International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 7 July 2021, pp: 2776-2780 www.ijaem.net ISSN: 2395-5252

Violence Detection Using Deep Learning

[1] Mr. Vijaykumar Dudhanikar, [2] Mr. Pratheeksh, [3] Mr. Shetty Amar Ravindra, [4] Mr. Shodhan, [5] Mr. Shreesha Poojary.

[1] Assistant Professor, Dept. of CSE, Yenepoya Institute of Technology, Moodbidri, India-574227
[2,3,4,5] Students, Dept. of CSE, Yenepoya Institute of Technology, Moodbidri, India-574225

Submitted: 05-07-2021 Revised: 19-07-2021 Accepted: 22-07-2021

ABSTRACT - Over the previous couple of decades, remarkable infrastructure growths are noticed in security related issues throughout the planet . So, with increased demand for Security, video-based Surveillance has become a crucial area for the research. An Intelligent Video Surveillance system basically censored the performance, happenings, or changing information usually in terms of human beings, vehicles or any other objects from a distance by means of some electronic equipment (usually digital camera). In broad terms, advanced video-based surveillance could be described as an intelligent video processing technique designed to assist security personnel's by providing reliable real-time alerts and to support efficient video analysis for forensic investigations. Different modeling schemes are used for designing of efficient surveillance system under various illumination conditions. Further move surveillance camera will be used for crime detection and this can be implemented using deep learning technique. This system mainly aims at alerting the admin in case of crime detection through CCTV.

Key words-CNN, RNN**,** Deep learning, Spatio temporal encoder, 3D-CNN, LSTM, ROI, ODTS.

I. INTRODUCTION

In video surveillance, to critically assure public safety hundreds and thousands of surveillance cameras are deployed within cities, but it is almost impossible now a day to manually monitor all cameras to keep an eye on violent activities. Rather, there is a significant requirement for developing automated video surveillance systems to automatically track and monitor such activities. Thereby, in case of emergencies alarming the controlling authorities to take appropriate measures against detected violence. Violence recognition is a key step towards developing automated security surveillance systems, to distinguish normal human activities from abnormal/violent actions. Normal human activities are often categorized as routine life interactive behaviors, such as walking, jogging, running, hand waving. However, violence is subjected to unusual furious actions, such as fight activity happening between two or more people. In last few years, the task of human action recognition has received much attention of the researcher community, to detect normal day human activities through video analysis, see surveys.In this project we created two datasets specifically for fight activities detections, to distinguish violent/fight incidents from normal events. Historically, human activity recognition is achieved through traditional hand-crafted feature representation approaches such as Histogram of Oriented Gradient (HOG), Hessian3D and Local Binary Pattern (LBP) etc. More on, there is growing tendency to solve this problem by adopting learning based deep representation techniques, such as Convolutional Neural Networks (CNN), 3D-CNN for Spatio-temporal analysis, CNN followed by Recurrent Neural Network (RNN) and Spiking Neural Networks (SNN) etc.

II. DEEP LEARNING-BASED OBJECTDETECTION AND TRACKING SYSTEM

A. Concept

Figure 1 shows the method of object detection and tracking by the ODTS over time [7]. The corresponding type or class of each detected object is simultaneously classified by the object detection module. Then, supported the detected object information, a dependent object tracking module is initiated to assign the unique ID number to every of the detected objects, and predict the next position of each of the objects. But if past tracked BBox is 0, the amount of tracking BBox equals to the amount of the detected objects. For example, in time T+c, if u is 0, u' equals to n'. In other words, when the past tracking BBox did not exist, the current tracking BBox takes from the detected objects per each class. This object tracking module was composed by introducing an object tracking algorithm called SORT algorithm [5],Then, IOU of all the possible pairs between the predicted positions, ′ at time T and detected objects positions, at the time T+c are calculated. The nearest objects, namely the pair with the very best IOU value, are going to be assumed an equivalent object with an equivalent ID.

