International Journal of Advanced Science and Technology
Vol. 29, No. 9s, (2020), pp. 6465-6471

Identifying Criminal Activites in Video Footages using Deep Learning Methods

Utkarsh Paliwal, S.Iniyan, Dr R. Jebakumar
[1] *Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur, Chennai, India*
[2] *Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur, Chennai, India* [3] *Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur, Chennai, India*
E-mail: up9222@srmist.edu.in, iniyans@srmist.edu.in , jebakumr@srmist.edu.in
[*]*Corresponding Author: up9222@srmist.edu.in*

Abstract

Identifying Criminal Activities from video footages is the process of getting high level features from a set of training data and analyzing it to understand the related features which contribute to a video being labelled as "Criminal" or "Normal". We used image processing techniques to extract features from the related videos and the machine was able to learn the features to able to correctly classify the given video as "Criminal" or "Normal". The proposed system uses the following tools – a network of Convolutional Neural Network layers amalgamated with Artificial neural network which uses backpropagation to converge to a minimum loss.

Keywords *– Convolutional Neural Networks, Deep Learning, Artificial Neural Networks, Image processing, Video Analysis, Machine Learning.*

I. INTRODUCTION

With the increase in the number of cameras around us the video footages are increasing and the task of monitoring those videos is increasing with it. The growth is exponential and also needs to be taken care of as careful analysis and monitoring of the videos can help the government agencies and various other authorities to be aware of the fact about what is going on under their jurisdiction. Video analysis is one of the major breakthroughs in technologies wherein we can instruct the machine learning models to be able to learn and identify various spatial and temporal features necessary for classification of videos. The proposed methodology is limited not only to the domain of criminal activity but can also be applied to various other video classification problems.

II. LITERATURE SURVEY

There is a lot of research work done earlier in the field of video classification. The works that we came across encompassed a various range of methodologies. A video can be classified on the basis of the spatial activities occurring inside the video footage and various other parameters. The earlier works have focussed on an extensive range of algorithms ranging from Pose Detection to Tube Extraction. The works include techniques that are discussed further in this paper. We studied various works and most of them focussed on image processing and converting the videos frame by frame and feeding them to the

prediction algorithm. Federico Landi, in his paper "Anomaly locality in video surveillance" talks about tube extraction process which is the manually feeding the spatiotemporal coordinates of the activity taking place inside the full frame videos. The model is then able to better identify the coordinates in the video where a criminal activity is occurring. The coordinates are provided by the researchers themselves and the annotations can be downloaded and used. Their work uses regression network for predictions. X. Wang, and X. Tang in their paper "Hybrid Deep Learning for Face Verification" propose the usage of a Hybrid Deep Learning algorithm which is generally used for facial features extraction. They used the Hybrid Deep Learning Algorithm for video analysis and achieved quite an improvement in the metrics. Their methodology was good and can be used to improve upon the currently existing systems. Their works argues that their methodology improves upon the already existing techniques. S. Chackravarthy, in the paper "Intelligent Crime Anomaly Detection in Smart Cities using Deep Learning" used the earlier described Hybrid Deep Learning algorithm for identifying criminal activities in video sequences. Le, Quoc V. in "A Tutorial on Deep Learning Part 2: Autoencoders, Convolutional Neural Networks and Recurrent Neural Networks" talks about the usage of auto encoders and propose an unsupervised learning approach to the given problem.

A. Thyagarajmurthy in his work "Anomaly Detection in Surveillance Video Using Pose Estimation" talks about an entirely new methodology of Pose Estimation for classification of videos. This method proves to be a good classifier for certain set of videos but the main problem with the model is that videos may contain criminal activities which may or may not involve the presence of a human. Events like accident, bomb blast, fire etc. This leads to a need for robust methods like the ones which are able to identify features in a video to classify videos.

III. DATA DESCRIPTION

A. Data Acquisition

The dataset was provided by the University of California Florida as UCF Crime Dataset. The dataset consisted of 96GBs of video footages captured mainly from surveillance cameras. We use this data itself for classification purposes. The dataset consists of over 13 classes of criminal activities and a set of normal videos. The dataset is already well structured into folders of appropriate activity, i.e., each of the thirteen folders contains criminal activity of the type of the name of the folder. The videos provided by UCF had a frame rate of 30 fps. Various criminal activity classes include classes such as "Abuse", "Assault", "Arrest", "Explosion", "Burglary" etc. Data was collected by UCF and is publicly available for download on their website. The dataset can also be downloaded from dropbox in parts. We downloaded this data and used it for our work discussed in this paper. The average length of the videos in total was around 40 seconds. The total length of all the videos combined was around 180 minutes and the total number of videos was 300. The data distribution into training and test sets can be chosen by anyone as per the requirements. A general division of about 80:20 where 80 percent of the videos in the training set and 20 percent of the videos in the test set would suffice. But other divisions such as 70:30 or 75:25 or 60:40 would need some experimentation. Snapshots from two types of activities and their classes are shown ahead. A short summary about the dataset is given in the table below:

Provider	Average Length (s)	Total Length (min)	Number of videos
UCF	40 sec	180 min	300

Table 1: Dataset Description

Fig.1: Arson

Fig.2: Abuse

B. Data Cleaning and pre-processing

The second stage is data cleaning. The videos are converted from video format to a collection of images/frames. The processed data is prepared by adding important videos from the dataset, converting them to numpy arrays of suitable sizes for the neural network. The dataset is converted into frames of suitable sizes and then the data is normalized for better results. The dataset provided by the University of California Florida was split into various folders each containing a specific type of activity only. This folder name was used as the dependent variable for the network. The dependent variable is extracted from the folder name of the criminal activity. Any regex or a user defined function could read the names of all the folders to extract the names of the folder and feed it to make the dependent variable of the dataset.

C. Feature Extraction

The third stage talks about feature extraction. We used Convolutional Neural Network for high level feature extraction from the video frames. A Convolutional Neural Network (ConvNet/CNN) is a type Deep Learning algorithm which generally takes in the input as an image, is able to assign learnt weights and other biases to various features inside the image and correctly distinguish one feature from other type of feature. The better aspect about CNNs is that the CNNs require comparatively lesser preprocessing as compared to other algorithms dealing with the problems of video classification. Without the use of Deep Learning algorithms, the rules for identifying features in the video footages have to be manually fed in. Whereas, with the usage of a ConvNet based approach, the CNNs learn and assign importance to the features themselves via training. The design structure of a ConvNet is based on the network of Neurons inside a human brain. This ConvNet is assisted by an Artificial Neural Network (ANN).

IV. EXPERIMENTAL RESULT AND EVALUATION

The classification methodology implemented in our project comprises of various aspects. They include the following three points

A. Video Analysis

First stage of our implementation involves finding out the features in an image which contribute the maximum towards a frame of a video being classified as "Criminal" or "Normal". The task of identifying which feature or pattern in a video can help in predicting the class of a video is to be accomplished. This task was accomplished with the usage of the before described Convolution Neural Network (CNN). The Convolution on the image is done in the Convolution Layer. The CNN network is described in the upcoming subsection below.

B. Convolution Layer

Convolutional Layer consists of neurons which take input as the image in the data set. Convolutional layer is nothing but a set of neurons that are capable of identifying various features in an image. These convolutional neural networks are very efficiently able to identify patterns in an image that contribute to better predictions. The convolutional layer also needs a specific input about the size of the kernel or the sliding window. CNNs use a small matrix containing random weights which are updated as the network trains. The matrix is just like a sliding window that "slides" over the entire image looking for features and keeps calculating the matches with random weights that are set for the feature map at that current epoch. Feature maps are a set of maps that contains weights that are learnt by the CNN to classify an image properly. The convolutional layer generates a set of feature maps and feeds it forward to the layers ahead.

The convolution function used can be represented as:

$$G[m,n] = (f*h)[m,n] \sum_j \sum_k h[j,k[f[m-j, n-k]$$

Where-

- The input image is represented by f and the kernel is represented by h.
- The resultant matrix's index of row and column index are represented by m and n respectively.

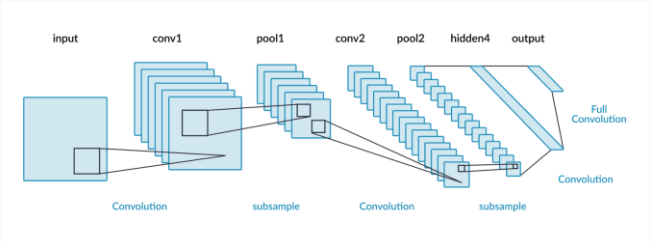

Fig.3: Sample Architecture of a Convolutional Neural Network

(Credits: https://missinglink.ai/wp-content/uploads/2019/08/LeNet-5-1998.png)

C. Pooling Layer

Pooling layer like the previous layers comprises a sliding window which does the pooling operation in each window of predefined size. The sliding window again similar to the previous layers is a matrix which does a computation and stores the output in a separate pooled feature map. This is done so as to retain the spatiotemporal features in an image and make our model more robust to newer inputs. For example, a robust model should be able to recognise a particular object even if the image is slightly rotated. The pooling layer not only helps in recognising the slightly rotated images but also mirror images sometimes. The pooling layer we used was in there for the abovementioned purpose itself. It helps in making our model robust and better classifier with respect to newer inputs that the model would never have seen during the training process. The pooling can of various types like Max Pooling or Mean Pooling etc.
Max Pooling calculates the maximum value in a window and stores it in the pooled feature map. Similar is the case with Mean pooling which calculates the mean of all the values. Any of such operations can be selected to be used as a pooling layer.

