H-008

Disentanglement Approach for Video Action Recognition

Lima Sanches Charles†‡ Dang Yaman†‡ Kanemaru Takashi† Nonaka Yuichi† Komatsu Yuto†

1. Introduction

With a total of 770 millions devices sold in 2019 and a constant increasing since 2016*, video surveillance cameras have become the main way to ensure safety in public areas. They are used to detect suspicious behaviors, accidents or persons in danger. However, the manual inspection of footages takes a lot of time. Thanks to the development of deep learning, the automatic processing of the videos is now easier. With a specialized model it is now possible not only to detect and track objects and humans in a video[1][2], but also recognize what action the person is performing with a high accuracy[3].

However, one current limitation of such deep learning models is that their performance drops when the background changes between training data and test data. This is a serious limitation as a new model needs to be built from scratch for each new application even if the task is identical.

Training the models with a larger amount of data including different backgrounds, or using transfer learning to better adapt to new backgrounds are two techniques which have been used to tackle this issue. However, they require large and various datasets[4] and/or sharing the models between applications. This is a serious limitation in business settings as making datasets is costly and it is not always possible to share models between customers.

The recent “disentanglement” technique is a solution to overcome these limitations. A disentanglement model has the ability to separatly encode different factors of the input. It has been used in the image domain to separate writing style from content[5], and in the video domain to separate gait from appearance[6]. Using disentanglement, Denton et al. have shown that it is possible to separate the background from the foreground in the video[7]. They have used the disentangled foreground representation as input for an action recognition classifier. They have shown that the classifier gives better performance when fed with a disentangled representation.

In this paper we focus on Denton’s model, and show that even if it works fine for simple datasets with small background changes, the classifier’s performance drops in case of more complex datasets. We propose two new methods to improve the performance of the orignial classifier on complex datasets. First, we show that by using convolutional layers we can better grasp the temporal relationship between the frames and therefore increase the overall performance by 44.7% compared to the original model. Second, we show that combining disentangled features and features extracted by a model pretrained on a simpler video dataset also improves the performance by 26.5%.

†Hitachi, Ltd. Research & Development Group
‡Contributed equally to the work
*source: Information Handling Services (IHS Markit)

2. Related Work

Since the development of deep learning , several solutions have been proposed to solve the action recognition task on videos with various backgrounds. Most of them are based on Convolutional Neural Networks (CNN) to extract visual features from the frames of the videos[8]. Some approaches combine those features with the optical flow of the video to get the motion information[9]. The temporal information is captured using three dimensional CNN[10] or Long Short Term Memory Networks (LSTM)[11]. Eventhough these approaches give high perfomances, and are robust to background changes, they usually require very large datasets with thousands of videos to work properly[12][13].

Disentanglement is another approach to make models robust to background changes. The idea is to encode separately the background and the foreground information and only feed the foreground information to the network. Since the background information is not accessible for the model, it can only learn features which are independent from the background and therefore it is robust to background changes. Xunyu *et al.* have proposed a method to disentangle foreground from background in videos[14]. However their method is based on the manual labelling of the background and the foreground before training the model, which requires intense manual work. Denton *et al.* have proposed an automatic background disentanglement method for videos. They have not only shown that the method can successfully disentangle background and foreground, but also that the foreground features can be used by a classifier for activity recognition.

In this paper we investigate Denton’s disentanglement method, and show that even if it gives some good results with a simple dataset, the classifier’s performance drops with a more complex dataset. We propose two ways of modifying the classifier part of the original method to enhance the performance on complex datasets.

3. DRNet model and proposed methods

3.1 DRNet model

The original DRNet model proposed by Denton *et al.* is a deep neural netork architecture with two encoders and one

decoder. The “Content Encoder” is trained to take some video frames as input and output a representation vector capturing contents that does not vary over time in the video (typically the background of the video). Similarly, the “Pose Encoder” is trained to output a representation vector capturing contents that varies over time. Finally, the decoder ensures that the original frames can be reconstructed from the vectors produced by the encoders. Details about the training process and the different parts of the model are described in the original paper[7].

In a second step, the Pose Encoder is used for the classification task. As shown in Figure 1, a set of frames is fed to the encoder. For each of the frames, the encoder produces a representation vector which encodes only the foreground information. These vectors are concatenated and used as input for a linear classifier.

