VIDEO ANOMALY PREDICTION BASED ON SPATIAL AND TEMPORAL DATA

MASTERS PROJECT: ARTIFICIAL INTELLIGENCE AND BIG DATA

TARIQ HAQUE, AISHWARYA PHANSE, LIYA ZHANG

LIYA.SHANG@RHSMITH.UMD.EDU, AISHWARYA.PHANSE@RHSMITH.UMD. EDU , TARIQ.HAQUE@RHSMITH.UMD.EDU

UNIVERSITY OF MARYLAND

ABSTRACT

Surveillance videos can capture varied anomalous activities. In this project, we aim to classify videos as normal and anomalous. We chose a dataset of real-world surveillance activities of 376 videos that are labelled as normal or anomalous and extracted frames from the videos. Instead of a frame-level annotation, we made predictions at a video-level. After a preliminary processing, we used four types of models to train our dataset on, namely, CNN-LSTM, Conv3D, ConvLSTM and VGG-16 LSTM. A prior research using similar videos dataset for anomaly detection implemented Conv3D and TCNN using a multiple instance learning model however, the loss function used was incapable of capturing temporal information. Our project eliminates this issue by deploying models capable of capturing both spatial and temporal information.

Our first model, CNN-LSTM uses Conv2D wrapped in a Time-distributed layer followed by LSTM layer. While the resultant model captures spatial as well as temporal information, 2D CNN alone does not have the capability to capture temporal information which is why a Time-Distributed layer is required. To better this model, we use a Conv3D model that uses Conv3D layers as the name suggests, to capture spatial as well as temporal information. Just as a 2D CNN layer cannot capture temporal information, an LSTM layer cannot capture spatial information. Therefore, another variant we use is a ConvLSTM model that comprises convolutional operations within an LSTM layer enabling the model to again capture spatial and temporal information. The fourth variant we use is a VGG-16 LSTM model which comprises a pre-trained VGG-16 model that functions equivalent to a CNN layer followed by an LSTM layer.

Our research suggests ConvLSTM as the best model for this dataset which tested highest on accuracy as compared to other models. Going further, there is a lot more scope to improve accuracy by expanding our video dataset, modifying video processing and improving model architectures.

LITERATURE RESEARCH

Video surveillance has been an area of significant interest in both academia and industry. With the increasing demand for security, surveillance cameras are increasingly being used in public places e.g. streets, intersections, banks, shopping malls etc. to improve public safety, reduce crime rate, help catch criminals and curb illegal activities. Surveillance videos can capture a variety of realistic anomalous activities. However, there is a glaring deficiency in the utilization of surveillance cameras. This problem can be curbed by deploying an intelligent system like a machine learning algorithm to identify the number of instances of such activities. The goal of a practical anomaly detection system is to timely signal an activity that deviates normal patterns and identify the time window of the occurring anomaly. Therefore, anomaly detection can be considered as coarse level video understanding, which filters out anomalies from normal patterns.

Anomaly detection differs from the traditional classification problem in the following aspects:
1) It is very difficult to list all possible negative (anomaly) samples.
2) It is a daunting task to collect sufficient negative samples due to the rarity.

There has been some considerable work done in this challenging area of anomaly detection by researchers like Datta et al. who proposed to detect human violence by exploiting motion and limbs orientation of people, Kooij et al. who employed video and audio data to detect aggressive actions in surveillance videos, Gao et al. who proposed violent flow descriptors to detect violence in crowd videos and Mohammadi et al. who proposed a new behavior heuristic based approach to classify violent and non-violent videos.

However, there were certain limitations in previous researches:

1. Previous datasets captured certain activities that were non-anomalous like running and in some instances vague.
2. Certain researches captured activities only at one location.
3. Number of videos in majority of previous researches were very few leading to lack of enough training data.