Similarly, any object in : which has no object pair with above IOU value of 0.3 are going to be considered as newly appeared into the RoI at T+c. The freshly emerged object are going to be assigned by a replacement ID number not overlapped with the previous ID number.

This system utilizes a faster RCNN learning algorithm [5] for object detection and a kind [6] for ID assignment and object tracking.These system processes object tracking using SORT [6] algorithm supported IoU value, therefore the object tracking ability was suffering from video frame interval c [7]. Video frame interval can reduce the computation amount over time by adjusting the detection interval of the thing detection network. To check this, object tracking ability over the frame interval experimented, then it had been possible to trace the objects until six frame intervals [7]. Increasing frame interval significantly reduces object tracking ability, in order that the video frame interval should be optimized for the amount of camera devices simultaneously connected to a deep-learning server.

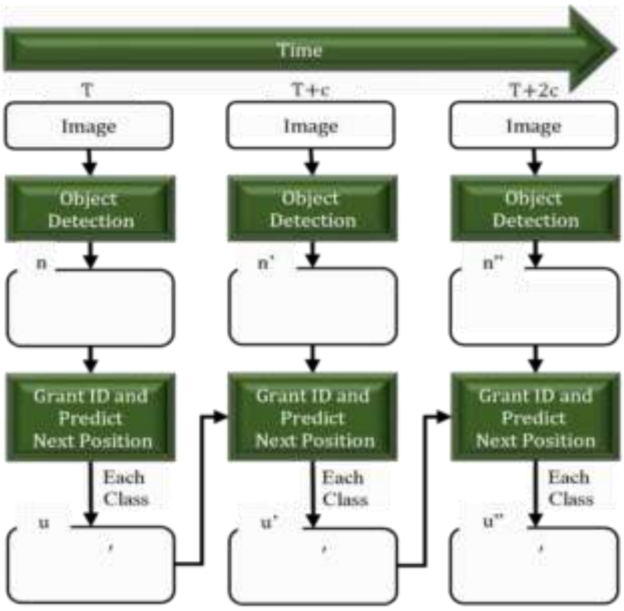

Figure 1. Object detection and tracking process of Object Detection-Tracking system over time. When class and BBox were obtained by object detection, object tracking algorithm grants ID and predicts next position using current and past BBox.

B. Violence detection system

In machine learning domain, Feature learning is very appealing due to learning complex underlying data representation, especially for complex task of image recognition, as compare to hand-crafted feature descriptors. The learnt features acquired through learning a specific problem, can be reutilized for solving another problem in a new task, a concept known as transfer learning. This approach has been successfully used in object classification and categorization domain. The figure 3.1 below represents CNN deep model which is originally data driven, it requires large labeled dataset for training. Annotated dataset preparation is complex and demanding task. On the other hand, providing insufficient amount of data would not leverage CNN model to learn optimal deep features instead network suffer from significant overfitting issue. To solve the problem of overfitting for small dataset, utilizing modern deep learning network architectures, the approach of transfer learning comes into play. In which, existing network architecture with pre-trained learned features as source task network is employed to build new target task network architecture for limited dataset shows general representation of source task network, with convolutional blocks followed by dense fully connected subsequent layers, pre-trained on ImageNet with 1000 output classes. The source task network is utilized for transfer learning to create a target task network, to be trained on dataset with 2 output classes for violent/fight and non-fight activities.

