An Efficient Deep Learning Model for Violence Detection

Mahmudul Haque[a], Syma Afsha[a], Hussain Nyeem[a]

[a]*Department of Electrical, Electronic and Communication Engineering, Military Institute of Science and Technology (MIST), Dhaka, 1216, Bangladesh*

Abstract

Automatic Violence Detection and Classification (AVDC) with deep learning has garnered significant attention in computer vision research. This study presents the development of a new hybrid model, BrutNet, based on a Deep Convolutional Neural Network (DCNN) and Gated Recurrent Unit (GRU). BrutNet was designed to identify violent patterns across numerous frames of a video or video clips of form 160×90, with a duration of at least 3–5 seconds. A time-distributed convolutional network (TD-ConvNet) was used to extract the feature set and pattern of each frame, which the model converted from 4D to 2D, producing a 512-feature set for each frame. The GRU layer then leveraged the temporal nature of these frames and returned a 1D vector, which was subsequently processed by multiple dense layers, enabling binary classification of the content as violent or non-violent. To avoid overfitting, the model included dropout layers with a dropping rate of 0.5. Moreover, ReLu-activation and sigmoid-activation functions were developed for the hidden and output layers, respectively. The model was trained on Google Colab's NVIDIA Tesla K80 GPU using a custom movie clip dataset and suitable hyper-parameters, achieving a test accuracy of 80.58%. The model's performance was compared to other models using various video datasets, including hockey fights, movie fights, AVD, and RWF-2000. BrutNet outperformed previous state-of-the-art models in terms of efficiency and runtime, requiring only 3.415 M parameters in the model and achieving test accuracy of 97.62%, 100%, 97.22%, and 86.43% for the respective datasets. BrutNet is thus found to be an effective AVDC model that can accurately and efficiently detect violent content in videos.

Keywords:
BrutNet, violence detection, activity recognition, deep learning, CNN, GRU

1. Introduction

In the realm of Deep Learning (DL), violence detection has recently emerged as a critical problem [1], leading to an increased focus on using DL, computer vision, and image processing methods for flagging unsuitable content and detecting violent behaviour [2]. Real-time video review is imperative to identify violent offenders and maintain the safety of cities [3]. As visual materials become more abundant on online platforms such as YouTube, Facebook, Twitter, and Netflix, their open availability and the lack of a suitable monitoring or certification body demand an automated classification of sensitive content [4, 5, 6]. Detecting violent behaviour could also be beneficial in video surveillance applications for facilities like prisons, mental or elderly care facilities, and camera phones [7]. It would also help the public presentation of visual contents, allowing viewers to easily avoid inappropriate content.

Violent video content is characterized by conflict aggressiveness, damaging conduct, psychological instability, and the deliberate use of physical strength or force against another person or group [8]. However, despite the complexity of violent scenarios, little attention was paid to identifying violent or hostile behaviours. For example, research on action recognition has primarily focused on identifying basic motions such as clapping, walking, or running [9]. Besides, videos consist of images displayed sequentially to convey movement, and therefore, it is important to use both spatial and temporal characteristics while analysing video. Detecting violent behaviour in videos is challenging because it involves a variety of events and activities [10].

Various methods have been proposed to detect violent behaviour using machine learning, support vector machines, and deep learning techniques [1]. Deep Convolutional Neural Networks (DCNN) are commonly used with deep learning techniques to identify violent situations [11]. Although Convolutional Neural Networks (CNNs) have shown remarkable performance in image categorization and object recognition tasks, their limitation is that they can analyse only one picture at a time and cannot identify visual data in a time series [12]. To overcome this limitation, researchers have used sequential learning models such as Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) to increase classification accuracy in action recognition [13]. LSTM, which contains three gates (input, output, and forget gates) and a memory cell that stores previous sequence information, is computationally expensive, which makes the activity recognition system inefficient. In contrast, Gated Recurrent Unit (GRU), a simpler RNN variation with two reset and update gates but no memory cell, has the ability to learn long-term sequences.

In this research paper, we propose an architecture named BrutNet that combines DCNN with GRU to improve the accuracy and efficiency of the violence detection process in videos. The proposed model leverages the strengths of both techniques and can be applied in various real-time applications that require

the detection of violent behaviour in videos. specifically, the BrutNet model is designed to detect and classify video data based on the presence of violence, and it has been trained on our custom Movie Clip dataset along with other related datasets. The promising results of our BrutNet model demonstrate its ability to classify violent and non-violent videos accurately, making it a suitable candidate for use in the AVD and Classification System (AVDCS). With our proposed architecture, we contribute to the development of more effective violence detection systems.

This paper provides a detailed account of the procedures used, implementation strategies, and performance analysis to the new development of AVDCS. In Sec. 2, we outline the procedures used in related works on AVDCS, followed by the implementation of the BrutNet algorithm in Sec. 3. We then describe the datasets used in this study and their processing in Sec. 4. Sec. 5 presents the experiment settings and performance evaluation parameters, while Sec. 6 analyses the results and key findings. Finally, Sec. 7 summarizes the outcomes and significance of our work, suggests avenues for future research, and presents our concluding remarks.

2. Related Work

Automatic violence detection methods are primarily investigated for scenarios such as video surveillance, movie clips, and road-traffic [14, 15]. DL-based approaches have recently shown great potential in violence characterization using multi-processing layer models, resulting in significant improvements in violence detection and classification with human activity identification, image or video-based object or pattern recognition, and emotion detection [16, 17]. We will analyse the development of these methods below, considering their varying video contents, network architectures, and training datasets.

2.1. Surveillance Video analysis

To address the issue of activity detection, Ullah *et al*. [13] proposed a lightweight DL-aided system. They employed a CNN network trained on two surveillance datasets to detect a person in a surveillance stream initially. They used a Minimum Output Sum of Squared Error object tracker, MOSSE, to track the person through the video stream. The Efficient LiteFlowNet CNN extracts pyramidal convolutional features from two consecutive frames for each monitored person. Additionally, they utilized a Deep Skip Connection Gated Recurrent Unit (DS-GRU) to learn the temporal variations in a frame sequence for activity detection.

To identify and locate violent activities in video surveillance, Roman *et al*. [18] used dynamic pictures to categorize a video as violent or non-violent, and then used CNNs and weakly supervised localization algorithms to locate violent regions. Hockey Fight [7], Violent Flows and UCFCrime2Local were all employed in the study. According to their technique, the Hockey Fight dataset had an accuracy of 96.40%, Violent Flows had an accuracy of 92.04%, and UCFCrime2Local had an accuracy of 79.11%. According to Chatterjee *et al*. [19], the purpose of this study was to improve the categorization of violent and non-violent actions in public settings. They employed a convolutional bidirectional LSTM to identify violent activities, and the results were compared to other techniques that were already in use. Their strategy provides a classification accuracy of 94.06% for the standard Hockey dataset, which is extensively utilized.