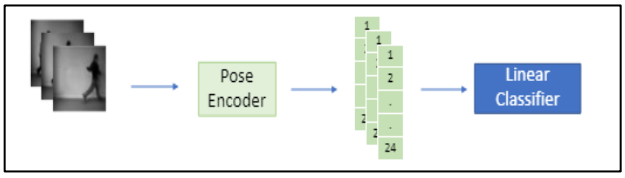

Figure 1: Classification step in the DRNet model. The Pose Encoder takes a set of frames as input and outputs a representation vector for each frame. The classifier uses a concatenation of the vectors as input and outputs the corresponding activity

3.2 Proposed methods

To improve the classification accuracy, we propose two different modifications to the classifier. In both cases, we keep the same Pose Encoder as in the original DRNet paper.

3.2.1 Convolutional classifier

First we investigate the effect of adding convolutional layers before the linear classifier as shown in Figure 2.

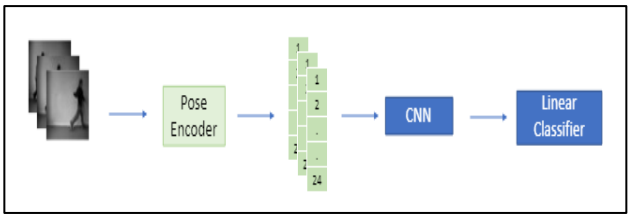

Figure 2: Convolutional classifier. Some convolutional layers are added before the linear classifier to better capture the temporal relationship between the representation vectors

For each frame, the Pose Encoder generates a representation vector capturing the foreground information. All the vectors are then concatenated and fed to some convolutional layers. Then, the output of the convolutional layers is used by the linear classifier to retrieve the activity. The idea is to use the convolution to better capture the temporal relationship between the representation vectors.

3.2.2 Double feature vectors

We also investigate the effect of extracting for each frame another feature vector using a pretrained network, as shown in Figure 3.

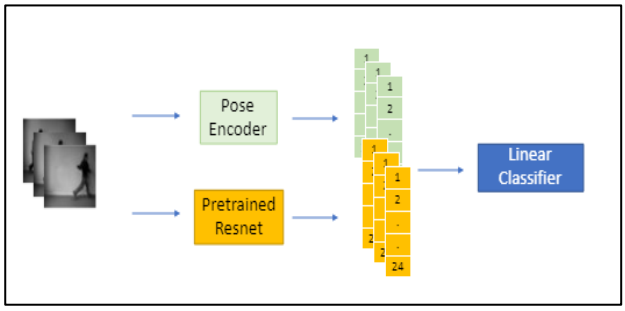

Figure 3: Double feature vectors classifier. The Pose Encoder and the pretrained Resnet take a set of frames as input and output a representation vector for each frame. The classifier uses a concatenation of the vectors as input and outputs the corresponding activity

For each frame, the Pose encoder generates a representation vector capturing the foreground information. Also, a pretrained network is used to generates a feature vector for each frame. Then, we concatenate the representation vectors from the Pose Encoder and the feature vectors from the pretrained network, and use the concatenation as input for the linear classifier. The idea is that using the knowledge from a pretrained network will help the classifier to reach a better accuracy. During the expriment, we use a Resnet model[15] which is a classic model to extract features from images.

4. Datasets

For our experiment, we use two different datasets. Both include people performing some simple activities. The content of each dataset is detailed in this section.

4.1 KTH

The KTH dataset [16] is a dataset of real-world videos of people performing different actions (walking, jogging, running, boxing, handwaving, hand-clapping) as shown in Figure 4.

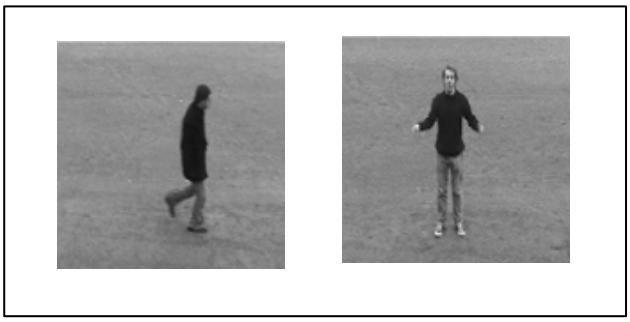

Figure 4: Two frames of the KTH dataset

The dataset includes 2391 videos taken with a static camera with a rate of 25 frames per second (fps). The spatial resolution of the videos is 160 × 120 pixels and the average length of a video is four seconds. There are small variations of camera angles between the videos. The particularity of this dataset is that the different backgrounds are fairly simple and the videos are in grayscale.