C. System design

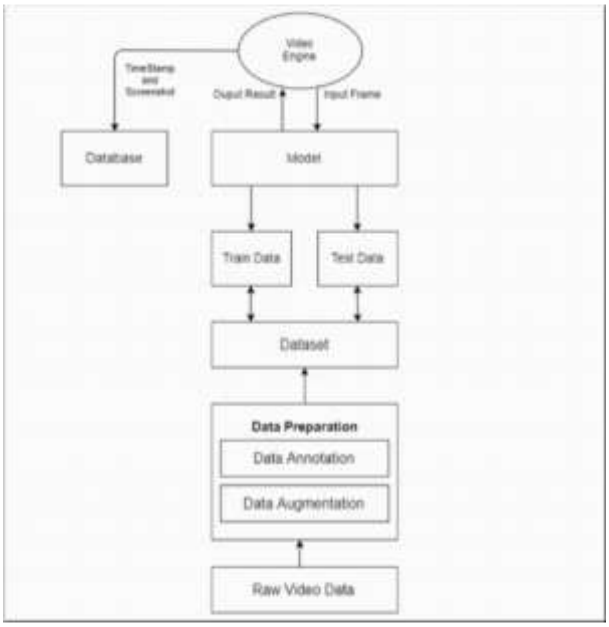

Fig 2. System Design

The figure 2 above represents the event flow diagram of the system. System architecture contains 5 modules namely Data preparation module, dataset, Deep Leaning model, video engine and database. This system is implemented in python and TensorFlow as a backend. User gives video file as an input and system gives output as video is violent or nonviolent. System supports .mp4 and .avi video formats.

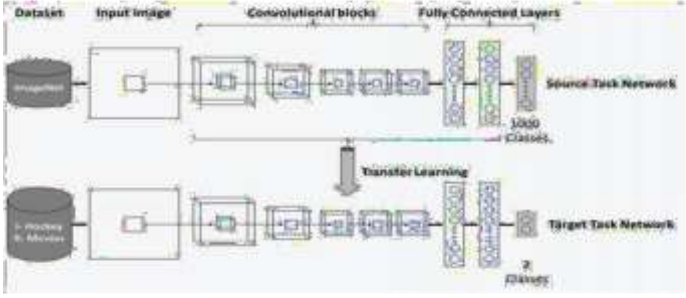

Fig 3. Network Architecture of System

Modules: This system is divided into five parts according to functions performed by individuals.

Data Preparation: This module deals with raw video data. It consists of two submodules Data Augmentation and Data Annotation.

Data Augmentation: It is a method of augmenting the available data. Main purpose of augmentation is to increase the size of available dataset.

Data Annotation: System is based of supervised learning so annotation is an important module which labels the data.

Dataset: Dataset consist of data prepared by data preparation module. Dataset is further split into training and testing.

Deep Learning Model: This is the deep learning model trained using input dataset. This model will be invoked by video engine and model will classify input as violent or nonviolent.

Video Engine: This module is an interface between user and deep learning model. The video engine will take the input from user and will pass it through the DL model. It has feature of alerting govt authorities if any suspicious activity is detected.

Database: This database contains timestamp and screenshot of suspicious activities identified by system.

III. EXPERIMENTS

A. Deep learning training

The models have been trained on the training portion of the data set and after that have been tested against the videos which were reserved for the testing purpose. Table III shows the performance of the CNN model after 10, 20, 50, 100, 200 and 500 epochs in terms of percentage accuracy.

We see that after 50 epochs the accuracy is not converging instead decreasing to some margin.The graph shows that the model is guilty of overfitting to some scale and the accuracy is fluctuating in the range of 76% to 96%. Fig 4 is the epoch vs loss graph of the CNN model which shows us the fluctuating loss of test set.

However, we have used the pre-trained model to leverage transfer learning in two different approaches.