To overcome these limitations, Waqas Sultani et al. used 128 hours, 1900 long and untrimmed surveillance videos from various webcam sources at different locations, divided them into a fixed number of segments (instances) during training, used positive (anomalous) and negative (normal) bags as input and trained anomaly detection model using proposed deep multiple instance learning (MIL) ranking loss which computed the ranking loss between the highest scored instances in the positive bag and the negative bag. The underlying assumption of the proposed approach was that given a lot of positive and negative videos with video-level labels, the network could automatically learn to predict the location of the anomaly in the video. To achieve this goal, the network should learn to produce high scores for anomalous video segments during training iterations. After 3,000 iterations, the network started to produce low scores for normal segments and keep high scores of anomalous segments. As the number of iterations increased and the network saw more videos, it automatically learnt to precisely localize anomaly. Although it did

not use segment level annotations, the network was able to predict the temporal location of an anomaly in terms of anomaly scores. Performance was evaluated on normal videos only as a major part of a surveillance video is normal and a robust anomaly detection method should have low false alarm rates on normal videos. The proposed false alarm rate was 1.9, much lower compared to previous researches. This validated that using both anomalous and normal videos for training helped the deep MIL ranking model to learn more general normal patterns. The models used Conv 3D in a 3-layer fully connected neural network and TCNN (Tube Convolutional Neural Network) and gave accuracies 23% and 28.4% respectively. While, this research worked on previous shortcomings, there arose another limitation of the loss function's inability to capture the underlying temporal structure of an anomalous video.

To enjoy representative capacity of neural networks, some other researchers like J. R. Medel et al. and Y. S. Chong et al. proposed a neural network which consists of a recurrent neural network (RNN) accompanied with convolutional filters. Their methods could adaptively learn long range contextual dynamics so that the motion and the appearance are implicitly encoded. Although these methods have shown promising performance, they have suffered from following two limitations. On the one hand, motions and appearances are encoded by the RNN and the convolutional filters separately, which implies that the spatial-temporal relations between motions and appearances are broken. As a result, inferior performance may be achieved. On the other hand, the features are typically learned from scratch without considering the well-established pre-trained model from relevant related tasks.

To improve upon this, in a more recent research, Joey Tianyi Zhou et al. proposed a new feature learning network, AnomalyNet which consisted of motion fusion block and feature transfer block. Specifically, the motion fusion block compressed video clips into a single image while suppressing the irrelevant background. As a result, the motion and appearance could be simultaneously fused into a single image. By feeding the compressed images into the feature transfer block, the spatial-temporal (i.e., appearance and motion) features could be extracted based on a transferable model. To overcome limitations like fixed learning rate, non-consideration of historical information and difficulty in optimization, this research used a novel RNN, sparse long short-term memory (SLSTM).

Performance evaluation metrics used were Area Under Curve(AUC) and equal error rate (EER). Here, two scales of measurement were picked, frame and pixel-level anomaly detection to determine number of true/false positive frames. Also, it checked the influence of different pre-trained models like Alexnet, VGG-16, Resnet-50 and Resnet-152 on AUC and ERR.