4.2 MMAct

The MMAct dataset [17] includes 20 participants performing 37 different activities as shown in Figure 5. Each activity is performed 64 times corresponding to four different backgrounds and four different camera angles (four sessions for each). Among the 37 activities, we only considered the videos of walking, running and handwaving in our experiments. The videos have been captured by a high definition camera, at a resolution of 1920 × 1080 pixels at 30 fps. With its various backgrounds and camera angles, the MMAct dataset is the most challenging of the two datasets of our experiment.

Figure 5: Two frames of the MMAct dataset

5. Experiment

In all of our experiments, we kept the original architectures from Denton *et al.* for the Pose Encoder, the Content Encoder and the Decoder: Resnet-18, VGG[18] and mirrored VGG respectively. For the original approach described in Section 3.1, the encoder part has been trained on the KTH dataset and the recordings of one participant of the MMAct dataset. The classifier part is a three linear layers network with ReLU activations pretrained on the KTH dataset. For the approach described in Section 3.2.1, the CNN is a six layers convolutional network. The encoder part has been trained on the KTH dataset. The classifier part is a three linear layers network pretrained on the KTH dataset. For the approach described in Section 3.2.2, the pretrained network is a Resnet-18 pretrained on the KTH dataset. The encoder has been trained on the KTH dataset and one participant of the MMAct dataset. The classifier is a three linear network pretrained on the KTH dataset. For all approaches, the number of frames used as input for the different models is set to 24.

As a first step, we investigate the performance of the original classifier described in Section 3.1 on the KTH and the MMAct datasets. The results are gathered in Table 1.

Table 1: Classification accuracy of the original model on the KTH and MMAct dataset.

Dataset	Accuracy
KTH	87.4%
MMAct	46.5%

From Table 1 we can see that the classifier reaches a very good accuracy with the KTH dataset. The representation vectors produced by the Pose Encoder contains enough information for the classifier to retrieve the activity. This result confirms the findings of Denton's original paper. However, we can see a performance drop of 40.9% with the MMAct dataset. In this dataset, the backgrounds are more complex and there are different camera angles which make the disentanglement more difficult. Therefore, the vectors produced by the Pose Encoder cannot be used efficiently by the classifier. We have two hypotheses to explain this result. First, the temporal relationship between the representation vectors is not correctly retrieved by the classifier. Second, the features included in the representation vectors are not informative enough to retrieve the activity. To verify these hypotheses we test the approaches described in Sections 3.2.1 and 3.2.2: we use convolutional layers to better grasp the temporal relationship between the vectors, and we use a pretrained network to extract additional relevant features from the frames. The associated results are shown in Table 2.

Table 2: Classification accuracy for the CNN and the Double Feature Vectors approaches on the MMAct dataset

Approach	Accuracy
CNN	91.2%
Double Feature Vectors	73.0%

From Table 2 we can see that using convolutional layers before the linear classifier improves the accuracy significantly by 44.7%. The temporal information in the set of frames is correctly retrieved by the classifier. The remaining error can be explained by the similarity between the "running" and the "walking" activities. In the MMAct dataset, the participants performed the activities inside a small room which forced them to run quite shortly and slowly. Therefore, a set of frames belonging to the running activity can be easily confused with a set of frames belonging to the walking activity.

Table 2 also shows that the double feature vectors approach leads to an improvement of 26.5% compared to the basic approach. In that case, the additional information included in the feature vectors produced by the pretrained Resnet is useful for the classifier to recognize the activity. As for the CNN approach, the origin of the remaining error can be explained by the strong similarity between the walking and the running activities.

Considering the results shown in Table 2, the natural following step would be to combine the CNN and the

double feature vectors approaches to benefit from the advantages of both. However, our preliminary experiments towards this direction have not shown any improvement. The combination of both approaches will be further studied in a future work.

