IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 27, 2021, accepted January 25, 2022, date of publication January 28, 2022, date of current version February 14, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3147519

A Deep Learning-Based Approach for Inappropriate Content Detection and Classification of YouTube Videos

KANWAL YOUSAF AND TABASSAM NAWAZ
Department of Software Engineering, University of Engineering and Technology, Taxila 47050, Pakistan
Corresponding author: Kanwal Yousaf (kanwal.yousaf@uettaxila.edu.pk)

ABSTRACT The exponential growth of videos on YouTube has attracted billions of viewers among which the majority belongs to a young demographic. Malicious uploaders also find this platform as an opportunity to spread upsetting visual content, such as using animated cartoon videos to share inappropriate content with children. Therefore, an automatic real-time video content filtering mechanism is highly suggested to be integrated into social media platforms. In this study, a novel deep learning-based architecture is proposed for the detection and classification of inappropriate content in videos. For this, the proposed framework employs an ImageNet pre-trained convolutional neural network (CNN) model known as EfficientNet-B7 to extract video descriptors, which are then fed to bidirectional long short-term memory (BiLSTM) network to learn effective video representations and perform multiclass video classification. An attention mechanism is also integrated after BiLSTM to apply attention probability distribution in the network. These models are evaluated on a manually annotated dataset of 111,156 cartoon clips collected from YouTube videos. Experimental results demonstrated that EfficientNet-BiLSTM (accuracy = 95.66%) performs better than attention mechanism-based EfficientNet-BiLSTM (accuracy = 95.30%) framework. Secondly, the traditional machine learning classifiers perform relatively poor than deep learning classifiers. Overall, the architecture of EfficientNet and BiLSTM with 128 hidden units yielded state-of-the-art performance (f1 score = 0.9267). Furthermore, the performance comparison against existing state-of-the-art approaches verified that BiLSTM on top of CNN captures better contextual information of video descriptors in network architecture, and hence achieved better results in child inappropriate video content detection and classification.

INDEX TERMS Deep learning, social media analysis, video classification, bidirectional LSTM, CNN, EfficientNet.

I. INTRODUCTION

The creation and consumption of videos on social media platforms have grown drastically over the past few years. Among the social media sites, YouTube predominates as a video sharing platform with plethora of videos from diverse categories. According to YouTube statistics [1], the global user base of YouTube is over 2 billion registered users and more than 500 hours of video content is uploaded every minute. Consequently, billions of hours of videos are available where users of all age groups can explore generic as well as personalized content [2]. Considering such a large-scale crowdsourced database, it is extremely challenging to monitor and regulate the uploaded content as per platform guidelines. This creates opportunities for malicious users to indulge in spamming activities by misleading the audiences with falsely advertised content (i.e., video, audio or text). The most disruptive behavior by malicious users is to expose the young audiences to disturbing content, particularly when it is fabricated as safe for them. Children today spend most of their time on the Internet and the YouTube platform for them has distinctly established itself as an alternative to traditional screen media (e.g., television) [3], [4]. The YouTube press release [5] also confirmed the high popularity of this social media site among younger audiences compared to other age groups, and the reason for this high level of approval is due to fewer restrictions [6].

The associate editor coordinating the review of this manuscript and approving it for publication was Aasia Khanum.

Unlike television, children can be presented with any type of content on the Internet due to lack of regulations. Exposing children to disturbing content is considered as one among other internet safety threats (like cyberbullying, cyber predators, hate etc.) [7]. Bushman and Huesmann [8] confirmed that frequent exposure to disturbing video content may have a short-term or long-term impact on children's behavior, emotions and cognition. Many reports [9]–[12] identified the trend of distributing inappropriate content in children's videos. This trend got people's attention when mainstream media reported about the Elsagate controversy [13], [14], where such video material was found on YouTube featuring famous childhood cartoon characters (i.e., Disney characters, superheroes, etc.) portrayed in disturbing scenes; for instance, performing mild violence, stealing, drinking alcohol and involving in nudity or sexual activities.

In an attempt to provide a safe online platform, laws like the children's online privacy protection act (COPPA) imposes certain requirements on websites to adopt safety mechanisms for children under the age of 13. YouTube has also included a ''safety mode'' option to filter out unsafe content. Apart from that, YouTube developed the YouTube Kids application to allow parental control over videos that are approved as safe for a certain age group of children [15]. Regardless of YouTube's efforts in controlling the unsafe content phenomena, disturbing videos still appear [16]–[19] even in YouTube Kids [20] due to difficulty in identifying such content. An explanation for this may be that the rate at which videos are uploaded every minute makes YouTube vulnerable to unwanted content. Besides, the decision-making algorithms of YouTube rely heavily on the metadata of video (i.e., video title, video description, view count, rating, tags, comments, and community flags). Hence, filtering videos based on the metadata and community flagging is not sufficient to assure the safety of children [21]. Many cases exist on YouTube where safe video titles and thumbnails are used for disturbing content to trick children and their parents. The sparse inclusion of child inappropriate content in videos is another common technique followed by malicious uploaders. Fig. 1 displays an example among such cases where video title and video clips are safe for children (as shown in Fig. 1(a)) but included inappropriate scenes in this video (as shown in Fig. 1(b) and Fig. 1(c)). The concerning thing about this example, including many similar cases, is that these videos have millions of views with more likes than dislikes, and have been available for years. Many other cases (as shown in Fig. 1(d)) also identified where videos or the YouTube channel is not popular, yet contains child unsafe content especially in the form of animated cartoons. It is evident from examples that this problem persists irrespective of channel or video popularity. Furthermore, YouTube has disabled the dislike feature of videos which resulted in viewers being incapable of getting the indirect video content feedback from statistics. Since the YouTube metadata can be easily manipulated, it is suggested to better use video features for detection of inappropriate content than metadata features associated with videos [22].

Prior techniques [23]–[28] addressed the challenge of identifying disturbing content (i.e., violence, pornography, etc.) from videos by using traditional hand-crafted features on frame-level data. In recent years, the state-of-the-art performance of deep learning has motivated researchers to employ it in image and video processing. The most frequent applications of image/video classification employed the convolutional neural networks [29]–[31]. Apart from that, the long-short term memory (LSTM), a special type of recurrent neural network (RNN) architecture, has proven to be an effective deep learning model in time-series data analysis [32]. Hence, this study targets the YouTube multiclass video classification problem by leveraging CNN (EfficientNet-B7) and LSTM to learn video effective representations for detection and classification of inappropriate content. We targeted two types of objectionable content geared towards young viewers, one, which contains violence and the second, which includes sexual nudity connotations. The main contributions of this study are threefold:

1. We propose a novel CNN (EfficientNet-B7) and BiLSTM-based deep learning framework for inappropriate video content detection and classification.
2. We present a manually annotated ground truth video dataset of 1860 minutes (111,561 seconds) of cartoon videos for young children (under the age of 13). All videos are collected from YouTube using famous cartoon names as search keywords. Each video clip is annotated for either safe or unsafe class. For the unsafe category, fantasy violence and sexual-nudity explicit content are monitored in videos. We also intend to make this dataset publicly available for the research community.
3. We evaluate the performance of our proposed CNN-BiLSTM framework. Our multiclass video classifier achieved the validation accuracy of 95.66%. Several other state-of-the-art machine learning and deep learning architectures are also evaluated and compared for the task of inappropriate video content detection.

To summarize, this work can assist any video sharing platform to either remove unsafe video or blur/hide any portion of video involving unsafe content. Secondly, it may also help in the development of parental control solutions on the web via plugins or browser extensions where children inappropriate content filters automatically. The upcoming sections of the article are outlined as follows: Section II covers the related work in this research area. The methodology of our proposed system is explained in Section III. The experimental setup of the proposed system is presented in Section IV. The results obtained from the experimental setup are analyzed and discussed in Section V, and finally, Section VI concludes the work and directs some future scope for improvements.