2.2. Movie clips analysis

Peixoto *et al*. [20] proposed using two deep neural network (DNN) frameworks, C3D and CNN-LSTM, to detect violence in movies. The frameworks were applied to learn spatial-temporal information from video clips under two distinct scenarios: subjective and conceptual-based violence detection. In the former, violent ideas were detected by searching for required concepts in the videos, while in the latter, the unique notion of violence was applied independently of the first situation. The fusion of ideas was analysed as a whole to determine the higher-level notion of violence. Two DNNs were then employed to identify violence in videos. Similarly, Gruosso *et al*. [21] developed a content grading system based on CNN for evaluating materials for children, teens, and adults. They also created an algorithm to categorize and restrict violent situations automatically. To train and verify the Inception v3 architectural model, they utilized a large hand-labelled dataset containing visual components useful for categorization. For model evaluation, they created an algorithm to enhance the network performance for video input.

2.3. Road-traffic analysis

Deep convolutional neural networks (DCNNs) have significantly impacted various domains of pattern recognition [22]. Fu *et al*. [23] demonstrated that DL approaches based on RNNs, such as LSTM and GRU, outperform other sequential models when used to predict short-term traffic flow. Cheng *et al*. [24] introduced the RWF-2000 database, which contains 2,000 films recorded by security cameras in real-world scenarios and utilizes 3D-CNN and optical flow. The model employs self-learned pooling to adapt to both appearance and temporal features. Guedes *et al*. [25] proposed a CNN and SVM classifier-based strategy for identifying instances of aggressive behaviour in video streams containing violent altercations.

Das *et al*. [26] presented a technique for recognizing violence that involves selecting several frames from each video clip using image removal and averaging, and extracting lower-level features using HOG. Finally, SVM, LDA, Naive Bayes, and K-Nearest Neighbors (KNN) were utilized for classification. Jain *et al*. [27] discussed the different CNNs employed for video violence detection, their advantages, and drawbacks. The technique suggested by Honarjoo *et al*. [28] involves utilizing pre-trained deep neural networks, specifically ResNet-50 and VGG16, to identify violent actions using extracted features from pre-trained models, providing a technique with a minimal level of complexity for identifying instances of violence.

According to Miriana *et al*. [29], most of the datasets that may be used for violence identification are made up of a few

clips, have poor resolution, and are often constructed on examples that are too particular, such as fights that take place in hockey. They suggested using high-resolution datasets to assess the resilience of violence detection systems against false positives. Ditsanthia *et al*. [30] suggested a unique DL-based video-based AVD. They analysed the findings of numerous AVD approaches and found ResNet50+LSTM to be the most accurate. They employed common datasets like *hockey*, *movie* and *real-violent* to get top accuracy of 83.19%, 88.74% and 77.75%. However, Haque *et al*. [31] developed a novel time-distributed DCNN and GRU-based architecture called BrutNet, and they utilized this model to present an AVDC system that can reliably recognize and categorize the visual content depending on the presence of violent actions. The model had been trained with a recent high-resolution AVD video dataset, and their model demonstrated the then state-of-the-art test accuracy of 90.00%.

Several other approaches like ViolenceNet [32], Efficient 3D CNN [33], Xception + BiLSTM-based approach [34], C3D [35], AlexNet + LSTM [36], CNN-LSTM [37], Hough Forests + 2D CNN [38], Three Streams + LSTM [39], MoSIFT [40], motion intensities + AdaBoost [41], ResNet50 + ConvLSTM [42], Fine-tuned MobileNet [43], Motion Blobs + Random Forest [44] etc. have also been considered for violence detection and demonstrated significant results. In a later work, Vijeikis *et al*. [45] showed a similar approach as Haque *et al*., for addressing an efficient violence detection problem. They used a time-distributed MobileNetV2 and LSTM-based model for the purpose and demonstrated an accuracy of 82.0 ± 3%, 96.1 ± 1% and 99.5% accuracy for RWF-2000, Hockey Fights and Movie Fights Dataset [7] respectively which is the current state-of-the-art accuracy given the number of parameters in their model (4.074 M).

3. The Proposed BrutNet Model and AVDC System

This section will now present the proposed AVDC system, which is composed of time-distributed convolutional networks (ConvNets) and GRU layers for the violence classification network. DCNN-based AVDC models have already demonstrated their potential for image and video-based detection and classification problems in real-time. However, ConvNets cannot be used directly for the surveillance and identification of violent web content because it operates on a single image to learn its properties. Determining the intrinsic characteristics of a violent incident in a video requires sequential patterns from successive image sets or frames. This indicates that sequential models, such as GRU, are more capable of recognizing the pattern between multiple frames of a video clip to recognize violent activities. Our model consequently combines custom time-distributed ConvNets and GRU-based RNN to optimize the learning of violent video characteristics. The primary processing phases of the AVDC system are depicted in Fig. 1. For clarity and without sacrificing generality, these steps are explained briefly in the subsections that follow.

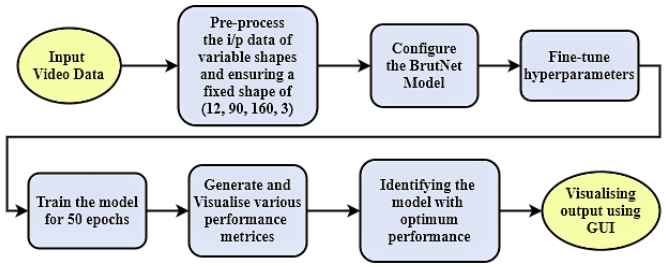


Figure 1: Processes of the proposed AVDC system

Table 1: Trainable parameters for BrutNet architecture.

Layer (type)	Output Shape	Parameter #
Time Distributed CNN	(None, 24, 512)	1591744
GRU	(None, 64)	110976
Dense	(None, 1024)	66560
Dense	(None, 1024)	1049600
Dropout	(None, 1024)	0
Dense	(None, 512)	524800
Dropout	(None, 512)	0
Dense	(None, 128)	65664
Dropout	(None, 128)	0
Dense	(None, 64)	8256
Dense	(None, 1)	65
Trainable Parameters		3,415,745

3.1. The BrutNet architecture

The proposed AVDC system operates on a set of temporal data obtained from the processed video clips. Since video data is essentially a sequential flow of images, the temporal nature of specific features, *i.e*., the components present in these images are to be determined for detecting the activity within the video. Particularly, once the raw data is pre-processed as discussed in the earlier section, each sample of the dataset, *i.e*., video clips of shape (24, 90, 160, 3), has been fed to the developed BrutNet network of the proposed AVDC system. The BrutNet model is designed with a set of initial convolutional layers for each frame of the time-distributed layer, as shown in Fig. 2. The model encodes the sample input of 4D data to 2D data, where we obtained 512 feature values for each frame. As a result, the model can determine the features present in each frame of the video clip sample for obtaining the pattern in a video clip sample. To analyse the temporal nature of the frames, an RNN-based GRU layer was designed to find the patterns between all the frames of the video clip, which resulted in an output of a 1D vector. This 1D vector was passed through several dense layers. These subsequent dense layers processed the 1D vector to perform a binary classification where the generated output is either 1 or 0, which denotes violent and non-violent, respectively. We have also used dropout layers in between layers with a dropping rate of 0.5 to avoid overfitting the model. The hidden layers had ReLu activation functions, and the output layer had a sigmoid activation function to generate binary outputs. The designed model had a total of 3,415,745 (3.416 M approx.) trainable parameters as shown in Table. 1.