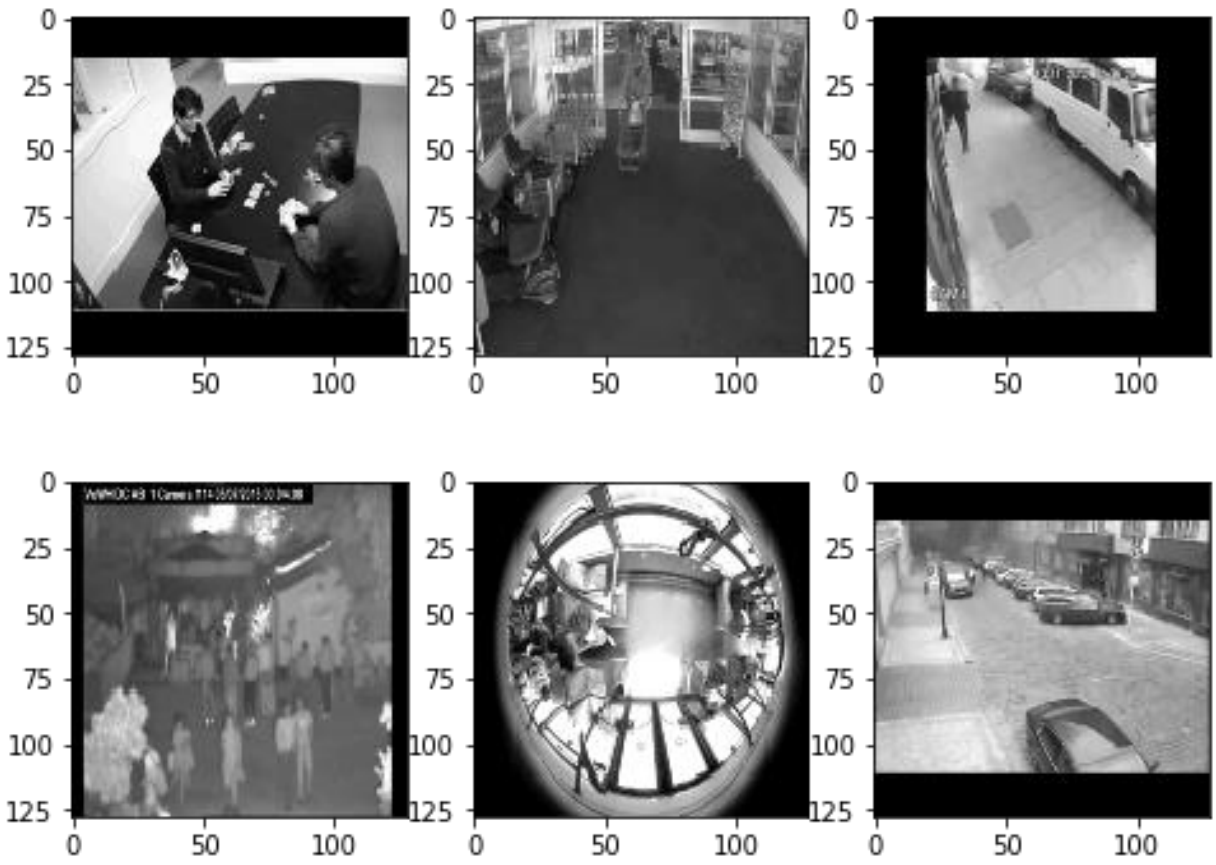
Previously used video surveillance datasets:

	No. of videos	Average frames	Dataset length	Example anomalies
UCSD Ped1	70	201	5 min	Bikers, small carts, walking across walkways
UCSD Ped2	28	163	5 min	Bikers, small carts, walking across walkways
Subway Entrance	1	121,749	1.5 hours	Wrong direction, No payment
Subwa Exit	1	64,901	1.5 hours	Wrong direction, No payment
Avenue	37	839	30 min	Run, throw, new object
UMN	5	1290	5 min	Run
BOSS	12	4052	27 min	Harass, Disease, Panic
Surveillance	1900	7247	128 hours	Abuse, arrest, arson, assault, accident, burglary, fighting, robbery

DATA DESCRIPTION

We used a labelled dataset of 376 real-world videos that comprise 232 normal videos and the remaining anomalous videos including 13 types of realistic anomalous activities such as fighting, road accident, burglary, robbery, etc. These videos are security webcam recordings at different public locations.