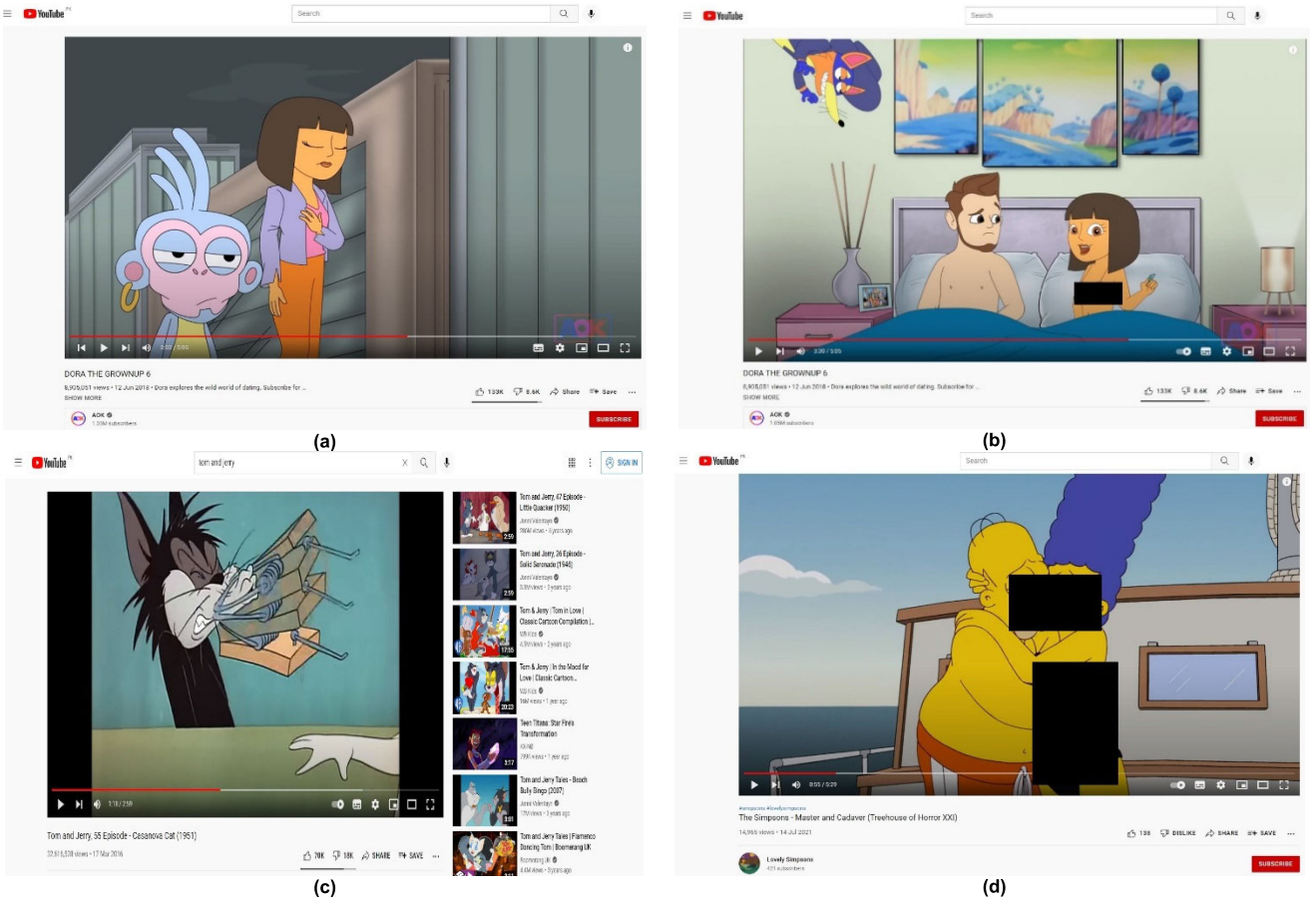


FIGURE 1. Examples of YouTube cartoon videos depicting different scenarios (a) video showing safe content with 8.9 million views, uploaded since 2018 by AOK channel of 1.05 million subscribers, (b) video showing sexual-nudity content with 8.9 million views, uploaded since 2018 by AOK channel of 1.05 million subscribers, (c) video showing fantasy violence content with 32 million views, uploaded since 2016 by WB Kids channel of 19.9 million subscribers, and (d) video showing sexual-nudity content with 14.6k views, uploaded since 2021 by Lovely Simpsons channel of 421 subscribers.

II. RELATED WORK

The explosion of multimedia data on YouTube has presented a lot of opportunities for researchers [33]. However, the challenging task is to find an optimal technique to understand the context of videos. In both computer vision and machine learning, video classification is studied extensively as one of the fundamental approaches for video understanding [34]. The problem of detecting inappropriate video falls in the category of video classification or event detection problem.

A. MACHINE LEARNING METHOD

Most of the earlier studies used hand-crafted features on images or video-level data in identifying the discriminative patterns of inappropriate content. The skin (i.e., skin color) [35] and motion information based features were used for nudity or pornography detection [26], [36]. The multimodal approach was also followed by fusing different modalities of data (i.e., audio, video) with skin and motion-based features. Rea *et al.* [37] proposed a periodicity-based audio feature extraction method which was later combined with visual features for illicit content detection in videos.

The machine learning algorithms are usually employed as classifiers Liu *et al.* [38] classified the periodicity-based audio and visual segmentation features through support vector machine (SVM) algorithm with Gaussian radial basis function (RBF) kernel. Later on, they extended the framework [39] by applying the energy envelope (EE) and bag-of-words (BoW)-based audio representations and visual features. Ulges *et al.* [23] used MPEG motion vectors and Mel-frequency cepstral coefficient (MFCC) audio features with skin color and visual words. Each feature representation is processed through an individual SVM classifier and combined in a weighted sum of late fusion. Ochoa *et al.* [40] performed binary video genre classification for adult content detection by processing the spatiotemporal features with two types of SVM algorithms: sequential minimal optimization (SMO) and LibSVM. Jung *et al.* [41] worked with the one-dimensional signal of spatiotemporal motion trajectory and skin color. Tang *et al.* [42] proposed a pornography detection system—PornProbe, based on a hierarchical latent Dirichlet allocation (LDA) and SVM algorithm. This system combined an unsupervised clustering in LDA and supervised

learning in SVM, and achieved high efficiency than a single SVM classifier. Lee *et al.* [43] presented a multilevel hierarchical framework by taking the multiple features of different temporal domains. Lopes *et al.* [44] worked with the bag-of-visual features (BoVF) for obscenity detection. Kaushal *et al.* [21] performed supervised learning to identify the child unsafe content and content uploaders by feeding the machine learning classifiers (i.e., random forest, K-nearest neighbor, and decision tree) with video-level, user-level and comment-level metadata of YouTube Reddy *et al.* [45] handled the explicit content problem of videos through text classification of YouTube comments. They applied bigram collocation and fed the features to the naïve Bayes classifier for final classification.

B. DEEP LEARNING METHOD

In contrast to machine learning algorithms, there is a growing trend of using deep learning architectures to learn the video-based feature representations in video classification. The study of Ngiam *et al.* [46] is the first deep learning-based research that processed different data modalities such as video, image, audio and text, and performed greedy layer-wise training of restricted Boltzmann machine (RBM) model. Karpathy *et al.* [29] showed multiclass video classification results on a large-scale video dataset (ImageNet) by using the convolutional neural network. This study is followed by the research of Yue-Hei Ng *et al.* [32] in which information across the full-length duration of videos is examined. The LSTM model is employed on top of frame-level CNN activations for better video classification. Some other studies have also taken the benefits of CNN-LSTM model to capture the context of a given sequence of video frames [47]. Wu *et al.* [31] presented the CNN-LSTM hybrid model for obtaining the short-term spatial-motion patterns through CNNs and long-term temporal clues by employing LSTMs Simonyan and Zisserman [30] built a two-stream CNN architecture for action detection from videos. Wehrmann *et al.* [48] reported the best accuracy results with CNN-LSTM than other baselines on the NPDI pornography dataset [24]. Perez *et al.* [49] classified pornography videos by deploying CNNs with static and motion information (i.e., MPEG motion vectors). This study yielded the same accuracy scores when static and motion information are combined by late fusion or mid-level fusion (96.4%), but the performance degraded in early fusion (90.5%) Aldahoul *et al.* [50] explored and compared different state-of-the-art deep learning approaches and found that the pre-trained CNN model (EfficientNet-B7) based features with SVM classifier performed better in unsafe video detection.

In the literature, many studies explored different YouTube data modalities (i.e., text, audio, image, and video) individually for inappropriate content detection. Yenala *et al.* [51] proposed a novel CNN-BiLSTM model for automatic detection and filtering of the inappropriate text in query suggestions of YouTube search. Trana *et al.* [52] investigated naïve Bayes, SVM and CNN classifiers for harassment detection in YouTube comments Dadvar and Eckert [53] identified cyberbullying in YouTube comments by experimenting with four deep learning models (i.e., CNN, LSTM, BiLSTM and BiLSTM with attention) using GloVe, SSWE and random word embeddings. In this study, the BiLSTM with an attention mechanism performed better than conventional machine learning algorithms. Mohaouchane *et al.* [54] performed an automatic detection of YouTube offensive comments using the CNN-BiLSTM model. Alshamrani [55] used an ensemble classification model to detect the age-inappropriate comments posted on YouTube videos. Later on, they extended the work [17] by leveraging natural language processing in an ensemble classifier. They detect five age-inappropriate remarks (toxic, absence, insult, threat and identity hate) which appear in YouTube comments Alshamrani *et al.* [56] applied a neural network model on YouTube comments for toxicity detection and an LDA model on video captions for topic modeling.