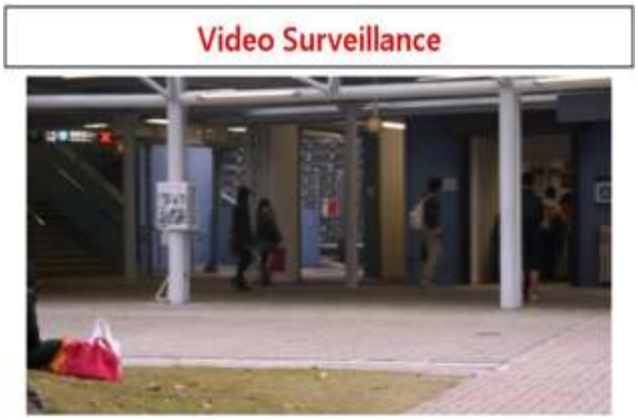

Fig 4. Non-violence Video as input

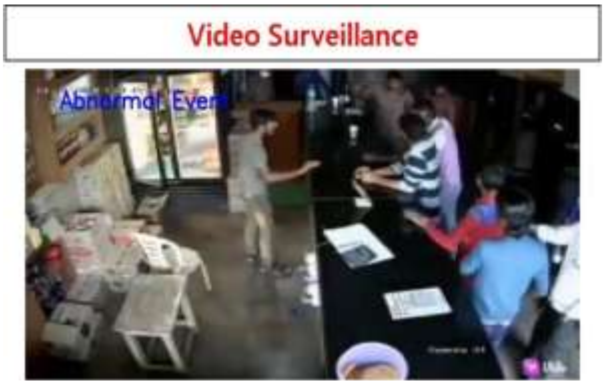

Fig 5. Violence video as Input

The CNN model is trained with the non-violent video as the input in the train.py program. While training the videos are broken down into frames and then each frame is passed through the CNN model and the frames gets filtered. These CNN layers then generate the arrays based on the shapes and the figures in the frames and these values are stored in binary format in the file named training.npy.

This training.npy file is then passed to the test.py file. With the use of Flask framework we built the webpage for the final output of the program. The program breaks the input videos into frames and then processes it by comparing with the trained model data if the value of the loss is greater than 0.00068 then the event can be classified as an abnormal event and thus the respective frame which satisfies this condition is labeled as abnormal event and a warning sound is generated by the system to alert the person in charge of the surveillance. Further an email is sent to the person in charge of the surveillance with time, date specified informing him/her about the abnormal event that occurred at what time and in which day. The following snapshots shows the results generated by the program.

Approach	Training	Testing
LSTM	87.23	67.41
CNN+LSTM	89.79	75.10

Table 1Performance of the LSTM and CNN+LSTM model

IV. CONCLUSION

The paper explores a relatively small dataset containing violent and non-violent videos and applies various deep learning algorithms to detect violent crowd flows. The paper reveals that a convolutional neural network which leverages transfer learning outperforms all the opposite variance of convolutional neural networks and long STM networks. Moreover, the model finds out that the sequence models like LSTMs have performed worse than other models. By combining CNN with LSTM, the accuracy increases to a particular margin but still cannot beat the transfer learning models. However, in our future study, we'll be making this model more lightweight by pruning and can deploy it in an unmanned aerial vehicle. Moreover, we've plans to host our model during a web server and make an API to offer people access to our model.

REFERENCES

[1] C. Schuldt, I. Laptev, and B. Caputo, “Recognizing human actions: A local SVM approach,” in Proceedings - International Conference on Pattern Recognition, 2004.

[2] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri, “Actions as space-time shapes,” IEEE Trans. Pattern Anal. Mach. Intell., 2007.

[3] M. Marszałek, I. Laptev, and C. Schmid, “Actions in context,” in 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009.

[4] R. Poppe, “A survey on vision-based human action recognition,” Image Vis. Comput., vol. 28, no. 6, pp. 976–990, 2010.

[5] S.-R. Ke, H. Thuc, Y.-J. Lee, J.-N. Hwang, J.-H. Yoo, and K.-H. Choi, “A Review on Video- Based Human Activity Recognition,” Computers, 2013.

[6] A. Bewley, Z. Zongyuan, L. Ott, F. Ramos, B. Upcroft, “Simple Online and Realtime Tracking,” in Proc. IEEE International Conference on Image Processing, 2016, pp. 3464-3468.

[7] K. B. Lee, H. S. Shin, D. G. Kim, “Development of a deep-learning based automatic tracking of moving vehicles and incident detection processes on tunnels,” Korean Tunnelling and Underground Space Association, 2018, vol. 20, no.6, pp. 1161-1175.

[8] E. Bermejo, O. Deniz, G. Bueno, and R. Sukthankar, “Violence Detection in Video Using Computer Vision Techniques,” CAIP’11 Proc. 14th Int. Conf. Comput. Anal Images patterns - Vol. Part II, 2011.