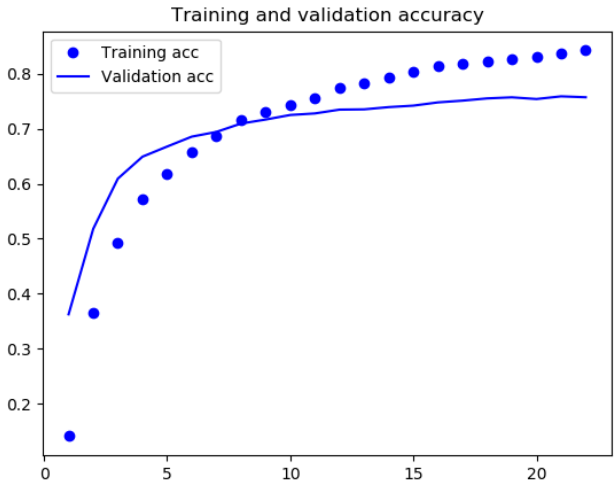

figure 6

3.3 model fusion and performance evaluation

The three models are fused by simply adding up their final decision outputs, which are three 49-dimension probability vectors. The fused model has a validation accuracy of 84.1%, and a test accuracy of 87.7% over 49 action classes, which indicates that these 3 models complement well with each other. For the performance of the combine model on everyday action classes, drop, pick up, sit down, stand up, put on jacket, take of a hat, kick something, jump up, and shake head have test accuracy greater than 98%, with shake head achieved a 100% test accuracy. On the other hand, about the performance of the combine model on actions that indication a person's poor health, staggering, falling down, back pain, and neck pain have test accuracy greater than 90%, with falling down achieved a impressive 100% accuracy. The target action for this report—chest pain have a test accuracy of 83.5%. The code and demo can be found on `https://github.com/Yipeng-LU/ECEN691-2020fall/blob/main/demo_on_dataset.ipynb`.

4 Performance of models on YouTube videos

4.1 Selecting YouTube videos

The three models were evaluated on 13 manually-selected YouTube videos that contain chest pain moments. The total length of these 13 videos are 4942 seconds. The total number of clips generated is 4916 as the models predict a 3-second clip every 1 second. A clip is considered chest pain moment if the subject performs chest pain action within any part of the 3-second period. The

total number of chest pain moments contained in these videos is 347. Although the ratio between number of total moments and number of total clips is as low as 0.07, the ratio will only be lower when checking videos without manual selection.

4.2 performance evaluation of RNN

First of all, the rotation code of the input video is checked, and the rotation is applied to every frame extracted by cv2 library. Note that this check-rotation process is applied to all three models, which are RNN, I3D-RGB, I3D-flow. Next, openpose is used to get the skeleton sequence of the entire video. As the videos in NTU RGB+D dataset are sampled to 24 frames during training, and the average length of videos in NTU RGB+D dataset is about 3s, the skeleton sequence of YouTube video is down-sampled to 8fps. Then, the skeleton sequence is divided into many 24-frame skeleton sequences. The pose feature and motion feature are then generated from these 24-frame skeleton sequences. Then input tensor is obtained after concatenating pose feature and motion feature from each skeleton sequence, and has a shape of (numSkeletonSequences, 23, 975).

After testing on a few YouTube videos that contain chest pain moments, it was observed that the RNN model did not work well on these videos. Only 1 moment was detected from 13 videos containing 347 moments, indicating that the False Negative rate of the RNN model is too high. The poor performance of the RNN model on YouTube videos regardless of its effectiveness on the NTU RGB+D dataset is probabily because the body parts positions outputted by openpose are less accurate, as YouTube videos usually contain multiple people, only show partial body of a person, and have low image quality. Due to the above observations, the RNN model is not adopted in processing YouTube videos.

4.3 performance evaluation of I3D-RGB

After check-rotation process, the YouTube video is down-sampled to 3FPS, as the videos in NTU RGB+D dataset are sampled to 9 frames during training. Then, the full RGB sequence of the video is obtained by center cropping and resizing every frame to (224,224,3). The full RGB sequence is then divided into many 9-frame sequences. The input tensor has a shape of (numRGBSequences, 9, 224, 224, 3). As the input tensor could take up too much memory when the video is long, each RGB sequence is stored in disk, and a data generator is implemented to produce input tensor in a batch size of 32.

After testing on a few YouTube videos, I3D-RGB detected 59 chest moments in total, with 26 moments to be correct and 33 moments to be incorrect. The false positive rate is 0.007, and the false negative rate is 0.925. The code and demo can be found on `https://github.com/Yipeng-LU/ECEN691-2020fall/blob/main/demo_on_youtube_video.ipynb`.

4.4 performance evaluation of I3D-flow

After obtaining the full RGB sequence similar to the last section, it is divided into many 10-frame sequences. Next, the optical flow is generated sequence by sequence using TV-L1 algorithm. The input tensor has a shape of (numRGBSequences, 9, 224, 224, 2). A data generator is also used to generate flow feature.

Because using TV-L1 algorithm will take up too much time when the input video is too long, only 8 of the 13 videos are tested. I3D-flow detected 7 moments in total, with all moments to be incorrect. Due to the poor performance, the I3D-flow is not adopted in processing YouTube videos.

References

[1] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2019.

[2] João Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. *CoRR*, abs/1705.07750, 2017.

[3] Fei Han, Brian Reily, William A. Hoff, and Hao Zhang. Space-time representation of people based on 3d skeletal data: A review. *CoRR*, abs/1601.01006, 2016.

[4] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015.

[5] Will Kay, João Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset. *CoRR*, abs/1705.06950, 2017.

[6] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. Hmdb: A large video database for human motion recognition. In *2011 International Conference on Computer Vision*, pages 2556–2563, 2011.

[7] Huy-Hieu Pham, Louahdi Khoudour, Alain Crouzil, Pablo Zegers, and Sergio A. Velastin. Skeletal movement to color map: A novel representation for 3d action recognition with inception residual networks. *CoRR*, abs/1807.07033, 2018.

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

[9] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *CoRR*, abs/1212.0402, 2012.

[10] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime tv-l¡sup¿1¡/sup¿ optical flow. In *Proceedings of the 29th DAGM Conference on Pattern Recognition*, page 214–223, Berlin, Heidelberg, 2007. Springer-Verlag.

[11] Rui Zhao and Patrick van der Smagt. Two-stream rnn/cnn for action recognition in 3d videos. pages 4260–4267, 09 2017.