The combination of different YouTube modalities is also explored in deep learning architectures. Mariconti *et al.* [57] showed an ensemble method for the detection of proactive remarks in YouTube videos. Features from three different sources (video metadata, audio transcripts and thumbnail) are extracted and fed to individual classifiers such as metadata to SVM with linear kernel, thumbnails to the random forest and audio transcripts to RNN-based gated recurrent unit (GRU) classifier Hou *et al.* [10] recognized the bloody videos by combining the audio-visual features and passed them to the CNN-LSTM model. Ali and Senan [11] also explored the audio-visual features of YouTube videos with the DNN model for violence classification. Sumon *et al.* [58] proposed the CNN-LSTM model for identifying the violent crowd flow in YouTube videos Alghowinem [59] showed a multimodal approach for inappropriate content filtering of the YouTube Kids application. Papadamou *et al.* [20] studied an Elsagate phenomenon in the YouTube Kids application by using YouTube video metadata such as video title, tags, statistics (likes/dislikes, view count, comments count), style features (same as proposed in [21]) and thumbnails. The authors deployed a fully connected neural network model for statistics and style features, CNN for thumbnail features, and LSTMs for video title and tags features. Vitorino *et al.* [60] explored transfer learning and fine-tuning with CNN for sexually exploitative imagery of children (SEIC) detection. Ishikawa *et al.* [61] proposed an end-to-end pipeline for combating the Elsagate content in YouTube videos. They evaluated the classification performances of mobile-platform based different deep learning architectures such as GoogLeNet, SqueezNet, NASNet, and MobileNetV2 Tahir *et al.* [22] processed multimodal data with the CNN-LSTM framework for disturbing and fake embedded content in videos. Lastly, Singh *et al.* [62] performed a fine-grained approach for child unsafe video content detection. They employed the LSTM autoencoder to learn video representations from CNN (VGG-16) feature descriptors.

The abundance of literature is available for inappropriate video content detection by using hand-crafted or deep learning feature extraction techniques. These studies have performed binary classification to classify the videos as safe or unsafe. However, it lacks the research of detecting or classifying different categories of disturbing content in real-time YouTube videos, particularly in children-oriented videos. Secondly, the existing studies have not explored the BiLSTM network for inappropriate video content detection. This paper addresses the aforementioned problems by working with EfficientNet-B7 and BiLSTM-based deep learning framework for children inappropriate video content detection and classification.

III. PROPOSED METHODOLOGY

The proposed methodology provides a system to resolve the problem of disturbing content in videos. This work employs deep learning architecture which has already been applied successfully in several applications for video classification problems. As shown in Fig. 2, the proposed system is divided into three main modules, namely (1) video preprocessing, (2) deep feature extraction, and (3) video representation and classification. In the video preprocessing stage, the collected YouTube videos are preprocessed to remove all irrelevant or missing video information. It also rescales the extracted frames of each video clip into fixed dimensions (224 × 224). The preprocessed video frames of each video clip are forwarded as an input to an ImageNet pre-trained EfficientNet-B7 model for feature extraction. The extracted features are experimented with the BiLSTM network to learn effective video representations, which subsequently passed to the fully connected and softmax layers for final video classification. Furthermore, the detailed descriptions of each step are explained in the following subsections.

A. VIDEO PREPROCESSING

Video preprocessing plays an important role in deep learning techniques, as it helps in acquiring the relevant features for better video classification. In this work, a video (V_i) is first represented as N small segments of videos referred to as *video clips* (c_1^i, c_2^i, …., c_N^i) with one-second length each. These video clips are labeled through a manual annotation process where all clips with incomplete information or video context are ignored. After splitting and labeling of video clips, it is noticed that the initial 3-4 frames of each video clip have a piece of information from the previous clip of the same video. Therefore, considering an average video frame rate of 23-24 frames per second (fps), each of the j^{th} video clip (c_j^i) is sampled at the frame rate of 22 fps by ignoring some initial frames. The last frame is duplicated for all video clips containing fewer frames than an average frame rate of a video. Overall, the frames are represented as $f_1^{i,j}, f_2^{i,j}, \ldots, f_{22}^{i,j}$, where $f_k^{i,j}$ means k^{th} frame of video clip c_j^i. Finally, the selected frames of each video clip are rescaled to a fixed resolution of 224 × 224 pixels that correspond to the input size of the pre-trained convolutional neural network model.

B. DEEP FEATURE EXTRACTION

In this module, the deep features of preprocessed video frames are extracted by using a deep learning model with advanced architecture. Instead of training an entire CNN model from scratch, this study employed a pre-trained CNN architecture known as EfficientNet for the extraction of visual representations from video frames.

1) EFFICIENT-NET

The EfficientNet model is a convolutional neural network model and scaling method that uniformly scales network depth, width and resolution through compound coefficient. It is trained on a large-scale ImageNet dataset with 1.3 million images from 1000 object classes [63]. Tan and Le [64] reported state-of-the-art accuracy of EfficientNet on ImageNet dataset with much smaller and faster inference than best existing CNN models. The baseline network of EfficientNet is referred to as B0, whereas other scaling networks include B1, B2, B3, B4, B5, B6 and B7 respectively. In general, all scaling networks show better accuracy but with the cost of FLOPS.

The proposed framework included EfficientNet-B7 by working with preprocessed extracted frames ($f_1^{i,j}, f_2^{i,j}, \ldots, f_{22}^{i,j}$) of each video clip c_j^i as an input. The EfficientNet module performed feature extraction with the transfer learning technique in which each input frame with dimensions of $224 \times 224 \times 3$ (image_width x image_height x RGB_channel) is processed through a stack of 813 layers. The last three layers including the fully connected layer generating 1000 ImageNet output labels are discarded, which makes the EfficientNet-B7 to generate an output of feature descriptors $X_k^{i,j}$ with $7 \times 7 \times 2560$ dimensions for each frame $f_k^{i,j}$. These feature descriptors are used as an input to the BiLSTM model for video representation and classification.

C. VIDEO REPRESENTATION AND CLASSIFICATION

The third stage of pipeline trained bidirectional LSTM network with supervised learning so that effective video representations can be learnt from feature descriptors of video clips. Subsequently, the proposed system is added with two fully connected layers for acquiring the final video classification results.

1) BiLSTM NETWORK

The recurrent neural networks produce good network performance in modeling the hidden sequential patterns of time-series data. However, the vanishing gradient problem hampers an update of network parameters during the back-propagation process. It is usually resolved by using two variations of RNNs, which are: LSTM and gated recurrent unit (GRU). Conceptually, the network structure of LSTM is as same as RNN, but a special unit ''memory cell'' is introduced in LSTM to replace the update process of RNN. The memory cell of LSTM maintains information for a longer duration of time. Considering the current input vector x_t, the last hidden

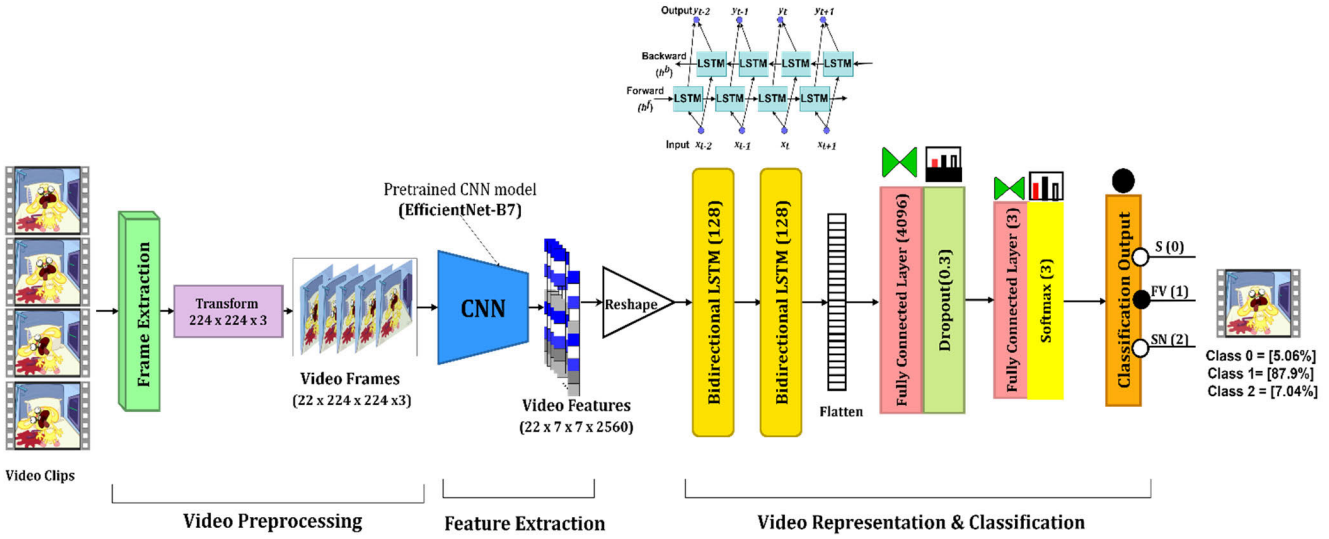


FIGURE 2. The proposed framework architecture works into three stages for children inappropriate video content detection and classification. In the first stage, video clips are preprocessed to discard irrelevant video frames and transform the selected frames into fixed-size images (224,224,3). Next, the frames are fed to EfficientNet-B7 to get feature vectors. All feature vectors are reshaped and passed into the two-layer stack of BiLSTMs for video representation. A fully connected layer followed by an output layer with softmax activation is integrated to return the probabilities of each video clip against the three classes including safe (class 0), fantasy violence (class 1) and sexual-nudity (class 2).

