3D Convolutional Neural Networks for Human Action Recognition

Liangjie Cao

Aug. 1, 2018

1. A 3D CNN Architecture

Based on the 3D convolution described above, a variety of CNN architectures can be devised. In the following, they describe a 3D CNN architecture that we have developed for human action recognition on the TRECVID data set. In this architecture shown in Figure 1, they consider 7 frames of size 6040 centered on the current frame as inputs to the 3D CNN model. They first apply a set of hardwired kernels to generate multiple channels of information from the input frames. This results in 33 feature maps in the second layer in 5 different channels known as gray, gradient-x, gradient-y, optflow-x, and optflow-y. The gray channel contains the gray pixel values of the 7 input frames. The feature maps in the gradient-x and gradient-y channels are obtained by computing gradients along the horizontal and vertical directions, respectively, on each of the 7 input frames, and the optflow-x and optflow-y channels contain the optical flow fields, along the horizontal and vertical directions, respectively, computed from adjacent input frames. This hardwired layer is used to encode our prior knowledge on features, and this scheme usually leads to better performance as compared to random initialization.

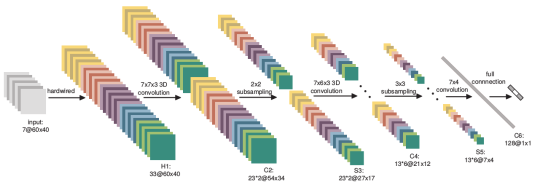

Figure 1. A 3D CNN architecture for human action recognition. This architecture consists of 1 hardwired layer, 3 convo- lution layers, 2 subsampling layers, and 1 full connection layer. Detailed descriptions are given in the text.

They then apply 3D convolutions with a kernel size of $7 \times 7 \times 3$ (7×7 in the spatial dimension and 3 in the temporal dimension) on each of the 5 channels separately. To increase the number of feature maps, two sets of different convolutions are applied at each location, resulting in 2 sets of feature maps in the C2 layer each consisting of 23 feature maps. This layer contains 1,480 trainable parameters. In the subsequent subsampling layer S3, we apply 2×2 subsampling on each of the feature maps in the C2 layer, which leads to the same number of feature maps with reduced spatial resolution. The number of trainable parameters in this layer is 92. The next convolution layer C4 is obtained by applying 3D convolution with a kernel size of $7 \times 6 \times 3$ on each of the 5 channels in the two sets of feature maps separately. To increase the number of feature maps, we apply 3 convolutions with different kernels at each location, leading to 6 distinct sets of feature maps in the C4 layer each containing 13 feature maps. This layer contains 3,810 trainable parameters. The next layer S5 is obtained by applying 3×3 subsampling on each feature maps in the C4 layer, which leads to the same number of feature maps with reduced spatial resolution. The number of trainable parameters in this layer is 156. At this stage, the size of the temporal dimension is already relatively small (3 for gray, gradient-x, gradient-y and 2 for optflow-x and optflow-y), so we perform convolution only in the spatial dimension at this layer. The size of the convolution kernel used is 7×4 so that the sizes of the output feature maps are reduced to 1×1. The C6 layer consists of 128 feature maps of size 1×1, and each of them is connected to all the 78 feature maps in the S5 layer, leading to 289,536 trainable parameters.

2. Experiments

They perform experiments on the TRECVID 2008 data and the KTH data (Schldt *et al.*, [1]) to evaluate the developed 3D CNN model for action recognition.

3. Action Recognition on TRECVID Data

The TRECVID 2008 development data set consists of 49-hour videos captured at the London Gatwick Airport using 5 different cameras with a resolution of 720×576 at 25 fps. The videos recorded by camera number 4 are excluded as few events occurred in this scene. In this experiments, they focus on the recognition of 3 action classes (CellToEar, ObjectPut, and Pointing). Each action is classified in the one-against-rest manner, and a large number of negative samples were generated from actions that are not in these 3

classes. This data set was captured on five days (20071101, 20071106, 20071107, 20071108, and 20071112). The 3D CNN model used in this experiment is as described in Section 2 and Figure 1, and the number of training iterations are tuned on a separate validation set.

To evaluate the effectiveness of the 3D CNN model, they report the results of the frame-based 2D CNN model. In addition, they compare the 3D CNN model with two other baseline methods, which follow the state-of-the- art bag-of-words (BoW) paradigm in which complex handcrafted features are computed. For each image cube as used in 3D CNN, we construct a BoW feature based on dense local invariant features. Then a one-against-all linear SVM is learned for each action class. Specifically, we extract dense SIFT descriptors from raw gray images or motion edge history images (MEHI) (Yang et al., 2009). Local features on raw gray images preserve the appearance information, while MEHI concerns with the shape and motion patterns. These SIFT descriptors are calculated every 6 pixels from 7×7 and 16×16 local image patches in the same cubes as in the 3D CNN model. Then they are softly quantized using a 512-word codebook to build the BoW features. To exploit the spatial layout information, they employ similar approach as the spatial pyramid matching (SPM) to partition the candidate region into 2 × 2 and 3 × 4 cells and concatenate their BoW features. The dimensionality of the entire feature vector is 512×(2×2+3×4) =8192. We denote the method based on gray images as SPM^{cube}_{gray} and the one based on $MEHI$ as SPM^{cube}_{MEHI}.

Figure 2. Sample human detection and tracking results from camera numbers 1, 2, 3, and 5, respectively from left to right.

4. Conclusions and Discussions

They developed a 3D CNN model for action recognition in this paper. 2 This model construct features from both spatial and temporal dimensions by performing 3D convolutions. The developed deep architecture generates multiple channels of information from adjacent input frames and perform convolution and subsampling separately in each channel. The final feature representation is computed by combining information from all channels. They evaluated the 3D CNN model using the TRECVID and the KTH data sets. Results show that the 3D CNN model outperforms compared methods on the TRECVID data, while it achieves competitive performance on the KTH data, demonstrating its superior performance in real-world environments.

References

[1] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: a local svm approach. In *ICPR*, 2004. 1