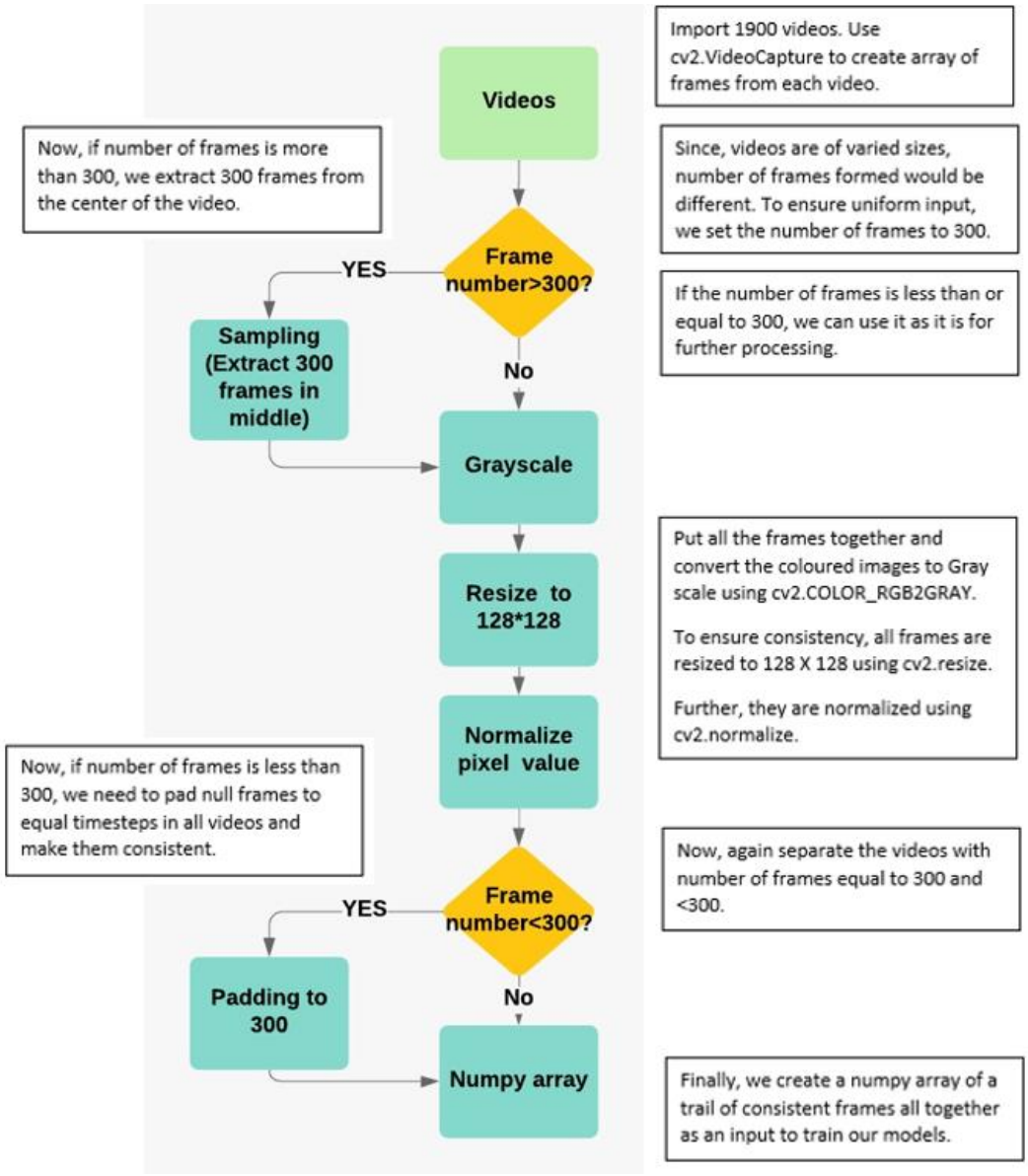
Below are certain samples of frames extracted from our videos to get a glimpse of the dataset we shall be training our models on.



VIDEO PROCESSING

Videos in our dataset need to be converted into an apt input format to feed into our models. We begin by extracting frames and equating number of frames for each video. Then we convert the videos to grayscale, resize the them to l * h = 128 X 128 and normalize them. Then we pad them and create an array of dimensions (376, 300, 128, 128, 1).

Notations: For simplicity, from now on we refer video clips with a size of c × l × h × w where c is the number of channels, l is length in number of frames, h and w are the height and width of the frame, respectively. Hence, here c=376, l=300, h=128, w=128. Fifth dimension 1 denotes grayscale.