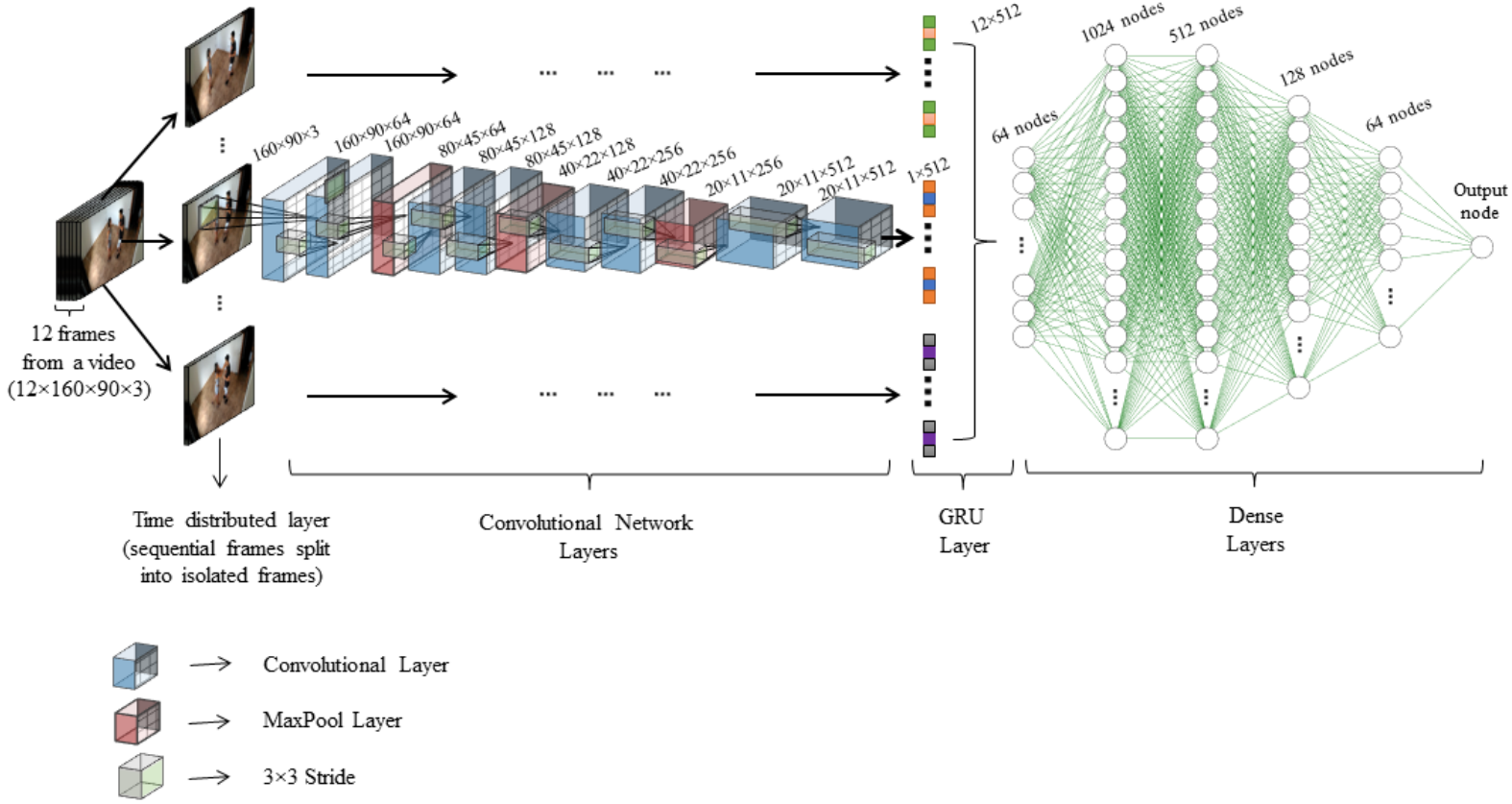


Figure 2: Architecture of BrutNet

3.2. Model training

The proposed BrutNet model has been trained with several datasets mentioned subsequently using a Google Colaboratory environment having a 2.30GHz Intel(R) Xeon(R) CPU, 12.63 GB RAM, and 12 GB NVIDIA Tesla K80 GPU. After preprocessing the data, a batch size of 24 was determined, considering the hardware limitations considering the size of each sample. While training the BrutNet model, an initial learning rate of 10^{-5} has been set, to minimize the loss function. For optimization weighted binary cross entropy using equation (1) and assigning the class-wise weights while configuring the training, was used as the loss function since many of the datasets used, were imbalanced.

$$E_{BC} = -\frac{1}{N}\sum_{m=1}^{N} y_m \times log(P(y_m)) + (1 - y_m) \times log(1 - P(y_m)) \tag{1}$$

In equation (1), E_{BC} is the binary cross entropy, N is the number of samples in the dataset and y_m is the label for the m^{th} sample. By default, the relative weights were 1:1 for both classes for the balanced dataset. To optimize this minimization process, an Adam optimizer has been used. The model was trained for 150 epochs and the minimization of the loss function along with the increase in accuracy has been monitored. After each epoch, for readjustment and fine-tuning of the weights, we validated the model with a validation dataset that we created earlier. This instance of the model after every epoch of training has been evaluated, with a test dataset created earlier to verify its performance. After each epoch, the model has been saved so that the model with the best accuracy can be used further.

The BrutNet model is optimized for the best accuracy determined from the performance metrics stored during training, as discussed in Sec. III-C. The output is thus generated from the optimized model to classify the video clip into violent and non-violent classes. A python script has been written in OpenCV 4.5 and TF for this classification using a default GUI. We note that for testing video clips of a duration smaller or larger than 3-5 seconds (which is the size of the training video clips), necessary padding or looping is carried out. For example, the smaller clips have been padded with additional blank frames at both the beginning and ending to equalize the clip size, *i.e.*, maintaining a clip size of 3-5 seconds. From each of these processed clips, 24 frames have been extracted. In contrast, for larger clips, we have looped through the whole clip, considering clips of the duration of 3-5 seconds at a time. When any of these clips in the loop has violent content in it, the entire clip is then classified *violent*, else the content is labelled as *non-violent*.

