Action Recognition in Videos using Two-Stream Convolutional Networks

Saurabh Kumar
2015088

Kaustav Vats
2016048

Manish Mahalwal
2016054

I. Problem Statement

We propose a two-stream approach for identification of actions in a video clip by analysing the still frames using the contextual information. We use optical flow for capturing contextual information about motion across still frames of a video. We will use two-stream approach to incorporate the spatial and temporal components across consecutive frames.

II. Literature Review

Action recognition task involves the identification of different actions from videos where the action may or may not be performed throughout the entire duration of the video.
Classifying actions (jumping, picking, playing etc) in still images has been solved with a high accuracy using Deep Convolutional Networks. However, classifying actions in videos is challenging when compared to action recognition in images. Action recognition from videos requires capturing context from the whole video rather than just capturing information from each frame as compared to an image. We aim to discover the principles to design effective ConvNet architecture for action recognition in videos and learn these models given limited training samples.
We aim to extend the existing model of Deep Convolutional Neural Network used for classifying action in still images. Relevant work already exists for the same which uses stacked frames for input to CNN, but the results are poor. So, to improve upon this architecture, we implement a new architecture for action recognition in videos which is based on the human visual system.
This architecture is based on the human visual cortex system which uses two-streams to detect and recognise motion:
1. Ventral Stream: used for performing object recognition
2. Dorsal Stream: used for recognising motion The author's architecture involves two separate recognition streams - spatial and temporal. Spatial Stream performs action recognition in still frames which extracted from the video. Temporal Stream recognises action from motion using dense optical flow.

A. Two-stream architecture for action recognition

We can decompose a video into two components i.e. spatial and temporal. Each of the two streams are implemented using deep CNN. [2]

1) Spatial Stream ConvNet: Spatial Stream carries information about the objects and scene in the video. This network operates on individual frames extracted from the video at some sampling rate.
2) Temporal Stream ConvNet: Temporal Stream carries information about the movement of the of the object(or the camera) in the scene (refer Fig. 2) i.e. the direction of the object.
   The input fed to this model, is formed by stacking multiple optical flow displacement images between consecutive video frames. This makes the task of recognition of motion easier as the network doesn't have to learn about the motion on its own. To represent motion across several frames, displacement vector of the frames are stacked together. An alternative representation of the motion is trajectory of the object. The flow is sampled along the motion trajectories. [3]

(a) Video frame (b) Vertical component of optical flow (c) Horizontal component of optical flow

Fig. 1: Example of extracted optical flow across a pair of consecutive frames

III. Dataset

A. Description of the dataset

We are using the famous dataset called UCF101 for our project. UCF101 is a dataset of human actions. Its and extended version of UCF50 which had 50 classes. UCF101 have 101 classes which are divided into five types:

1) Human-Object Interaction
2) Body-Motion Only
3) Human-Human Interaction
4) Playing Musical Instruments
5) Sports

The dataset consists of realistic user-uploaded videos containing cluttered background and camera motion. It gives the largest diversity in terms of actions and with the presence of large variations in camera motion, cluttered background,