state h_{t-1}, and the last memory cell state c_{t-1}, the following equations are used to implement an LSTM model:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (1)$$
$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \quad (2)$$
$$c_t = f_t c_{t-1} + i_t tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (3)$$
$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \quad (4)$$
$$h_t = o_t.\tanh(c_t) \quad (5)$$

where, σ represents the sigmoid activation function; $i, f, c, and\ o$ denote input gate, forget gate, memory cell state and output gate at time t, respectively. $W\ and\ b$ denote the weights and bias vector. Considering the video classification problem, one potential drawback of LSTM is that it captures the past context only. For getting the full context of any video, it is important to consider both directions i.e., past and future context of the video. Therefore, the bidirectional LSTM appears to be a suitable option in video classification as it preserves the information in both directions, as shown in Fig. 3.

In BiLSTM, there are two distinct hidden layers referred to as forward hidden layer (h_t^f) and backward hidden layer (h_t^b). The forward hidden layer h_t^f considers input vector x_t in ascending order i.e., $t = 1, 2, 3, \ldots, T$, and backward hidden layer h_t^b in descending order i.e., $t = T, T-1, T-2, \ldots, 1$. Lastly, the output y_t is generated by combining the results of h_t^f and h_t^b. Following equations are used to implement the BiLSTM model:

$$h_t^f = tanh(W_{xh}^f x_t + W_{hh}^f h_{t-1}^f + b_h^f) \quad (6)$$
$$h_t^b = tanh(W_{xh}^b x_t + W_{hh}^b h_{t+1}^b + b_h^b) \quad (7)$$
$$y_t = W_{hy}^f h_t^f + W_{hh}^b h_t^b + b_y \quad (8)$$

It is noticed that adding an excessive number of layers of BiLSTM increases the network complexity and slows down the training process. Hence, this work employed two layers of BiLSTM to understand the video representations.

2) BiLSTM NETWORK WITH AN ATTENTION MECHANISM

A neural network architecture with an attention mechanism decides when to look into data (or in this case, segments of videos) by automatically giving a high level of focus to feature vectors with most valuable information than the feature vectors with less valuable information. The architecture of BiLSTM with an attention mechanism is depicted in Fig. 4.

Considering the final hidden state of i-th BiLSTM as h_{it}, which is computed as:

$$h_{it} = [h_t^f, h_t^b] \quad (9)$$

Then, the attention mechanism is computed by using the following equations:

$$e_{it} = tanh(W_a h_{it} + b_a) \quad (10)$$
$$a_{it} = \frac{exp(e_{it})}{\sum_{j=1}^{T} exp(e_{jt})} \quad (11)$$
$$v_t = \sum_{t=1}^{T} a_{it}.h_{it} \quad (12)$$

The attention mechanism assigns attention weight a_{it} to the i-th BiLSTM output vector at time t, as calculated in equation 11. W_a and b_a represents the weight and bias from the attention layer. Finally, the output from attention layers generates an attention vector v_t, which is calculated as a weighted sum of the multiplication between attention weight a_{it} and i-th BiLSTM output vector at time t.

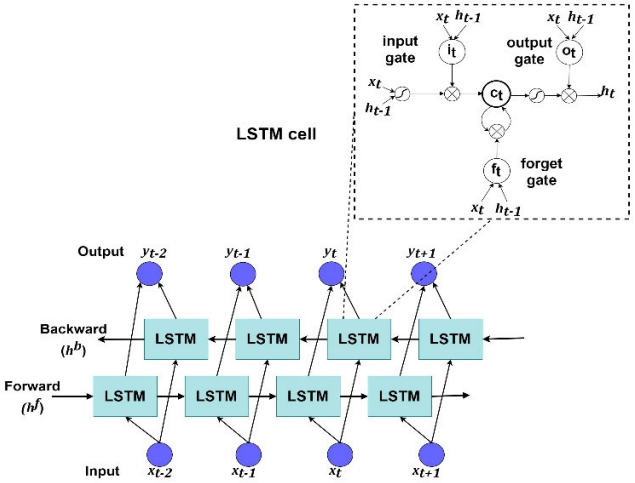


FIGURE 3. The architecture of the BiLSTM model.

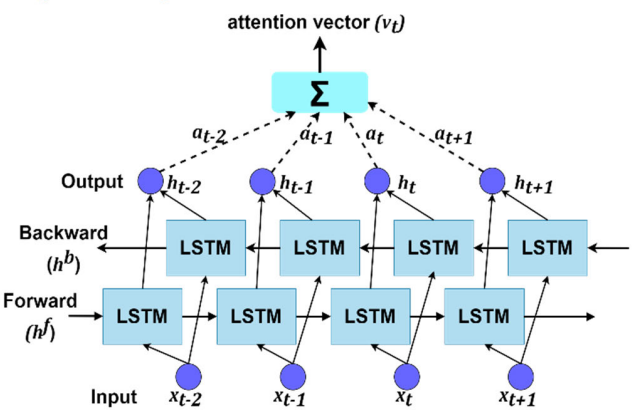


FIGURE 4. The architecture of the BiLSTM model with an attention mechanism.

3) SOFTMAX CLASSIFIER

The softmax activation function in an output layer of any deep learning model is considered as a softmax classifier. To classify each video clip into one of three classes (i.e., fantasy violence, sexual-nudity and safe), the proposed model integrated a softmax activation function in the last fully connected layer to determine the relative probability of three output units. The softmax activation function (σ) is calculated as:

$$\sigma(z_i) = \frac{\exp(z_i)}{\sum_{c=0}^{N-1} \exp(z_c)} \tag{13}$$

D. OVERALL PROPOSED FRAMEWORK

The general architecture of the proposed framework for multiclass video classification is illustrated in Fig. 2. The input is a sequence of 22 frames of fixed resolution with $224 \times 224 \times 3$ pixels. These frames are processed through EfficientNet-B7 for extracting features and generating the feature vector of $22 \times 7 \times 7 \times 2560$ shape. The high-level features are reshaped and directed towards the two-layer stack of BiLSTM network. The flattening layer is added to transform the feature representations into a 1-dimensional vector. Subsequently, a fully connected layer (or dense layer) of 4096 neurons with rectified linear unit (ReLU) activation function is added. As a fully connected layer generates a wide range of probabilities by connecting all inputs of one layer to every activation unit of the next layer; therefore, a dropout of 0.3 is carried out to prevent this model from an overfitting problem. Finally, the softmax output layer with 3 neurons gives the final classification scores. The algorithmic steps for training and validation of the proposed framework are presented in Algorithm 1.

IV. EXPERIMENTAL SETUP

A. DATASET

YouTube has a huge collection of videos and metadata of videos (i.e., likes, dislikes, view count, comments, etc.) that has been explored in numerous researches. Google released the YouTube-8M benchmark dataset of more than 8 million video IDs with corresponding labels from 4716 classes [65] Apart from it, there exists other video benchmarks of specific categories like sports (Sports-1M [29], UCF-101 [66]), action recognition (HMDB51 [67], Kinetics [68]), face recognition (YTF [69], YouTube Celebrities [70]), sentiment analysis ([71]), and video captioning (MSVD [72], MSR-VTT [73]). However, none of these existing benchmarks aims for the proposed video classification problem. The datasets closely related to our problem are the NPDI cartoon dataset [50], the Elsagate dataset [61], and the dataset of Singh *et al.* [62]. Comparatively, the NPDI dataset is the smallest with 900 images only and is not suitable to perform our deep learning-based video classification task. The Elsagate dataset is a publicly available dataset of cartoon videos from sensitive and non-sensitive classes. In this dataset, whole video is considered either safe or unsafe where the clean frames of video are also labeled as unsafe. Secondly, it lacks the complex behaviors of sensitivity content. The videos in this dataset are targeted for toddlers. Lastly, the dataset of Singh *et al.* [62] included Japanese anime videos dedicated for mature viewers. For this reason, a manually annotated video dataset is presented for identifying the disturbing content. Because the intention is to focus on content for children, this study included cartoon videos only. The videos are searched and collected using the four popular cartoon names (including Tom and Jerry, gravity falls, Simpsons and Sponge Bob) and "cartoon" as keywords through YouTube Data API. Once the list of videos is obtained and downloaded, next step involved video filtering in which all irrelevant videos (like non-cartoon) are discarded. The process of filtering resulted into 1126 videos with duration range between 2 to 600 seconds long. All collected videos are split into one-second duration clips using FFmpeg. Each video clip is manually annotated as belonging to either safe, fantasy violence or sexual-nudity class. The clips with any other act (i.e., extreme bloodshed or violence, smoking, use of drugs, frightening or horror scenes, etc.) are not included in this dataset.