4. Dataset Processing

As outlined in Sec. 2, the most significant obstacle is that the available datasets for AVDC are primarily low-resolution clips in insufficient quantities. Again, the widely used datasets have less diversity of violent scenes. Hence, our model was trained and validated using AVD dataset[29]. To compare the effectiveness of the proposed BrutNet architecture, the process needs to be repeated for a custom movie clip (MC) dataset using movie clips obtained from different English movies, the Hockey Fight (HF) Dataset [7], the RWF-2000 dataset [24], and the Movie Fights (MF) Dataset [7]. The details of the datasets and their processing have been described further.

4.1. Datasets

4.1.1. AVD Dataset

There were a total of 350 video clips, each with a 1920 × 1080 resolution and a frame rate of 30fps, which were captured by two cameras positioned from two separate vantage points and employing two cameras. The two cameras were the Asus Zenfone Selfie ZD551KL (13 MP, autofocus, f/2.2) and the TOPOP-Action-Cam OD009B (12 MP, fisheye lens 170°). We have categorized and labelled them as *violent* or *non-violent* clips from the AVDC dataset described above. The clip lengths varied between 75 and 435 frames. The dataset comprised a total of 350 video clips against both the labels. 20% (72 clips) of the total number of clips in the processed dataset were set aside for testing. Again, 80% (222 clips) and 20% (56 clips) of the remaining number of clips were randomly determined as the training and validation datasets, respectively. It was observed that the dataset was unbalanced between the *violent* and *non-violent* classes. Samples of the *violent* class were twice as many as the samples in the *non-violent* class. So, while training, this variation needs to be addressed in the weighted loss function for the proposed BrutNet architecture.

4.1.2. MC Dataset

In our work, we have created a dataset with a total of 1377 video clips which are segments obtained from several movies, each with a resolution of 1920 × 1080. These movie clips had a frame rate of 24 fps. The number of frames in these clips varied between 97 and 172 frames. Each of the movie clips was labelled as *"violent"* or *"non-violent"* classes. To prevent biasing when used to train a model, 464 movie clips for each class were considered. The dataset was preprocessed, and for training and evaluation of the trained model, the processed dataset was then divided into training, testing, and validation datasets, each containing 649 clips (70%), 139 clips (15%) and 140 clips (15%) out of the total 928 samples used for training, respectively.

4.1.3. HF Dataset

It is a very popular dataset, widely used for violence detection. The videos mainly comprise different scenes in several hockey matches. The dataset comprises a total of 1000 clips. Each clip of the dataset had between 41 and 50 frames. These clips were of two types, 50% were with fighting scenes and 50% did not contain any fighting scenes. The clips with fight scenes were considered to be violent scenes and the rest were considered non-violent, hence, were labelled 1 and 0 accordingly. There were 500 clips for each type of sample. The dataset was split into training, validation, and test dataset split each containing 700 clips (70%), 150 clips (15%) and 150 clips (15%) respectively.

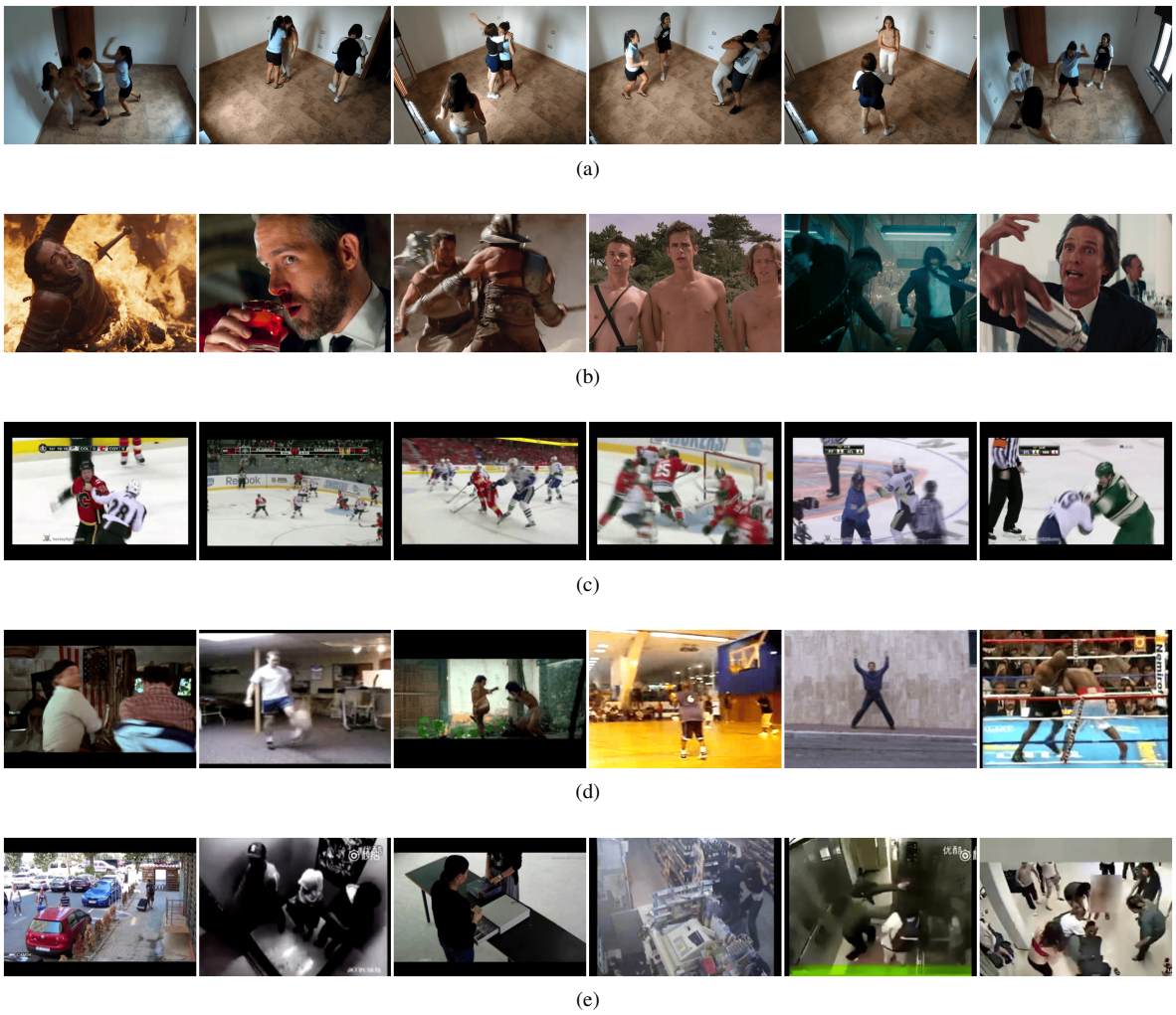


Figure 3: Sample of different datasets: (a) AVD Dataset, (b) MC Dataset (c) HF Dataset, (d) MF Dataset and (e) RWF-2000 Dataset.