6. Conclusion

In this paper we have investigated an action recognition system based on the disentanglement technique. We have confirmed that the method proposed by Denton *et al.* can efficiently disentangle simple backgrounds and a disentangled feature vectors can be successfully used by a linear classifier to perform activity recognition. We have seen that in case of simple backgrounds the classification accuracy reaches 87.4%, but this performance drops to 46.5% in case of more complex dataset.

We have proposed two different modifications to the classifier part of this model to overcome this limitation: a CNN based approach which allow the system to reach 91.2% accuracy; and a “double feature vector” approach which reaches 73.0% accuracy for the complex dataset. However, we have seen that the models tend to confuse similar activities such as running and walking. We have seen that distinguishing both activities is difficult because of the way the videos have been shot in the MMAct dataset.

In a future work, we will focus on the distinction between similar activities and test our method with a larger number of activities. We will also explore in more details the combination of both CNN and double feature vector approaches. Finally, we will test our approaches on larger and even more challenging datasets such as the Kinetics dataset[12] or the UCF101 dataset[13].

References

[1] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Region-Based Convolutional Networks for Accurate Object Detection and Segmentation," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 1, pp. 142-158, 1 Jan. 2016, doi: 10.1109/TPAMI.2015.2437384

[2] Khan, Gulraiz, Zeeshan Tariq, and Muhammad Usman Ghani Khan. "Multi-Person Tracking Based on Faster R-CNN and Deep Appearance Features." In Visual Object Tracking in the Deep Neural Networks Era. IntechOpen, 2019.

[3] Feichtenhofer, Christoph, Axel Pinz, and Andrew Zisserman. "Convolutional two-stream network fusion for video action recognition." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1933-1941. 2016.

[4] Kuehne, Hilde & Jhuang, Hueihan & Garrote, Estibaliz & Poggio, Tomaso & Serre, Thomas. (2011). HMDB51: A Large Video Database for Human Motion Recognition. Proceedings of the IEEE International Conference on Computer Vision. 2556-2563. 10.1109/ICCV.2011.6126543.

[5] Mathieu, Michael & Zhao, Junbo & Sprechmann, Pablo & Ramesh, Aditya & Lecun, Yann. (2016). Disentangling factors of variation in deep representations using adversarial training.

[6] Zhang, Ziyuan, Luan Tran, Xi Yin, Yousef Atoum, Xiaoming Liu, Jian Wan, and Nanxin Wang. "Gait recognition via disentangled representation learning." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4710-4719. 2019.

[7] Denton, Emily L. "Unsupervised learning of disentangled representations from video." In Advances in neural information processing systems, pp. 4414-4423. 2017.

[8] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar and L. Fei-Fei, "Large-Scale Video Classification with Convolutional Neural Networks," 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, 2014, pp. 1725-1732, doi: 10.1109/CVPR.2014.223.

[9] Simonyan, Karen, and Andrew Zisserman. "Two-stream convolutional networks for action recognition in videos." In Advances in neural information processing systems, pp. 568-576. 2014.

[10] Tran, Du, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. "Learning spatiotemporal features with 3d convolutional networks." In Proceedings of the IEEE international conference on computer vision, pp. 4489-4497. 2015.

[11] Donahue, Jeffrey, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. "Long-term recurrent convolutional networks for visual recognition and description." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2625-2634. 2015.

[12] Kay, Will, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola et al. "The kinetics human action video dataset." arXiv preprint arXiv:1705.06950 (2017).

[13] Soomro, Khurram, Amir Roshan Zamir, and Mubarak Shah. "UCF101: A dataset of 101 human actions classes from videos in the wild." arXiv preprint arXiv:1212.0402 (2012)

[14] Lin, Xunyu, Victor Campos, Xavier Giro-i-Nieto, Jordi Torres, and Cristian Canton Ferrer. "Disentangling motion, foreground and background features in videos." arXiv preprint arXiv:1707.04092 (2017).

[15] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778. 2016.

[16] Schuldt, Christian, Ivan Laptev, and Barbara Caputo. "Recognizing human actions: a local SVM approach." In Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004., vol. 3, pp. 32-36. IEEE, 2004

[17] Kong, Quan, Ziming Wu, Ziwei Deng, Martin Klinkigt, Bin Tong, and Tomokazu Murakami. "MMAct: A Large-Scale Dataset for Cross Modal Human Action Understanding." In Proceedings of the IEEE International Conference on Computer Vision, pp. 8658-8667. 2019.

[18] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).