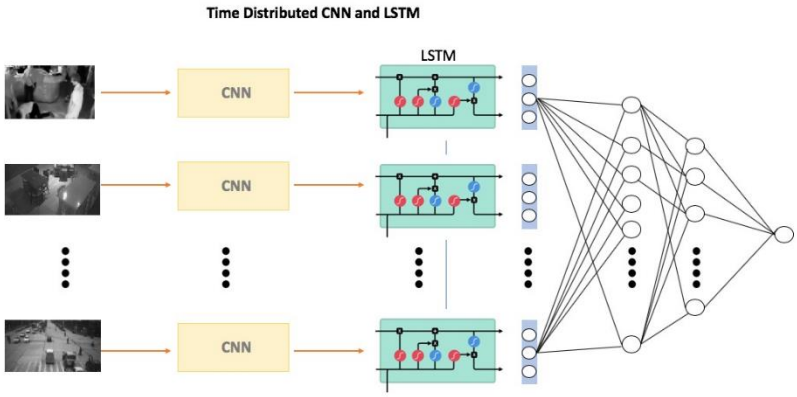


PROPOSED APPROACH

A video comprises of frame of images stacked together to form a continuous stream. This additional temporal feature adds to the complexity of video classification problems. While the added feature provides more information to train models, it also significantly increases computation requirement. In fact, most of the models trained in this research document are implemented using the computation power of Amazon Web Services (AWS). Compared to image classification problems, video classification problems also require different network architectures that can capture both spatial and temporal information. The following sections discuss the implementation of 4 suitable neural network architectures.

Notation: We also 3D convolution and pooling kernel size by d×k ×k, where d is kernel temporal depth and k is kernel spatial size and 2D convolution and pooling kernel size by k ×k, where k is kernel spatial size

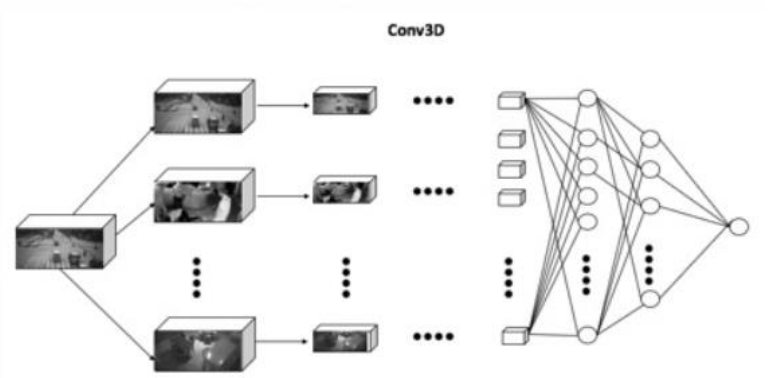
Models Architecture



CNN-LSTM: The model is a combination of CNN and LSTM layers where Conv2D (Keras) layers learn features in frames of a video followed by LSTM, which interprets features across time steps of a video. However, Conv2D model is only capable of handling single image. To make sure that CNN is applied to each image of a video and passed to LSTM as a single time step, CNN layers are wrapped in a TimeDistributed Layer. The resultant model is compositional in spatial and temporal layers [1]. The model architecture is defined below. Total trainable parameters were 7,182,069. We have used binary_crossentroy as loss function and adam as optimizer.

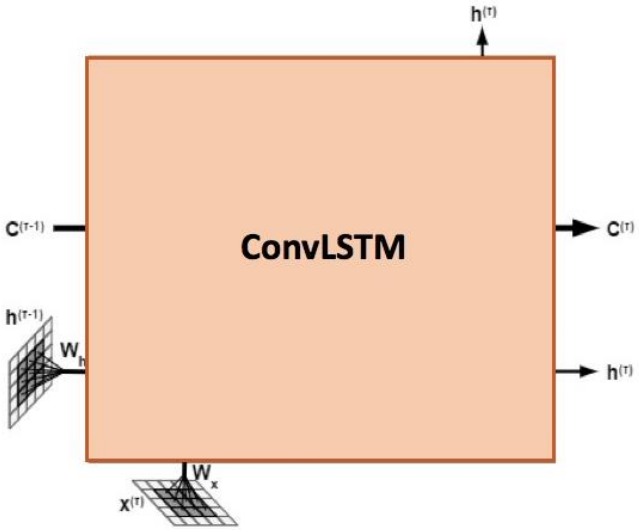
Layers (Type)	Output Shape	Number of Parameters
2D CNN, TimeDistributed	(None, 300, 124, 124, 64)	1,664
Maxpooling, TimeDistributed	(None, 300, 123, 123,64)	0
2D CNN, TimeDistributed	(None, 300, 119, 119, 32)	51,232
Maxpooling, TimeDistributed	(None, 300, 118, 118,32)	0
Flatten, TimeDistributed	(None, 300, 445568)	0
Dropout	(None, 300, 445568)	0
LSTM	(None, 4)	7,129,168
Dense	(None, 1)	5