4.1.4. MF Dataset

It is another popular dataset for violent content detection. It contains a total of 200 video clips. These clips are divided into two classes, *fights* and *non-fights*, each class containing 100 clips. The fight scenes were considered to be violent scenes and the non-fights scenes were considered non-violent, hence, were labelled 1 and 0 accordingly. The dataset was split into training, validation, and test dataset split each containing 140 clips (70%), 30 clips (15%) and 30 clips (15%) respectively.

4.1.5. RWF-2000 Dataset

It is another dataset similar to HFD and MFD with 2000 samples, each containing 151 frames with highly varying resolutions and aspect ratios. These samples, *i.e.*, video clips, are split into 1600 (80%) test samples and 400 (20%) training samples by default. The total dataset is labelled into two classes, *Fight* and *non-fight* each with 1000 samples. The *fight* classes are considered violent classes and *non-fight* classes are considered non-violent classes. For our work, we have further split the default training dataset into a training and validation dataset comprising 1200 clips (60%) and 400 clips (20%) respectively.

4.2. Processing

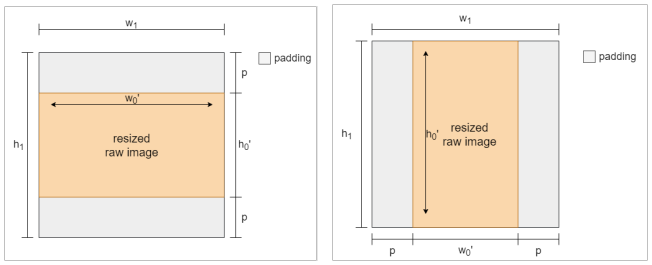
To address the limitations of the hardware resources utilized during model training, the resolution of each video clip was resized to 160×90 from its initial resolution, with each frame of its coloured frames having the shape of (90, 160, 3). It is seen that in some datasets used, the clips are of varying resolution and aspect ratios. To address this issue, we developed a method to scale up/down the resolution and add necessary padding to the image to ensure a consistent predefined resolution and aspect ratio. We consider h_0 and w_0, to be the number of pixels along the height and width of the frames of the raw video clip where the aspect ratio is a_0 in equation (2),

$$a_0 = \frac{w_0}{h_0} \tag{2}$$

and h_1 and w_1, to be the desired number of pixels along the height and width of the frames where the aspect ratio is a_1 in equation (3).

$$a_1 = \frac{w_1}{h_1} \tag{3}$$

Now, to fit the raw image with aspect ratio a_0 into the desired



(a) (b)

Figure 4: Resizing and padding of the raw image, keeping the aspect ratio intact.

shape with aspect ratio a_1, the raw frames are to be resized with width and height of w_0' and h_0' respectively and padding is added to the top and bottom of the resized image as $a_1 < a_0$ in Fig. 4a and left and right of the resized image for $a_1 > a_0$ in Fig. 4b to have an image of the desired shape without distortion due to change of aspect ratio using equation (4a), (4b) and (4c).

$$h_0' = \begin{cases} \frac{h_0 w_1}{w_0}, & a_1 < a_0 \\ h_1, & a_1 \geq a_0 \end{cases} \tag{4a}$$

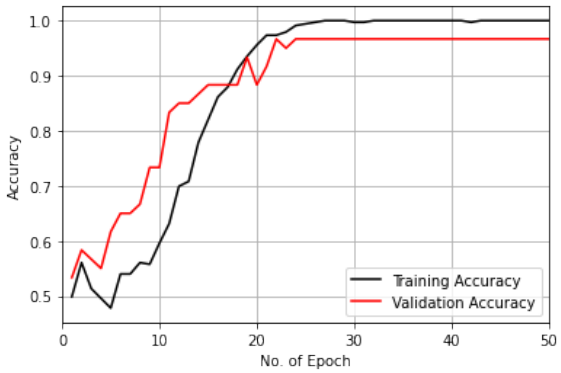
$$w_0' = \begin{cases} w_1, & a_1 \leq a_0 \\ \frac{w_0 h_1}{h_0}, & a_1 > a_0 \end{cases} \tag{4b}$$

$$p = \begin{cases} \frac{h_1 - h_0}{2}, & a_1 < a_0 \\ 0, & a_1 = a_0 \\ \frac{w_1 - w_0}{2}, & a_1 > a_0 \end{cases} \tag{4c}$$

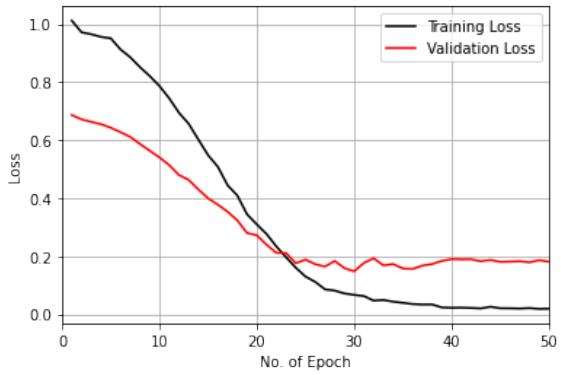
The range of pixel values for each coloured video frame is between 0 and 255. These pixel values have been normalized to a scale of 0 to 1 to prevent biasing of our model during training and hence need to be done before testing too. In addition, each video clip had a variable number of frames. For further processing and training, from each of these clips, 24 random and equally spaced frames were arbitrarily considered such that these frames could represent the overall content of the whole video. As a result, each sample of the dataset has a 4D data shape of (24, 90, 160, 3). For assessment purposes, the violent and non-violent labels have been binarised. The processed samples were compressed and saved locally, which were later loaded using a data generator on TensorFlow (TF) 2.5 framework.

5. Experiment Settings

This section discussed the implementation of our proposed method and its outcome. A model trained based on the proposed BrutNet architecture. The model was primarily trained using AVD Dataset. The model was trained for 50 epochs, and the training and validation accuracies and loss values per epoch were visualized for comparison in Fig. 5. From the plots, it is observed that the model demonstrated the most optimum accu-



(a)

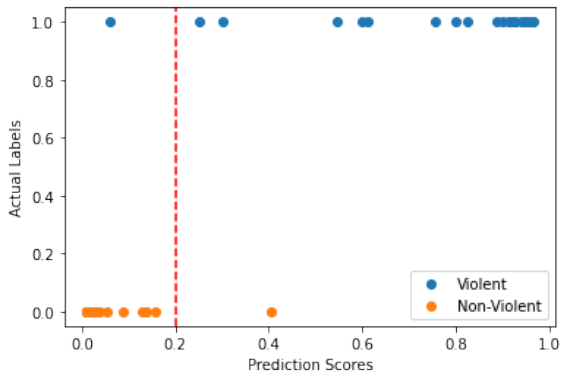


(b)

Figure 5: Convergence trends over the epochs: (a) accuracy, (b) loss

racy and loss value after 24 epochs, and hence the model obtained after 24 epochs of training, was the optimized model. This model was tested using the test dataset, and it was able to successfully classify 97.22% of the test samples.