Algorithm 1 Training and Validation of the EfficientNet-BiLSTM Video Classification Algorithm

Input: Training set $T = \{(c_i, \theta_i), i = 1, 2, 3, \ldots, n\}$, Validation set $V = \{(c'_i, \theta'_i), i = 1, 2, 3, \ldots, n\}$, $epoch\_num$ = number of epochs

Output: Trained model $Model_{epoch}$, Accuracy $accuracy_{epoch}$, Precision $precision_{epoch}$, Recall $recall_{epoch}$, F1 Score $f1 - score_{epoch}$

Basic Idea:

1. Divide the training instances T into k mini-batches t_k each containing a fixed number of unique video clips c_i such that $T = \{(t_k, \theta_k), c_i \in t_k, k = 1, 2, 3, \ldots, m, i = 1, 2, \ldots, n_1\}$, where $t_1 \cap t_2 \ldots \cap t_m = \emptyset$.
2. Extract features from each mini-batch t_k, where $k = 1, 2, 3, \ldots, m$.
3. Train model with extracted features of mini-batch t_k and assigned labels θ_k.
4. For validation, apply the same procedure as STEP 1 such that validation instances $V = \{(v_k, \theta'_k), c'_i \in v_k, k = 1, 2, 3, \ldots, m, i = 1, 2, \ldots, n_2\}$, where $v_1 \cap v_2 \ldots \cap v_m = \emptyset$.
5. Validate the model $Model_{epoch}$.
6. Calculate accuracy, precision, recall and F1 score using the confusion matrix $conf\_matrix_{epoch}$.

for $epoch \leftarrow 0$ *to* $epoch\_num$ **do**
  for all $t_k \epsilon T$ $(k = 1, 2, 3, \ldots, m)$ **do**
    $f_{t_k} = [], \theta_{t_k} = []$
    for all $c_i \epsilon t_k$ $(i = 1, 2, \ldots, n_1)$ **do**
      $f'_{c_i,t_k} \leftarrow feature\_extraction(c_i)$
      $f_{t_k}.append(f'_{c_i,t_k})$
      $\theta_{t_k}.append(\theta_{c_i,t_k})$
    end for
    $X_{t_k} \leftarrow np.array(f_{t_k})$
    $Y_{t_k} \leftarrow Transform(\theta_{t_k})$// `transform labels into one-hot encoder`
    $Model_{epoch} \leftarrow Train\_Classifier(X_{t_k}, Y_{t_k})$
  end for
  for all $v_k \epsilon V$ $(k = 1, 2, 3, \ldots, m)$ **do**
    $f_{v_k} = [], \theta_{v_k} = [], result_{epoch} = []$
    for all $c'_i \epsilon v'_k$ $(i = 1, 2, \ldots, n_2)$ **do**
      $f'_{c'_i,v_k} \leftarrow feature\_extraction(c'_i)$
      $f_{v_k}.append(f'_{c'_i,v_k})$
      $\theta_{v_k}.append(\theta'_{c'_i,v_k})$
    end for
    $X_{v_k} \leftarrow np.array(f_{v_k})$
    $Y_{v_k\_pred} \leftarrow Predict(X_{v_k})$
    $result_{epoch}.append(epoch, [Y_{v_{k_{pred}}} = Inverse\_Transform(Y_{v_k\_pred})], [Y_{v_{k_{actual}}} = \theta_{v_k}])$
  end for
  $conf\_matrix_{epoch} = Confusion\_Matrix(result_{epoch}.predicted, result_{epoch}.actual)$
  $accuracy_{epoch} = \frac{1}{N} \sum_{c=0}^{N-1} \frac{conf\_matrix_{epoch}(TP_c + TN_c)}{conf\_matrix_{epoch}(TP_c + TN_c + FP_c + FN_c)}$
  $precision_{epoch} = \frac{1}{N} \sum_{c=0}^{N-1} \frac{conf\_matrix_{epoch}(TP_c)}{conf\_matrix_{epoch}(TP_c + FP_c)}$
  $recall_{epoch} = \frac{1}{N} \sum_{c=0}^{N-1} \frac{conf\_matrix_{epoch}(TP_c)}{conf\_matrix_{epoch}(TP_c + FN_c)}$
  $f1 - score_{epoch} = \frac{1}{N} \sum_{class=0}^{N-1} (2 * \frac{precision_{class} * recall_{class}}{precision_{class} + recall_{class}})$
end for
return $Model_{epoch}, accuracy_{epoch}, precision_{epoch}, recall_{epoch}, f1 - score_{epoch}$

The manual annotation process results in total of 111,561 video clips including 57908 clips belonging to safe class, 27003 clips in sexual-nudity class, and 26650 clips in fantasy violence class. Overall, there is a balanced distribution of safe

and unsafe video clips (1.08:1). We also intend to make this dataset publicly available for research community. Table 1 summarizes the overall distribution of manually annotated cartoon videos according to three classes.

B. NETWORK HYPERPARAMETERS TUNING

1) FRAME SAMPLING

Due to frame overlap issue, each video clip is sampled at a frame rate of 22 fps by ignoring some of the starting frames. Video clips containing frames less than an average frame rate (23-24 fps) are padded with the last frame of same clip. The frame sampling rate of 22 fps is adopted in all training, validation and testing experiments of neural network models.

2) BILSTM PARAMETER

Considering an ImageNet pre-trained CNN model is employed for video frames feature extraction by using transfer learning approach, the hyperparameters tuning in proposed framework is required for BiLSTM and subsequent fully connected (dense) layer. Various design choices of bidirectional LSTM layers are evaluated, but it is confirmed through experiments that adding two simultaneous layers of bidirectional LSTMs perform better in video classification than working with single or multiple layers of bidirectional LSTM networks. Apart from that, the different number of hidden units (i.e., 64, 128, 256, 512) in each bidirectional LSTM layer are embedded. For consistency, the same number of hidden units are used in both bidirectional layers. Experiments showed that the highest validation accuracy is achieved by using 128 hidden units in two BiLSTM network layers. It is also noticed that the fully connected (FC) layer of 4096 units with ReLU activation function helps in selecting the most relevant and appropriate labels in classification layer than a fully connected layer of 2048 units with same activation function. To avoid the problem of overfitting, the dropout layer (value = 0.3) is added before the last fully connected output layer (dense layer). Finally, the softmax classifier is applied in an output layer of 3 units to get the final probability scores.

Table 2 enlists the complete layer configuration of our best-proposed model used in inappropriate video content detection and classification. This model has nine layers with each containing a different number of output size and learnable parameters. Overall, the proposed model is trained with 152 million parameters (number of neurons) that are updated during the backpropagation process. All the parameters of the pre-trained EfficientNet-B7 model are non-trainable parameters which means that these parameters are not optimized during model training.

3) COST FUNCTION AND OPTIMIZER PARAMETERS

The cost function measures an error between predicted and actual values. The optimizer function is responsible for reducing an error or overall loss of neural network model to improve the model accuracy. In this study, the categorical cross-entropy loss function is used for multiclass video classification, which is calculated as:

$$L_{cross\_entropy}(\hat{y}, y) = -\sum_{c=0}^{N-1} y_{i,c} * \log(\hat{y}_{i,c}) \tag{14}$$

In equation (14), c represents a particular class index from N number of classes, $y_{i,c}$ is the binary indicator (0 or 1) which indicates whether c is the actual class for instance i, $\hat{y}_{i,c}$ represents predicted probability of instance i for class c. Adam optimizer, the extension of stochastic gradient descent (SGD) algorithm, is used with a learning rate of 1e-5 to minimize the error of cost function in proposed model.

TABLE 1. Cartoon dataset distribution.

Category	Classes	Assigned Labels	# Clips	Safe/Unsafe distribution
Cartoon	Safe (S)	0	57908	57908
	Fantasy Violence (FV)	1	26650	53653
	Sexual-Nudity (SN)	2	27003	
Total video clips				111,561

4) TRAINING PARAMETER

Because of memory and computational constraints, it is found convenient to load the training dataset in memory by taking the small subsets of data for model training. In this research, a subset of 1000 samples (or video clips) of training datasets are processed simultaneously in each iteration of one epoch. Training of the proposed EfficientNet-BiLSTM model is performed through the mini-batch gradient descent optimization method, which further divides each subset of the training dataset into n number of mini-batches. After performing experiments using different mini-batch sizes (i.e., 8, 16 and 32), it is observed that the mini-batch size of 16 converges faster and performs better in model accuracy than other batch sizes. Overall, the weights of the proposed EfficientNet-BiLSTM model are updated in 90 iterations per epoch. Each iteration processed the chunk of training dataset in 63 batches (mini-batch size = 16).