Conv3D: While 2D CNN does not have the ability to model temporal information, 3D CNN can capture both temporal and spatial information and hence, are useful to build function to model videos that have both temporal and spatial information.[2] In 3D convolution neural network, kernels are also 3 dimensional. Due to lack of memory the model may be disadvantaged in learning a long sequence. Following are the model parameters and output size details. IN the first hidden layer, we have used 16 filters and kernel of 3*3*3, followed by maxpooling layer of pool_size=1*2*2. In the second Conv3D layer we have used 32 filters and kernel of 3*3*3, followed by maxpooling layer of pool_size=1*2*2. The Dense layer has 128 neurons followed by the output layer. Total trainable parameters were 1,091,188,961. We have used binary_crossentroy as loss function and adam as optimizer.



Layers (Type)	Output Shape	Number of Parameters
Conv3D	(None, 298, 126, 126, 16)	448
Maxpooling 3D	(None, 298, 63, 63,16)	0
Conv3D	(None, 296, 61, 61, 32)	51,232
Maxpooling 3D	(None, 8524800)	0
Flatten	(None, 300, 445568)	0
Dense	(None, 128)	1,091,174,528
Dropout	(None, 128)	0
Dense	(None, 1)	129

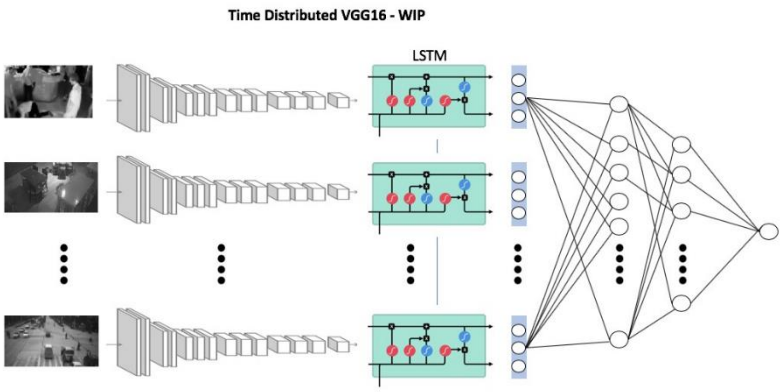
ConvLSTM: ConvLSTM is a variant of LSTM containing a convolution operation inside the LSTM cell. The models is a integration of CNN and LSTM where instead of doing matrix multiplication, convolution operation is conducted at each gate of a LSTM cell thereby, capturing the underlying spatial features by convolution operations in multiple-dimensional data. Standard LSTM input data is one-dimensional and not suitable to capture spatial data. As a result, ConvLSTM2D is expected to be more suitable for video classification problems.[3]



Following are the model parameters and output size details. In the first hidden layer, we have used 40 filters and kernel of 3*3, followed by second layer we have used 40 filters and kernel of 3*3, followed by a third layer of another 40 filters and kernel of 3*3. Finally, the output is flattened. Total trainable parameters were 19,660,801. We have used binary_crossentroy as loss function and adam as optimizer.

Layers (Type)	Output Shape	Number of Parameters
ConvLSTM2D	None, 300, 128,128,40	59,200
Batchnormalization	None, 300, 128,128,40	160
ConvLSTM2D	None, 300, 128,128,40	115,360
Batchnormalization	None, 300, 128,128,40	160
ConvLSTM2D	None, 300, 128,128,40	115,360
Batchnormalization	None, 300, 128,128,40	160
Flatten	None, 19660800	0
Dense	None, 1	19,660,801