Now, given this high accuracy, it is a requirement for an AVD classifier to be able to detect the maximum number of violent scenes. To ensure this, the scatter plot of the model to be able to classify the test data of AVD Dataset has been plotted in Fig. 6a. The confusion matrix has also been shown in Fig. 6b for the classifier to determine the suitable threshold for our desired outcome.



(a)

(b)

Figure 6: (a) Scatter Plot and (b) Confusion Matrix for BrutNet on AVD Dataset

From Fig. 6a, the threshold is set at 0.2 such that the maximum number of violent scenes are correctly detected without increasing the number of false detection significantly.

To further validate the effectiveness of this threshold to ensure a high recall/true positive rate (TPR) (Eq.(5a)) without compromising much of its false positive rate (FPR) (and Eq.(5b)), we have plotted the Receiver Operating Characteristic (ROC) curve and also find out the performance of the BrutNet classifier. The ROC curve has been shown in Fig. 7. Here, the TPR and the FPR have been calculated using the True Positive (TP), True Negative (TN), False Positive, and False Negative (FN) values for various thresholds such that,

$$TPR = \frac{TP}{TP + FN} \tag{5a}$$

$$FPR = \frac{FP}{FP + TN} \tag{5b}$$

It was observed that the trade-off between the recall and FPR was optimum for an FPR of 0.056 which resulted in a recall of 97.14%. This outcome is possible for a threshold of roughly 0.2. Moreover, the area under the ROC curve (AUC) was 0.983 which tends to the maximum possible area of 1 for the theoretically perfect classifier.

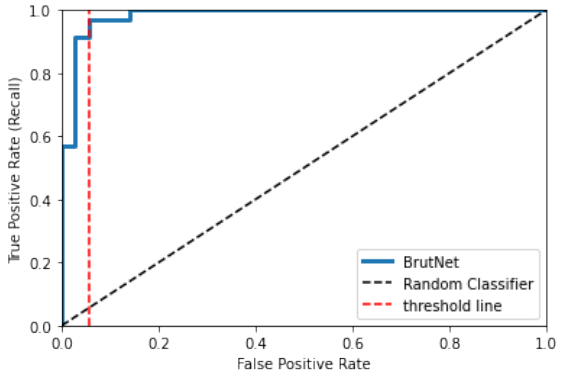


Figure 7: ROC curve for BrutNet Model.

Now, to validate the performance of the model, our model was further trained and tested on the following datasets with similar conditions:

1. HF Dataset
2. MF Dataset
3. MC Dataset
4. RWF2000 Dataset

After training on our BrutNet model, our model demonstrated test accuracies as summarized in Table. 2.

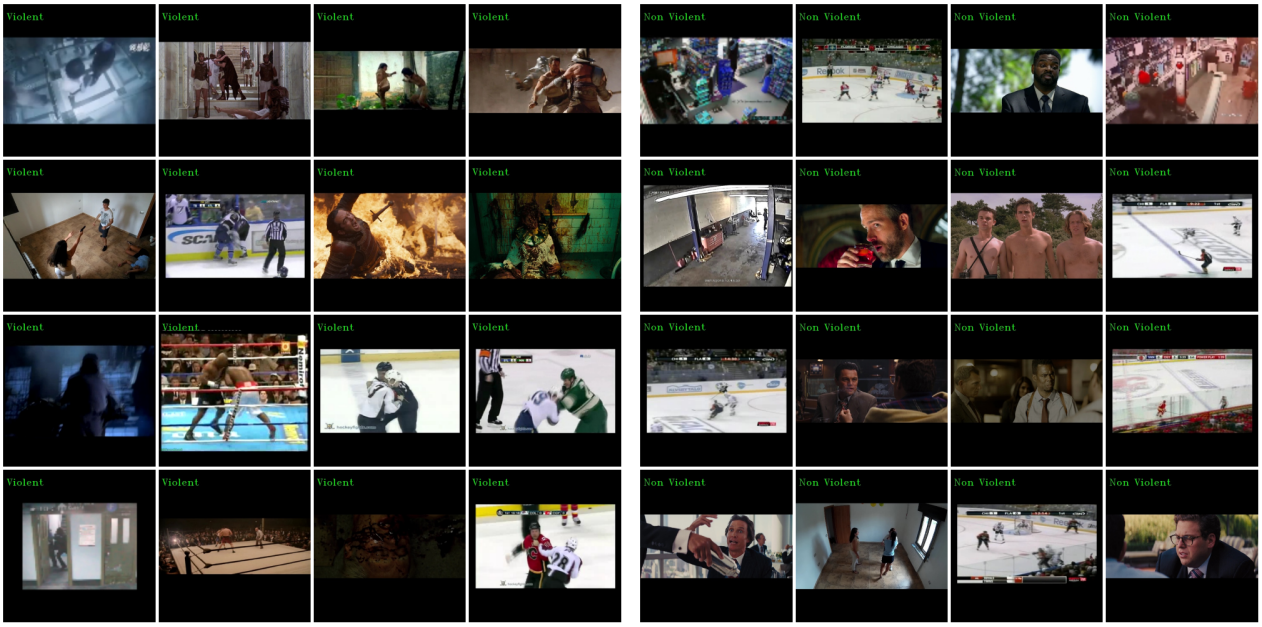
Table 2: Performance of BrutNet across AVD, HF, MF, MC and RWF2000 datasets.

Model	Dataset	Accuracy
BrutNet	**AVD**	**97.22%**
	HF	97.62%,
	MF	100%
	MC	80.58%
	RWF2000	86.43%

Some classified images by the model has been shown in Fig. 8.

6. Result Analysis

The performance metrics of the model have been compared with the performance of several models for violence detection



(a) (b)

Figure 8: (a) Violent and (b) Non-Violent classifications by BrutNet classifier.