C. EXPERIMENTAL ENVIRONMENT

The proposed model is implemented in Python programming using Keras, which is a high-level deep learning application programming interface (API) that runs on top of Google's TensorFlow open-source library [74], [75]. The Google Colab integrated development environment (IDE) is used which offers a cloud-based Jupyter notebook to write and run the Python codes for deep learning architectures. For all experiments, the Google Colab Pro+ is used to train the model, which offers 50 GB of T4 or P100 GPU and 255 GB of disk. Moreover, the Google drive with 2 TB of space is used to store the YouTube cartoon dataset. This dataset is partitioned

TABLE 2. Layer configuration of proposed EfficientNet-BiLSTM model for video classification.

Layers	Type	Output Size	Parameters
Layer 1	Input Layer	22 x 224 x 224 x 3	0
Layer 2	EfficientNet-B7 Layer	22 x 7 x 7 x 2560	64097687
Layer 3	Reshape	22 x 125440	0
Layer 4	Bidirectional LSTM (unit = 128)	22 x 256	128582656
Layer 5	Bidirectional LSTM (unit = 128)	22 x 256	394240
Layer 6	Flatten	5632	0
Layer 7	Fully connected (Dense)	4096	23072768
Layer 8	Dropout (value = 0.3)	4096	0
Layer 9	Fully connected (Softmax)	3	12291
Total parameters: 216,159,642			
Trainable parameters: 152,061,955			
Non-trainable parameters: 64,097,687			

with an 80:20 split such that 80% of the data is allocated for training and 20% for evaluation and testing of models.

D. EVALUATION METRICS

The performances of multiclass video classification models are evaluated by calculating the accuracy, precision, recall and f1 score using confusion matrices. Accuracy is the ratio of number of correct predictions for each class to the total number of predictions of all classes, and is calculated as:

$$Accuracy = \frac{1}{N}\sum_{c=0}^{N-1}\frac{(T_p^c + T_N^c)}{(T_p^c + T_N^c + F_p^c + F_N^c)} \times 100\% \quad (15)$$

In equation (15), c represents a particular class index from N number of classes, T_p denotes the true positives, T_N denotes true negatives, F_p denotes false positives and F_N denotes false negatives. Precision is the ratio of total number of correct predictions of positive instances to the total number of predictions with positive instances. It is calculated as:

$$Precision = \frac{1}{N}\sum_{c=0}^{N-1}\frac{T_p^c}{(T_p^c + F_p^c)} \times 100\% \quad (16)$$

The recall (also known as sensitivity) is the ratio of total number of correct predictions of positive instances to the total number of instances in an actual class. The recall and f1 score are calculated by using equations (17) and (18):

$$Recall = \frac{1}{N}\sum_{c=0}^{N-1}\frac{T_p^c}{(T_p^c + F_N^c)} \times 100\% \quad (17)$$

$$F1Score = 2 * (\frac{Precision * Recall}{Precision + Recall}) \quad (18)$$

V. RESULTS AND DISCUSSION

In this section, the results obtained through experimental evaluations of different machine learning and deep learning approaches for video classification are presented and discussed. Afterwards, the best-proposed approach is compared with existing state-of-the-art methods from literature.

A. ANALYSIS OF PRE-TRAINED CNN MODEL VARIANTS

At first, three pre-trained convolutional neural network models including Inception-V3, VGG-19 and EfficientNet-B7 are employed as video classifiers to determine the performances of these ImageNet pre-trained CNN models in our multiclass video classification problem. For each model, the last three layers of the pipeline are discarded and added with a fully connected layer of softmax activation function using three output nodes.

The transfer learning approach is implemented in a manner where weights of all layers in the model are fixed except the last fully connected layer. After training each pre-trained convolutional neural network model using the transfer learning approach, as shown in Table 3, it is analyzed that the EfficientNet-B7 model performs comparatively better than VGG-19 and Inception-V3 on the YouTube cartoon video dataset. It has achieved the highest recall score which means that the EfficientNet-B7 model retrieves more relevant instances than the remaining two pre-trained CNN models. Hence, further experiments are carried out with the EfficientNet-B7 as a base classifier.

B. ANALYSIS OF EFFICIENT-NET FEATURES WITH DIFFERENT CLASSIFIER VARIANTS

In this section, the performances of different classifiers trained on EfficientNet visual features are evaluated. For this purpose, some machine learning algorithms are considered for the video classification task. The experimental evaluation of Xu *et al.* [76] also presented that even a simple machine learning algorithm can play an effective role in video classification considering the features are distinctive enough.

This study applied three machine learning algorithms namely SVM, KNN and random forest as video classifiers

by training on EfficientNet features. The evaluation results of Table 4 shows that among three machine learning classifiers, SVM with RBF kernel achieved the highest accuracy of 72.48% on EfficientNet visual features. It is followed by the random forest and KNN (neighbors = 3) classifiers with accuracy values of 68.69% and 60.12%, respectively. The other evaluation metrics i.e., precision, recall and f1 score of SVM with RBF kernel outperformed the random forest and KNN classifiers. Apart from machine learning classifiers, an experiment is performed where EfficientNet-B7 itself is treated as a sole classifier by replacing the last three layers of architecture with a fully connected layer (units = 512) followed by an output layer (activation = softmax) of three units. Two main methods such as transfer learning and fine-tuning of EfficientNet-B7 are implemented for the video classification task. In transfer learning, the weights of all layers of EfficientNet are fixed except the last fully connected layer and fine-tuning updates the weights of an entire model.

From Table 4, it can be observed that the performances of all machine learning algorithms are relatively poor than transfer learning or fine-tuning of the EfficientNet-B7 model. In comparison, the EfficientNet model using transfer learning approach performed slightly better (accuracy = 89.07%) than fine-tuned model (accuracy = 87.89%). The main reason for such model behavior is because the ImageNet video classification dataset is much larger in scope (14 million images) than our self-curated cartoon video dataset (2.5 million video frames) used for fine-tuning of the EfficientNet model. By further examining the evaluation results of other classifier variants, it is noticed that although EfficientNet with transfer learning method yields best results than other classifiers, it still has a high ratio of false negatives (recall = 79.54%). Hence, model training by using a single fully connected layer with pre-trained CNN architecture is not sufficient. It requires some deep neural network classifier to effectively understand the hidden sequences of video representations by returning high precision-recall values for video classification. Thus, further experiments are conducted using EfficientNet-B7 as a feature extractor with other neural network models to not let any child inappropriate content go undetected in video classification.

C. ANALYSIS OF EFFICIENT-NET WITH BiLSTM AND ATTENTION-BASED BiLSTM CLASSIFIER VARIANTS

The experiments of previous sections revealed that ImageNet pre-trained EfficientNet-B7 works better as a feature extractor and this architecture in conjunction with any deep learning algorithm can successfully detect and classify unsafe video content. The bidirectional LSTM, a supervised deep learning algorithm, is opted for developing a deep learning-based framework because it preserves the contextual information in both directions of time-series data, which appears to be a suitable choice in our video classification problem. The experiments are conducted using the two-layer stack of BiLSTMs followed by fully connected (units = 4096, activation = ReLU), drop out (value = 0.3) and softmax (output units = 3) layers. Details of the complete architecture are mentioned in section III. This study implemented and evaluated different hidden units (i.e., 64, 128, 256, and 512) in each BiLSTM layer. For simplicity and consistency, the same number of hidden units are used in both layers of BiLSTM networks. You and Korhonen [77] reported that adding an attention

TABLE 3. Transfer learning using ImageNet pre-trained CNN models.

Model	Classes	Precision (%)	Recall (%)	F1 Score	Average Precision (%)	Average Recall (%)	Average F1 Score
Inception-V3 (Transfer learning)	S (0)	77.83	91.82	0.8425	80.04	77.36	0.7823
	FV (1)	79.36	71.18	0.7505			
	SN (2)	82.94	69.09	0.7539			
VGG-19 (Transfer learning)	S (0)	82.91	92.94	0.8764	80.30	78.71	0.7929
	FV (1)	80.39	72.55	0.7627			
	SN (2)	77.61	70.65	0.7397			
EfficientNet-B7 (Transfer learning)	S (0)	82.01	94.23	0.8769	**83.99**	**79.53**	**0.8154**
	FV (1)	82.86	73.98	0.7905			
	SN (2)	87.11	70.40	0.7787			

TABLE 4. Evaluation results in terms of accuracy, precision, recall and f1 score of using EfficientNet features with different classifier variants.