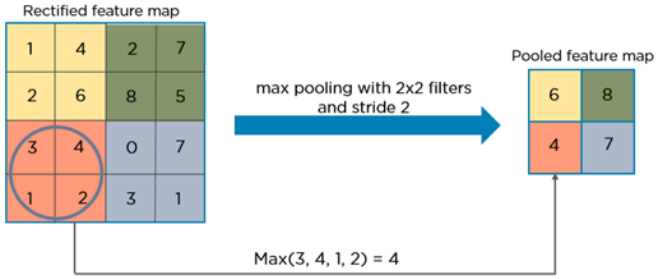

Fig.4: Max Pooling Operation Representation

(Credits: https://qph.fs.quoracdn.net/main-qimg-95aaf9084b7f1fe5eec3f37b1d8b89d3)

D. Flattening

The final pooled feature maps are then flattened into a single linear numpy array. This is done so as to feed to the final Artificial Neural Network which then detects patterns in the array and therefore in the image. The artificial neural networks are designed to operate on a long array of input features. An image as we know is a a matrix of pixel values. The matrix obtained after various operations is also 2D in shape. These needs to be converted to a long linear array. This is why flattening is important.

IV. CONCLUSION

To summarize, the input videos are converted into frames and fed to the network for training. The model trains on these videos frame by frame and learns using backpropagation on these values. The model learns from the huge dataset available and trains on the data. Convolutional Neural Networks are a powerful tool to identify patterns in image dataset. This is why CNNs are so widely used in image processing problems. They are able to efficiently identify features in an image. If ConvNets are not used, we might need to manually need to feed in programmed rules which could be used for feature identification. But then the model would not be robust enough. The accuracy would not be impressive. Here is where ConvNets come in.

CNN is able to map various features necessary for classifying the input videos. It identifies patterns using the process described earlier. It builds various feature maps. Pooling Layer next to the convolutional layer takes the max of each sliding window in the image and stores it into a set of new pooled feature maps. The newer built pooled feature maps are then flattened out and fed to the network for training.

The model is successfully able to classify a new video as "Criminal" or "Normal". The use of Convolutional neural networks with Artificial neural network were

V. FUTURE WORKS

Our work deals with the basic CNN plus ANN approach where ANN is Artificial Neural Network. There are a bunch of technologies that could be used to take our work further and gain different results. This work can further be extended to use the various algorithms such as Hybrid Deep Learning, Pose Detection, Long Short-Term Memory etc. Various models should be researched and each model should be compared and tried for results. A better testing and training accuracy is anticipated by the use of such training methods/techniques.

REFERENCES

1. S. Lawrence, C. L. Giles, Ah Chung Tsoi and A. D. Back, "Face recognition: a convolutional neural-network approach," in IEEE Transactions on Neural Networks, vol. 8, no. 1, pp. 98-113, Jan. 1997.
2. Xin Yao, "Evolving artificial neural networks," in Proceedings of the IEEE, vol. 87, no. 9, pp. 1423-1447, Sept. 1999.
3. Federico Landi, Cees G. M. Snoek, Rita Cucchiara1, ``Anomaly locality in video surveillance," arXiv:1901.10364v1 [cs.CV] 29 Jan 2019.
4. Sun, Y., X. Wang, and X. Tang. "Hybrid Deep Learning for Face Verification." IEEE Transactions on Pattern Analysis and Machine Intelligence. U.S. National Library of Medicine, Oct. 2016. Web. 19 June 2017.
5. Le, Quoc V. "A Tutorial on Deep Learning Part 2: Autoencoders, Convolutional Neural Networks and Recurrent Neural Networks." Semantic Scholar. N.p., n.d. Web. 19 June 2017.
6. A. Thyagarajmurthy, M. G. Ninad, B. G. Rakesh, S. Niranjan, B. Manvi, "Anomaly Detection in Surveillance Video Using Pose Estimation"
7. Mahmudul Hasan, Jonghyun Coi, Jan Neumann, Amit K Roy-Chowdhury, and Larry S Davis, "Learning temporal regularity in video sequences", in CVPR, 2016.
8. S. Chackravarthy, S. Schmitt , L. Yang, and Y. Tagawa, "Intelligent Crime Anomaly Detection in Smart Cities using Deep Learning," 2018 IEEE 4th International Conference on Collaboration and Internet Computing
9. Dan Xu, Elisa Ricci, Yan Yan, Jingkuan Song, and Nicu Sebe,"Learning deep representations of appearance and motion for anomalous event detection",in BMVC, 2015.
10. Haibing Wu and Xiaodong Gu, "Max-Pooling Dropout for Regularization of Convolutional Neural Networks", https://arxiv.org/ftp/arxiv/papers/1512/1512.01400.pdf
11. M. Ahmadi, S. Vakili, J. M. P. Langlois and W. Gross, "Power Reduction in CNN Pooling Layers with a Preliminary Partial Computation Strategy," 2018 16th IEEE International New Circuits and Systems Conference (NEWCAS), Montreal, QC, 2018, pp. 125-129.

12. Yu, D., Wang, H., Chen, P. and Wei, Z., 2014, October. Mixed pooling for convolutional neural networks. In International conference on rough sets and knowledge technology (pp. 364-375). Springer, Cham.