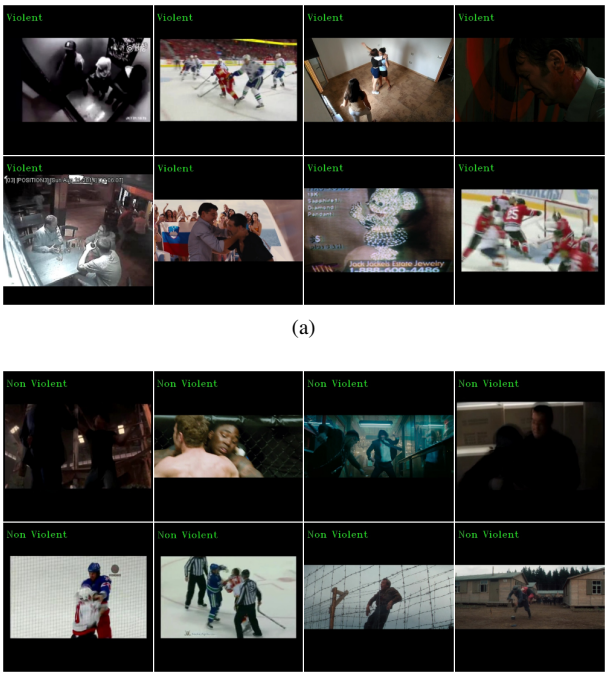
in this section to justify the significantly increased performance of BrutNet. For this, we have considered the performance of an earlier version of BrutNet [31], ViolenceNet [32], Efficient 3D CNN [33], Xception + BiLSTM-based approach [34], C3D [35], AlexNet + LSTM [36], CNN-LSTM [37], motion intensities + AdaBoost [41], ResNet50 + ConvLSTM [42], Motion Blobs + Random Forest [44] and the state-of-the-art MobileNetV2 + LSTM [45] models. A comparison between their performance across various datasets has been illustrated in the Table. 3.

From Table. 3 it is clear that the proposed BrutNet model provides a state-of-the-art high accuracy with significantly less number of the parameter that is crucial for the implementation of such a model for real-time application and in-edge devices.

Furthermore, to ensure the generality of the model for AVD applications it was trained with our custom MC Dataset which provided an accuracy of 80.58%. Still, several erroneous outputs were observed, as seen in Fig. 9. The reason behind it is the higher loss value of the model for more complex and diverse datasets. Hence, custom loss functions can be explored to optimize the model further.

7. Conclusion

Humans have always been concerned about their safety or security from the beginning of time. Effective monitoring is the only way to guarantee public safety and uphold the rule of law. Real-time violence detection may thus play a significant part in the cause. Mental health may be impacted by seeing violent acts. Science and technology have made it possible to post millions of pieces of material every day, which need adequate filtering to keep the public from seeing violent visual online content. For real-time monitoring and online content filtering, we have presented a new AVDCS. We came up with this idea after comparing our findings to those of other researchers in the field. Custom loss functions suitable for dealing with the dataset's complexity and diversity could improve the results



(a)

(b)

Figure 9: Some instances of erroneous outputs: (a) false positive and (b) false negative.

Table 3: Performance of BrutNet across MC, HF, MF, AVD and RWF2000 datasets.

Model	Accuracy on Different Datasets			No. of parameters
	HF Dataset	MF Dataset	AVD Dataset	
BrutNet (previous) [31]	-	-	90.00%	5.466 M
ViolenceNet Optical Flow [32]	**99.20%**	**100.00%**	-	4.5 M
ViolenceNet Pseudo-Optical Flow [32]	97.50%	**100.00%**	-	4.5 M
ResNet50+LSTM [30]	83.19%	88.74%	-	-
Efficient 3D CNN [33]	98.30%	**100.00%**	-	7.4 M
Hough Forests + 2D CNN [38]	94.60%	99.00%		-
Xception + BiLSTM + Attention for 10 frames [34]	97.50%	**100.00%**	-	9 M
Motion Blobs + Random Forest [44]	82.40%	96.90%	-	-
C3D [35]	87.40%	93.60%	-	78 M
AlexNet + LSTM [36]	97.10%	**100.00%**	-	9.6 M
motion intensities + AdaBoost [41]	90.10%	98.90%	-	-
ResNet50 + ConvLSTM [42]	89.00%	92.00%	-	-
MobileNetV2 + LSTM [45]	96.10%	99.50%	-	4.074 M
BrutNet (proposed)	97.62%	**100.00%**	**97.22%**	**3.416 M**

even more. The findings show that a higher dataset sample size may also help improve the existing accuracy. Many more areas of development are possible in the future as well.

References

[1] M. Ramzan, A. Abid, H. U. Khan, S. M. Awan, A. Ismail, M. Ahmed, M. Ilyas, A. Mahmood, A review on state-of-the-art violence detection techniques, IEEE Access 7 (2019) 107560–107575.

[2] K. Gkountakos, K. Ioannidis, T. Tsikrika, S. Vrochidis, I. Kompatsiaris, Crowd violence detection from video footage, in: 2021 International Conference on Content-Based Multimedia Indexing (CBMI), IEEE, 2021, pp. 1–4.

[3] A.-M. R. Abdali, R. F. Al-Tuma, Robust real-time violence detection in video using cnn and lstm, in: 2019 2nd Scientific Conference of Computer Sciences (SCCS), IEEE, 2019, pp. 104–108.

[4] S. Liu, T. Forss, New classification models for detecting hate and violence web content, in: 2015 7th international joint conference on knowledge discovery, knowledge engineering and knowledge management (IC3K), Vol. 1, IEEE, 2015, pp. 487–495.

[5] W. Han, M. Ansingkar, Discovery of elsagate: Detection of sparse inappropriate content from kids videos, in: 2020 Zooming Innovation in Consumer Technologies Conference (ZINC), IEEE, 2020, pp. 46–47.

[6] X. Zhao, J. Solé-Casals, B. Li, Z. Huang, A. Wang, J. Cao, T. Tanaka, Q. Zhao, Classification of epileptic ieeg signals by cnn and data augmentation, in: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2020, pp. 926–930.

[7] E. Bermejo Nievas, O. Deniz Suarez, G. Bueno García, R. Sukthankar, Violence detection in video using computer vision techniques, in: International conference on Computer analysis of images and patterns, Springer, 2011, pp. 332–339.

[8] S. Afsha, M. Haque, H. Nyeem, Machine learning models for content classification in film censorship and rating, in: 2022 International Conference on Innovations in Science, Engineering and Technology (ICISET), IEEE, 2022, pp. 396–401.

[9] C. Gu, X. Wu, S. Wang, Violent video detection based on semantic correspondence, IEEE Access 8 (2020) 85958–85967.

[10] J. Mahmoodi, H. Nezamabadi-pour, D. Abbasi-Moghadam, Violence detection in videos using interest frame extraction and 3d convolutional neural network, Multimedia tools and applications (2022) 1–17.

[11] A. Traoré, M. A. Akhloufi, Violence detection in videos using deep recurrent and convolutional neural networks, in: 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), IEEE, 2020, pp. 154–159.

[12] M. Haque, S. Afsha, T. B. Ovi, H. Nyeem, Improving automatic sign language translation with image binarisation and deep learning, in: 2021 5th International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT), IEEE, 2021, pp. 1–5.