Method	Accuracy (%)	Precision (%)	Recall (%)	F1 Score
EfficientNet-B7 + SVM (kernel = RBF)	72.48	53.18	72.23	0.6267
EfficientNet-B7 + KNN (Neighbors =3)	60.12	44.23	39.52	0.3101
EfficientNet-B7 + Random Forest	68.69	53.76	44.19	0.4099
EfficientNet-B7 + Fully connected (Transfer Learning)	**89.07**	**86.47**	**79.54**	**0.8286**
Fine-tuned EfficientNet-B7 + Fully connected	87.89	86.31	78.11	0.8201

TABLE 5. Evaluation results in terms of accuracy, precision, recall and f1 score of using EfficientNet-B7 with BiLSTM and attention-based BiLSTM classifier variants.

Method	Training	Validation				Testing			
	Accuracy (%)	Accuracy (%)	Precision (%)	Recall (%)	F1 Score	Accuracy (%)	Precision (%)	Recall (%)	F1 Score
EfficientNet-B7 + BiLSTM with attention (64)	99.48	95.10	91.9	91.6	0.918	94.95	91.48	91.42	0.9143
EfficientNet-B7 + BiLSTM with attention (128)	99.52	95.19	92.34	91.39	0.9184	95.16	92.23	91.30	0.9175
EfficientNet-B7 + BiLSTM with attention (256)	99.56	**95.33**	**92.79**	**91.48**	**0.921**	**95.30**	**92.86**	**91.16**	**0.9195**
EfficientNet-B7 + BiLSTM with attention (512)	99.49	95.11	91.94	91.43	0.9168	94.97	91.78	91.05	0.9142
EfficientNet-B7 + BiLSTM (64)	99.56	95.55	93.15	91.90	0.9249	95.52	93.01	91.79	0.9236
EfficientNet-B7 + BiLSTM (128)	99.50	**95.66**	**93.22**	**92.30**	**0.9274**	**95.66**	**93.17**	**92.22**	**0.9267**
EfficientNet-B7 + BiLSTM (256)	99.45	95.68	93.46	91.96	0.9265	95.51	92.63	92.10	0.9236
EfficientNet-B7 + BiLSTM (512)	99.31	95.36	92.06	92.29	0.9126	95.52	92.62	92.23	0.9242

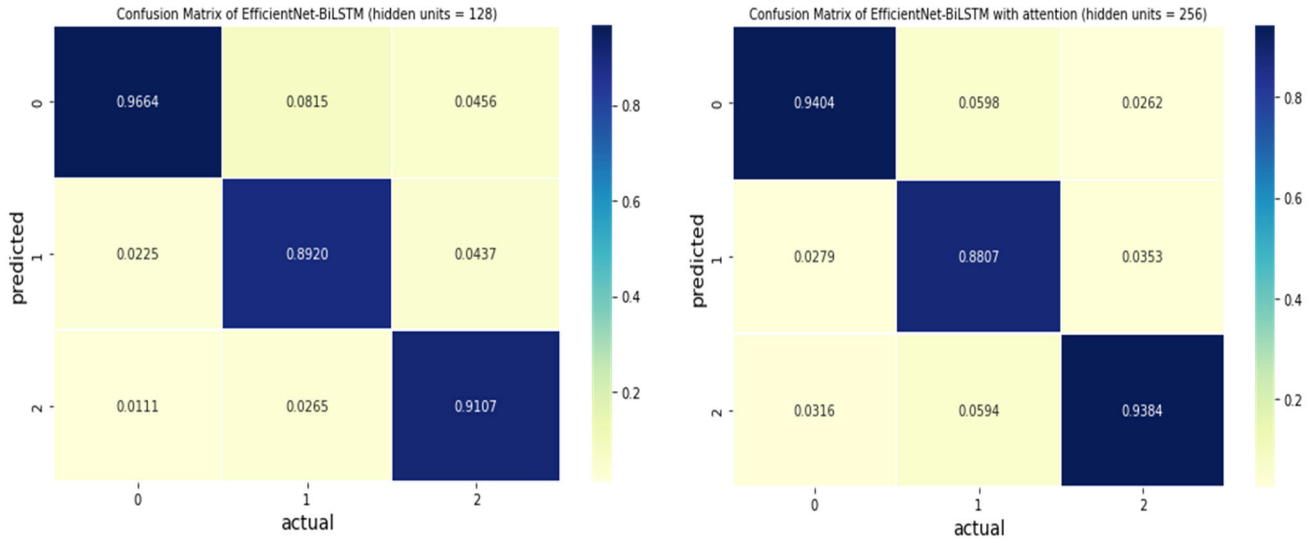


FIGURE 5. Confusion matrix results of EfficientNet-BiLSTM and attention-based EfficientNet-BiLSTM networks.

mechanism after the BiLSTM layer boosts the performance of deep neural networks for video classification. Hence, an attention mechanism-based BiLSTM model is also examined by integrating an attention block after each bidirectional layer followed by fully connected (units = 4096, activation = ReLU), drop out (value = 0.3) and softmax (output units = 3) layers. In all experiments, models are trained and evaluated for 20 epochs with an 80:20 split in which 80% of the YouTube cartoon dataset is used for training purposes and 20% for the evaluation and testing of the model. The trained model from the last epoch (epoch = 20) is tested for obtaining the final video classification scores. Table 5 demonstrates the experimental results of attention and without attention mechanism-based EfficientNet-BiLSTM models by working with the different number of hidden units (i.e., 64, 128, 256, and 512) in each bidirectional LSTM layer of the proposed framework.

The first observation from all evaluation results, as mentioned in Table 5, is that all EfficientNet-BiLSTM networks perform comparatively better than the attention-based EfficientNet-BiLSTM networks. For attention-based models, the f1 scores are improved by updating the hidden units from 64 to 128 and 256 in each BiLSTM network as it affects the network trainable parameters during backpropagation. However, it is also found that adding an excessive number of hidden units (i.e., units = 512) gradually decreases the overall network performance. Secondly, the overall behavior of attention mechanism-based neural network models is different from models with no attention blocks. In the EfficientNet-BiLSTM network, upgrading the hidden units in BiLSTMs from 64 to 128 immediately resulted in the best performing model of all experiments by showing the highest f1 score (0.9267). However, it drastically decreases the performance by adding more hidden units in BiLSTMs

TABLE 6. Performance comparison between the proposed EfficientNet-BiLSTM model with existing state-of-the-art video classification techniques.

Approach	Pre-trained Model	Modality	Epochs	Classes	Accuracy (%)	Precision (%)	Recall (%)	F1 Score
[61]	NASNet	Video	-	2	92.6	-	-	0.887
[50]	EfficientNet-B7	Video	-	2	87.87	86.65	89.12	0.878
[60]	GoogLeNet	Video	200	2	85.1	-	-	0.882
[62]	VGG-16	Video	-	4	-	81.0	80	-
[48]	ResNet-101	Video	-	2	95.5 ± 1	-	-	
[20]	Inception-V3	Multimodal	-	2	84.3	82.1	89.0	0.829
[22]	VGG-19	Multimodal	25	4	92.83	92.72	91.71	-
[10]	ResNet-50	Multimodal	10	2	95%	-	-	-
[11]	-	Multimodal	-	2	51%	-	-	-
[17]	-	Text	-	6	-	72.2	78.5	0.745
[21]	-	Video	-	2	74%	-	-	-
Proposed (BiLSTM with attention = 256)	EfficientNet-B7	Video	20	3	95.30	92.86	91.16	0.9195
Proposed (BiLSTM = 128)	EfficientNet-B7	Video	20	3	**95.66**	**93.17**	**92.22**	**0.9267**
Proposed (BiLSTM =128)	EfficientNet-B7	Video	20	2 (binary)	Safe= 95.08 Unsafe = 95.95	Safe = 94.177 Unsafe = 92.67	Safe = 96.61 Unsafe = 90.03	Safe = 0.9538 Unsafe = 0.9132

(i.e., 256 and 512). A detailed performance comparison between the EfficientNet with attention and without attention mechanism-based BiLSTM models are presented through confusion matrices for all three classes. The diagonal values represent the correctly classified number of instances in each class, but anything off the diagonal indicates incorrect classification instances. The evaluation results of the EfficientNet-BiLSTM model with and without attention blocks for the YouTube cartoon video dataset are illustrated in Fig. 5. Overall, the EfficientNet and BiLSTM network with 128 hidden units in each bidirectional layer achieved the highest validation (f1 score = 0.9274) and testing scores (f1 score = 0.9267).

D. PERFORMANCE COMPARISON WITH EXISTING STATE-OF-THE-ART CLASSIFICATION METHODS

We compare the performance of the proposed EfficientNet-BiLSTM model with existing state-of-the-art models and methods employed for inappropriate content classification using different YouTube data modalities.