VGG-16 LSTM : Like a CNN LSTM model architecture, VGG16 LSTM can be considered as a CNN layer, which learn spatial information followed by LSTM cells, which learn temporal information. However, with VGG-16 LSTM the CNN layer uses the pre-trained model weights. These pre-trained models are considered best-in-class and have proven to have extremely high capability to learn spatial information. As a result, in a VGG-16 LSTM model, the CNN is fixed for feature extraction from images and hence the CNN layer is not trained. This is expected to add to accuracy. VGG-16 model architecture comprises of 16 layers neural network pretrained on ImageNet 2012 classification challenge dataset. [5] In both CNN LSTM and VGG16 LSTM, conceptually there is a single CNN model and a sequence of LSTM models, one for each time step. To apply CNN model to each timestep (frame of a video), we wrap the layer as time-distributed. In the model, the last convolution layer of the VGG-16 network is connected with LSTM, followed by a flatten and output layer with sigmoid as activation function. Total trainable parameters were 15,766,337. We have used binary_crossentroy as loss function and adam as optimizer. Following are the model parameters and output size details.



Layers (Type)	Output Shape	Number of Parameters
VGG16 Layer – Time distributed	None, 300, 512	14,714,688
LSTM	None, 256	787,546
Dense	None, 1024	263,168

RESULTS AND CONCLUSION

In this study, we have successfully applied the machine learning approach, especially deep learning, to the challenging video anomaly prediction. Based on the models tested in this study, we found that ConvLSTM appears to have highest accuracy. The highest accuracy achieved during the study was 54.19%. Despite novelty of the approach adopted in the study, accuracy is still lower than the benchmark studies. We believe that there is a strong motivation to carry forward the models discussed in this report to further improve accuracy.

Model	Model Input Shape	Validation Accuracy
CNN LSTM	(376, 300, 128, 128, 1)	53.98%
CNN_3D	(376, 300, 128, 128, 1)	53.98%
ConvLSTM2D	(376, 300, 128, 128, 1)	54.19%
VGG16 LSTM	(376, 300, 128, 128, 1)	53.98%

FUTURE RESEARCH

In this paper, we have implemented various variant of CNN models to capture both spatial and temporal data in the video. We have achieved a significant progress in achieving a maximum accuracy of 54.19 % in prediction of anomalous videos. However, we believe that there is a significant need to carry this research forward especially by changing model architecture to improve prediction accuracy. In addition, of having used only 376 of the 1900 available videos due to computational restrictions. We believe that prediction can further improve if complete dataset is used. We would also like to highlight that we have sampled videos based on 300 frames in the middle of video. While, this is a regularly adopted method, other methods for sampling should also be explored in the pursuit of prediction accuracy. These include equal interval frame sampling and recently proposed plug-and-play PickNet method to pick framed based on a reinforcement-learning-based procedure where the reward of each frame picking action is designed by maximizing visual diversity and minimizing textual discrepancy.[4]

REFERENCES

1. *Long-term Recurrent Convolutional Networks for Visual Recognition and Description*, Jeff Donahue, Lisa Anne Hendricks, Marcus Rohrbach, Subhashini Venugopalan, Sergio Guadarrama, Kate Saenko, Trevor Darrell, TPAMI Journal (November 2014)
2. *Learning Spatiotemporal Features with 3D Convolutional Networks*, Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, Manohar Paluri, Facebook AI Research and Dartmouth College (October 2015)
3. *Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting*, Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, Wang-chun Woo, NIPS (Sepotember 2015)
4. *Less Is More: Picking Informative Frames for Video Captioning*, Yangyu Chen, Shuhui Wang, Weigang Zhang, and Qingming Huang, ECCV (March 2018)
5. *Very deep convolutional networks for large-scale image recognition,* K. Simonyan and A. Zisserman, arXiv:1409.1556, 2014
6. *Real-world Anomaly Detection in Surveillance Videos,* Waqas Sultani, Chen Chen, Mubarak Shah, University of Central Florida (Feb 2019)
7. *AnomalyNet: An Anomaly Detection Network for Video Surveillance,* Joey Tianyi Zhou, Jiawei Du, Hongyuan Zhu, Xi Peng, Yong Liu, Rick Siow Mong Goh, IEEE (2019)