[13] A. Ullah, K. Muhammad, W. Ding, V. Palade, I. U. Haq, S. W. Baik, Efficient activity recognition using lightweight cnn and ds-gru network for surveillance applications, Applied Soft Computing 103 (2021) 107102.

[14] P. Do, P. Pham, T. Phan, Some research issues of harmful and violent content filtering for social networks in the context of large-scale and streaming data with apache spark, Recent Advances in Security, Privacy, and Trust for Internet of Things (IoT) and Cyber-Physical Systems (CPS) (2020) 249–272.

[15] A. Khaksar Pour, W. Chaw Seng, S. Palaiahnakote, H. Tahaei, N. B. Anuar, A survey on video content rating: taxonomy, challenges and open issues, Multimedia Tools and Applications 80 (16) (2021) 24121–24145.

[16] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, nature 521 (7553) (2015) 436–444.

[17] P. Wang, P. Wang, E. Fan, Violence detection and face recognition based on deep learning, Pattern Recognition Letters 142 (2021) 20–24.

[18] D. G. C. Roman, G. C. Chávez, Violence detection and localization in surveillance video, in: 2020 33rd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), IEEE, 2020, pp. 248–255.

[19] R. Chatterjee, R. Halder, Discrete wavelet transform for cnn-bilstm-based violence detection, in: Advances in Systems, Control and Automations, Springer, 2021, pp. 41–52.

[20] B. Peixoto, B. Lavi, J. P. P. Martin, S. Avila, Z. Dias, A. Rocha, Toward subjective violence detection in videos, in: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2019, pp. 8276–8280.

[21] M. Gruosso, N. Capece, U. Erra, N. Lopardo, A deep learning approach for the motion picture content rating, in: 2019 10th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), IEEE, 2019, pp. 137–142.

[22] S. Albawi, T. A. Mohammed, S. Al-Zawi, Understanding of a convolutional neural network, in: 2017 international conference on engineering and technology (ICET), IEEE, 2017, pp. 1–6.

[23] R. Fu, Z. Zhang, L. Li, Using lstm and gru neural network methods for traffic flow prediction, in: 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC), IEEE, 2016, pp. 324–328.

[24] M. Cheng, K. Cai, M. Li, Rwf-2000: an open large scale video database for violence detection, in: 2020 25th International Conference on Pattern Recognition (ICPR), IEEE, 2021, pp. 4183–4190.

[25] A. R. M. Guedes, G. C. Chávez, Real-time violence detection in videos using dynamic images, in: 2020 XLVI Latin American Computing Conference (CLEI), IEEE, 2020, pp. 503–511.

[26] S. Das, A. Sarker, T. Mahmud, Violence detection from videos using hog features, in: 2019 4th International Conference on Electrical Information and Communication Technology (EICT), IEEE, 2019, pp. 1–5.

[27] A. Jain, D. K. Vishwakarma, State-of-the-arts violence detection using convnets, in: 2020 International Conference on Communication and Signal Processing (ICCSP), IEEE, 2020, pp. 0813–0817.

[28] N. Honarjoo, A. Abdari, A. Mansouri, Violence detection using pretrained models, in: 2021 5th International Conference on Pattern Recognition and Image Analysis (IPRIA), IEEE, 2021, pp. 1–4.

[29] M. Bianculli, N. Falcionelli, P. Sernani, S. Tomassini, P. Contardo, M. Lombardi, A. F. Dragoni, A dataset for automatic violence detection in videos, Data in brief 33 (2020) 106587.

[30] E. Ditsanthia, L. Pipanmaekaporn, S. Kamonsantiroj, Video representation learning for cctv-based violence detection, in: 2018 3rd Technology

Innovation Management and Engineering Science International Conference (TIMES-iCON), IEEE, 2018, pp. 1–5.
[31] M. Haque, S. Afsha, H. Nyeem, Developing brutnet: A new deep cnn model with gru for realtime violence detection, in: 2022 International Conference on Innovations in Science, Engineering and Technology (ICISET), IEEE, 2022, pp. 390–395.
[32] F. J. Rendón-Segador, J. A. Álvarez-García, F. Enríquez, O. Deniz, Violencenet: Dense multi-head self-attention with bidirectional convolutional lstm for detecting violence, Electronics 10 (13) (2021) 1601.
[33] J. Li, X. Jiang, T. Sun, K. Xu, Efficient violence detection using 3d convolutional neural networks, in: 2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), IEEE, 2019, pp. 1–8.
[34] Ş. Aktı, G. A. Tataroğlu, H. K. Ekenel, Vision-based fight detection from surveillance cameras, in: 2019 Ninth International Conference on Image Processing Theory, Tools and Applications (IPTA), IEEE, 2019, pp. 1–6.
[35] D. Tran, L. Bourdev, R. Fergus, L. Torresani, M. Paluri, Learning spatiotemporal features with 3d convolutional networks, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 4489–4497.
[36] S. Sudhakaran, O. Lanz, Learning to detect violent videos using convolutional long short-term memory, in: 2017 14th IEEE international conference on advanced video and signal based surveillance (AVSS), IEEE, 2017, pp. 1–6.
[37] N. AlDahoul, H. A. Karim, R. Datta, S. Gupta, K. Agrawal, A. Albunni, Convolutional neural network-long short term memory based iot node for violence detection, in: 2021 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET), IEEE, 2021, pp. 1–6.
[38] I. Serrano, O. Deniz, J. L. Espinosa-Aranda, G. Bueno, Fight recognition in video using hough forests and 2d convolutional neural network, IEEE Transactions on Image Processing 27 (10) (2018) 4787–4797.
[39] Z. Dong, J. Qin, Y. Wang, Multi-stream deep networks for person to person violence detection in videos, in: Chinese Conference on Pattern Recognition, Springer, 2016, pp. 517–531.
[40] L. Xu, C. Gong, J. Yang, Q. Wu, L. Yao, Violent video detection based on mosift feature and sparse coding, in: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2014, pp. 3538–3542.
[41] O. Deniz, I. Serrano, G. Bueno, T.-K. Kim, Fast violence detection in video, in: 2014 international conference on computer vision theory and applications (VISAPP), Vol. 2, IEEE, 2014, pp. 478–485.
[42] M. Sharma, R. Baghel, Video surveillance for violence detection using deep learning, in: Advances in data science and management, Springer, 2020, pp. 411–420.
[43] S. U. Khan, I. U. Haq, S. Rho, S. W. Baik, M. Y. Lee, Cover the violence: A novel deep-learning-based approach towards violence-detection in movies, Applied Sciences 9 (22) (2019) 4963.
[44] I. Serrano Gracia, O. Deniz Suarez, G. Bueno Garcia, T.-K. Kim, Fast fight detection, PloS one 10 (4) (2015) e0120448.
[45] R. Vijeikis, V. Raudonis, G. Dervinis, Efficient violence detection in surveillance, Sensors 22 (6) (2022) 2216.