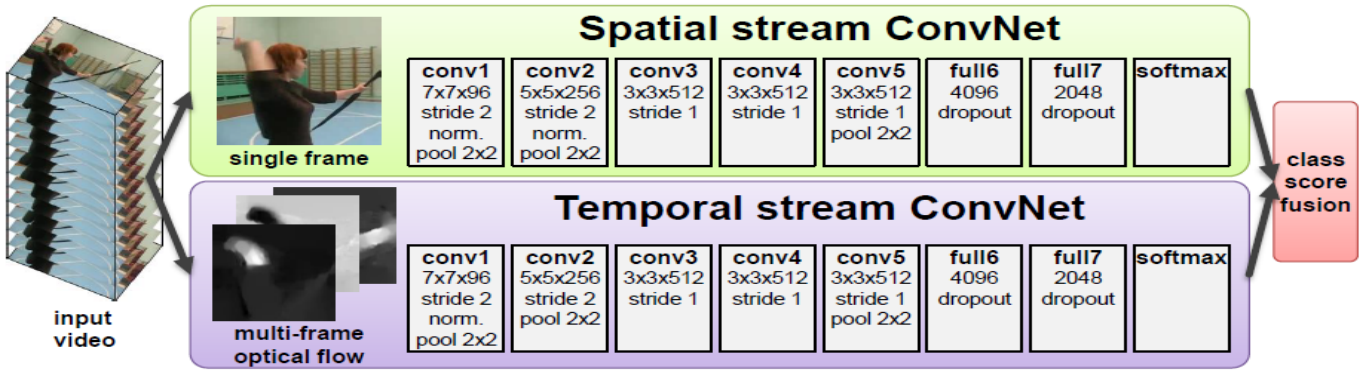

Fig. 2: Two stream ConvNet architecture.

illumination conditions.
But Since the dataset is extremely large(60GB+) and we dont have enough computation power, We decided to extract 10 classes from the dataset (around 20GB) for our project purpose.

IV. Implementation Details

Until now, we have implemented spatial ConvNets by fine-tuning Resnet34.

A. Preprocessing

To extract frames from the videos we sample the video at a rate of 10 and save the files as .jpg. This costs about 5.9GB for due to a large number of frames being extracted from the video. However, we have to bear this cost since we require these images to detect the motion in the video. For computing the flow of consecutive frames we used cv2 toolbox. We extract RGB frames from each video in UCF101 dataset with sampling rate: 10 and save as .jpg image in disk which cost about 5.9GB.

B. Training

The training process is same for both temporal and spatial networks. The only difference is in the input fed to the two networks, spatial takes in RGB frames while temporal takes in optical flow of the consective frames. The weights are learnt using stochastic gradient descent. For training, a mini-batch is taken. A sub-image of size 224x224 is randomly cropped from a frame randomly selected from the mini batch. The videos are rescaled to a common resolution so the all extracted frames are of same dimensions. A learning rate of 0.01 is set and is reduced after a fixed number of iterations.

C. Testing

For testing, when the model is fed a video, we sample frames RBG frames from it at equally spaced intervals. These frames are stacked together and fed to the ConvNet. The model returns the class scores for the input, from which the we take the argmax of the scores. The class scores for the whole video are computed by averaging the scores across the frames.

D. Evaluation

We evaluate our model on UCF101 benchmark. This dataset provides train-test splits by default. The spatial ConvNet was tested on the validation set of train and testing split. To measure the performance we used top-5 and top-1 accuracy.

V. Results

Since the dataset was huge we had to train our model for only few epochs with 53 videos (35 Training and 18 testing) from 10 classes, after receiving the predicted probabilities from the model. We calculated top 1 and top 5 score of our model. In top 1 score, we checked if the top class with highest probability is same as the target label. For top 5 score, we checked if the target label is one of the top 5 predictions. For both cases top scores are computed as times a predicted label matched the target label, divided by the number of data-points evaluated. Using, optical flow leads to better accuracy than existing model which directly use the stacked RGB frames. However, it leads to huge pre-computation costs and requires large disk space. Refer Table I

	Training	Validation
Loss	1.63	26.89
Top 1 Accuracy	37.14	11.11
Top 5 Accuracy	85.71	61.11

TABLE I: Accuracies on training and validation set

VI. Milestones

We have implemented spatial CNN for the mid-review using ResNet. We aim to implement Motion for the final evaluation. This will be followed by combining the results of the two CNN using average fusion technique. We will try results with different datasets.

References

[1] Khurram Soomro, Amir Roshan Zamir and Mubarak Shah "UCF101: A Dataset of 101 Human Action Classes From Videos in The Wild."
[2] Karen, Az "Two-Stream Convolutional Networks for Action Recognition in Videos", arXiv:1406.2199
[3] Wang L. et al. (2016) Temporal Segment Networks: Towards Good Practices for Deep Action Recognition. In: Leibe B., Matas J., Sebe N., Welling M. (eds) Computer Vision ECCV 2016. ECCV 2016. Lecture Notes in Computer Science, vol 9912. Springer, Cham