3D Convolutional Neural Networks for Human Action Recognition

Liangjie Cao

July 30, 2018

Abstract

The authors consider the fully automated recognition of actions in uncontrolled environment. Most existing work relies on domain knowledge to construct complex handcrafted features from inputs. In addition, the environments are usually assumed to be controlled. Convolutional neural networks (CNNs) are a type of deep models that can act directly on the raw inputs, thus automating the process of feature construction. However, such models are currently limited to handle 2D inputs. In this paper, we develop a novel 3D CNN model for action recognition. This model extracts features from both spatial and temporal dimensions by performing 3D convolutions, thereby capturing the motion information encoded in multiple adjacent frames. The developed model generates multiple channels of information from the input frames, and the final feature representation is obtained by combining information from all channels. They apply the developed model to recognize human actions in real-world environment, and it achieves superior performance without relying on handcrafted features.

1. Introduction

Recognizing human actions in real-world environment finds applications in a variety of domains including intelligent video surveillance, customer attributes, and shopping behavior analysis. However, accurate recognition of actions is a highly challenging task due to cluttered backgrounds, occlusions, and viewpoint variations, etc. Therefore, most of the existing approaches (Efros *et al.* [1]; Schldt *et al.* [2]; Dollr *et al.* [3]) make certain assumptions (*e.g.*, small scale and viewpoint changes) about the circumstances under which the video was taken. However, such assumptions seldom hold in real-world environment. In addition, most of these approaches follow the conventional paradigm of pattern recognition, which consists of two steps in which the first step computes complex handcrafted features from raw video frames and the second step learns classifiers based on the obtained features. In real-world scenarios, it is rarely known which features are impor- tant for the task at hand, since the choice of feature is highly problem-dependent. Especially for human action recognition, different action classes may appear dramatically different in terms of their appearances and motion patterns.

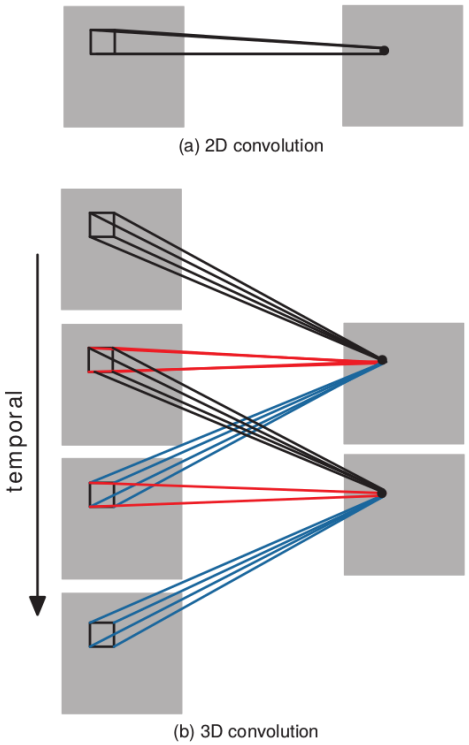

Figure 1. Comparison of 2D (a) and 3D (b) convolutions. In (b) the size of the convolution kernel in the temporal dimension is 3, and the sets of connections are color-coded so that the shared weights are in the same color. In 3D convolution, the same 3D kernel is applied to overlapping 3D cubes in the input video to extract motion features.

As a class of attractive deep models for automated feature construction, CNNs have been primarily applied on 2D images.(Figure 1) In this paper, they consider the use of CNNs for human action recognition in videos. A sim ple approach in this direction is to treat video frames as still images and apply CNNs to recognize actions at the individual

frame level. Indeed, this approach has been used to analyze the videos of developing embryos. However, such approach does not consider the motion information encoded in multiple contiguous frames. To effectively incorporate the motion information in video analysis, we propose to perform 3D convolution in the convolutional layers of CNNs so that discriminative features along both spatial and temporal dimensions are captured. They show that by applying multiple distinct convolutional operations at the same location on the input, multiple types of features can be extracted. Based on the proposed 3D convolution, a variety of 3D CNN architectures can be devised to analyze video data. They develop a 3D CNN architecture that generates multiple channels of information from adjacent video frames and performs convolution and subsampling separately in each channel. The final feature representation is obtained by combining information from all channels. An additional advantage of the CNN-based models is that the recognition phase is very efficient due to their feed-forward nature.

2. 3D Convolutional Neural Networks

In 2D CNNs, 2D convolution is performed at the convolutional layers to extract features from local neighborhood on feature maps in the previous layer. Then an additive bias is applied and the result is passed through a sigmoid function. Formally, the value of unit at position (x, y) in the j^{th} feature map in the i^{th} layer, denoted as v_{ij}^{xy} , is given by:

$$v_{ij}^{xy} = tanh(b_{ij} + \sum_{m} \sum_{p=0}^{P_i-1} \sum_{q=0}^{Q_i-1} w_{ijm}^{pq} v_{(i-1)m}^{(x+p)(y+q)}) \quad (1)$$

where tanh(·) is the hyperbolic tangent function, b_{ij} is the bias for this feature map, m indexes over the set of feature maps in the (i 1)th layer connected to the current feature map, w_{ijk}^{pq} is the value at the position (p, q) of the kernel connected to the kth feature map, and P_i and Q_i are the height and width of the kernel, respectively. In the subsampling layers, the resolution of the feature maps is reduced by pooling over local neighborhood on the feature maps in the previous layer, thereby increasing invariance to distortions on the inputs. A CNN architecture can be constructed by stacking multiple layers of convolution and subsampling in an alternating fashion. The parameters of CNN, such as the bias b_{ij} and the kernel weight w_{ijk}^{pq}, are usually trained using either supervised or unsupervised approaches.

Note that a 3D convolutional kernel can only extract one type of features from the frame cube, since the kernel weights are replicated across the entire cube. A general design principle of CNNs is that the number of feature maps should be increased in late layers by generating multiple types of features from the same set of lower-level feature maps. Similar to the case of 2D convolution, this can be achieved by applying multiple 3D convolutions with distinct kernels to the same location in the previous layer (Figure 2).

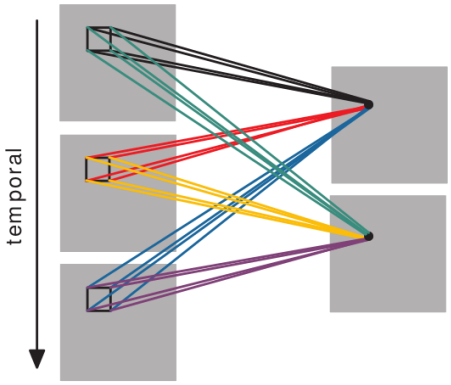

Figure 2. Extraction of multiple features from contiguous frames. Multiple 3D convolutions can be applied to con- tiguous frames to extract multiple features. As in Figure 1, the sets of connections are color-coded so that the shared weights are in the same color. Note that all the 6 sets of connections do not share weights, resulting in two different feature maps on the right.

References

[1] H. Jhuang, T. Serre, L. Wolf, and T. Poggio. A biologically inspired system for action recognition. In *ICCV*, 2007. 1

[2] Y. LeCun, F. J. Huang, and L. Bottou. Learning methods for generic object recognition with invariance to pose and lighting. In *CVPR*, 2004. 1

[3] F. Ning, D. Delhomme, Y. LeCun, F. Piano, L. Bottou, and P. E. Barbano. Toward automatic phenotyping of developing embryos from videos. *TIP*, 2005. 1