Table 6 summarizes the results and quality scores of existing and proposed classification methods. It is worth noting that existing studies explored different YouTube modalities (i.e., text, audio, video, and metadata) for different classifications. The most common strategy in existing studies, for unsafe content classification, is using pre-trained CNN models with either LSTM-based classifiers [10], [20], [22], [48], [62] or machine learning-based classifiers [50], [60], [61]. Compared with the approaches that use pre-trained CNN features with machine learning classifiers, our EfficientNet-BiLSTM classifier method yielded higher accuracy than GoogLeNet-SVM [60], fine-tuned NASNet-SVM [61], and EfficientNet-SVM [50] approaches by significant margins of 3.06%, 7.79%, and 10.1%, respectively. In comparison with base models using pre-trained CNNs and LSTMs, the Inception-V3 with LSTM approach [20] reported f1 score of 0.828 which is much lower than our with attention (f1 score = 0.9195) and without attention-based (f1 score = 0.9267) BiLSTM classifier variants. It is also worth mentioning that the ResNet-LSTM model in existing studies [10], [48] attained comparable accuracy results to our proposed technique. It can be explained by the fact that the studies reporting these approaches performed binary video classification which is much simpler than multi-class video classification. Note that the proposed model still outperformed some existing approaches of multiclass video classification using VGG-LSTM-based models [22], [62], which shows that BiLSTM has high robustness on time-series data modeling. In addition, some studies [11], [17], [21] used simple convolutional neural networks and reported the lowest classification accuracy and f1 scores. Hence, it is deduced that simple CNNs are not sufficient to understand the complexities of YouTube data modalities. Overall, the performance comparison showed that the proposed EfficientNet using BiLSTM (hidden units = 128) surpassed the existing studies in inappropriate video content detection and classification.

VI. CONCLUSION AND FUTURE WORK

In this paper, a novel deep learning-based framework is proposed for child inappropriate video content detection and classification. Transfer learning using EfficientNet-B7 architecture is employed to extract the features of videos.

The extracted video features are processed through the BiLSTM network, where the model learns the effective video representations and performs multiclass video classification. All evaluation experiments are performed by using a manually annotated cartoon video dataset of 111,156 video clips collected from YouTube. The evaluation results indicated that proposed framework of EfficientNet-BiLSTM (with hidden units = 128) exhibits higher performance (accuracy = 95.66%) than other experimented models including EfficientNet-FC, EfficientNet-SVM, EfficientNet-KNN, EfficientNet-Random Forest, and EfficientNet-BiLSTM with attention mechanism-based models (with hidden units = 64, 128, 256, and 512). Moreover, the performance comparison with existing state-of-the-art models also demonstrated that our BiLSTM-based framework surpassed other existing models and methods by achieving the highest recall score of 92.22%. The advantages of the proposed deep learning-based children inappropriate video content detection system are as follows:

1) It works by considering the real-time conditions by processing the video with a speed of 22 fps using EfficientNet-B7 and BiLSTM-based deep learning framework, which helps in filtering the live-captured videos.
2) It can assist any video sharing platform to either remove the video containing unsafe clips or blur/hide any portion with unsettling frames.
3) It may also help in the development of parental control solutions on the Internet through plugins or browser extensions where child unsafe content can be filtered automatically.

Furthermore, our methodology to detect inappropriate children content from YouTube is independent of YouTube video metadata which can easily be altered by malicious uploaders to deceive the audiences. In the future, we intend to combine the temporal stream using optical flow frames with the spatial stream of the RGB frames to further improve the model performance by better understanding the global representations of videos. We also aim to increase the classification labels to target the different types of inappropriate children content of YouTube videos.

ACKNOWLEDGMENT

The authors are appreciative of Prof. Dr. Hafiz Adnan Habib (Head of Department of Computer Engineering, University of Engineering and Technology, Taxila) for providing valuable advice and suggestions in this study.

REFERENCES

[1] L. Ceci. *YouTube Usage Penetration in the United States 2020, by Age Group*. Accessed: Nov. 1, 2021. [Online]. Available: https://www.statista.com/statistics/296227/us-youtube-reach-age-gender/
[2] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for YouTube recommendations," in *Proc. 10th ACM Conf. Recommender Syst.*, Sep. 2016, pp. 191–198, doi: 10.1145/2959100.2959190.
[3] M. M. Neumann and C. Herodotou, "Evaluating YouTube videos for young children," *Educ. Inf. Technol.*, vol. 25, no. 5, pp. 4459–4475, Sep. 2020, doi: 10.1007/s10639-020-10183-7.
[4] J. Marsh, L. Law, J. Lahmar, D. Yamada-Rice, B. Parry, and F. Scott, *Social Media, Television and Children*. Sheffield, U.K.: Univ. Sheffield, 2019. [Online]. Available: https://www.stac-study.org/downloads/STAC_Full_Report.pdf
[5] L. Ceci. *YouTube—Statistics & Facts*. Accessed: Sep. 01, 2021. [Online]. Available: https://www.statista.com/topics/2019/youtube/
[6] M. M. Neumann and C. Herodotou, "Young children and YouTube: A global phenomenon," *Childhood Educ.*, vol. 96, no. 4, pp. 72–77, Jul. 2020, doi: 10.1080/00094056.2020.1796459.
[7] S. Livingstone, L. Haddon, A. Görzig, and K. Ólafsson, *Risks and Safety on the Internet: The Perspective of European Children: Full Findings and Policy Implications From the EU Kids Online Survey of 9-16 Year Olds and Their Parents in 25 Countries*. London, U.K.: EU Kids Online, 2011. [Online]. Available: http://eprints.lse.ac.U.K./id/eprint/33731
[8] B. J. Bushman and L. R. Huesmann, "Short-term and long-term effects of violent media on aggression in children and adults," *Arch. Pediatrics Adolescent Med.*, vol. 160, no. 4, pp. 348–352, 2006, doi: 10.1001/archpedi.160.4.348.
[9] S. Maheshwari. (2017). *On YouTube Kids, Startling Videos Slip Past Filters*. The New York Times. [Online]. Available: https://www.nytimes.com/2017/11/04/business/media/youtube-kids-paw-patrol.html
[10] C. Hou, X. Wu, and G. Wang, "End-to-end bloody video recognition by audio-visual feature fusion," in *Proc. Chin. Conf. Pattern Recognit. Comput. Vis. (PRCV)*, 2018, pp. 501–510, doi: 10.1007/978-3-030-03398-9_43.
[11] A. Ali and N. Senan, "Violence video classification performance using deep neural networks," in *Proc. Int. Conf. Soft Comput. Data Mining*, 2018, pp. 225–233, doi: 10.1007/978-3-319-72550-5_22.
[12] H.-E. Lee, T. Ermakova, V. Ververis, and B. Fabian, "Detecting child sexual abuse material: A comprehensive survey," *Forensic Sci. Int., Digit. Invest.*, vol. 34, Sep. 2020, Art. no. 301022, doi: 10.1016/j.fsidi.2020.301022.
[13] R. Brandom. (2017). *Inside Elsagate, The Conspiracy Fueled War on Creepy YouTube Kids Videos*. [Online]. Available: https://www.theverge.com/2017/12/8/16751206/elsagate-youtube-kids-creepy-conspiracy-theory
[14] Reddit. *What is ElsaGate?* Accessed: Dec. 14, 2020. [Online]. Available: https://www.reddit.com/r/ElsaGate/comments/6o6baf/
[15] B. Burroughs, "YouTube kids: The app economy and mobile parenting," *Soc. media+ Soc.*, vol. 3, May 2017, Art. no. 2056305117707189, doi: 10.1177/2056305117707189.
[16] H. Wilson, "YouTube is unsafe for children: YouTube's safeguards and the current legal framework are inadequate to protect children from disturbing content," *Seattle J. Technol., Environ. Innov. Law*, vol. 10, no. 1, p. 8, 2020. [Online]. Available: https://digitalcommons.law.seattleu.edu/sjteil/vol10/iss1/8
[17] S. Alshamrani, A. Abusnaina, M. Abuhamad, D. Nyang, and D. Mohaisen, "Hate, obscenity, and insults: Measuring the exposure of children to inappropriate comments in YouTube," in *Proc. Companion Proc. Web Conf.*, Apr. 2021, pp. 508–515, doi: 10.1145/3442442.3452314.
[18] N. Elias and I. Sulkin, "YouTube viewers in diapers: An exploration of factors associated with amount of toddlers' online viewing," *Cyberpsychol., J. Psychosoc. Res. Cyberspace*, vol. 11, no. 3, p. 2, Nov. 2017, doi: 10.5817/cp2017-3-2.
[19] D. Craig and S. Cunningham, "Toy unboxing: Living in a (n unregulated) material world," *Media Int. Aust.*, vol. 163, no. 1, pp. 77–86, May 2017, doi: 10.1177/1329878X17693700.
[20] K. Papadamou, A. Papasavva, S. Zannettou, J. Blackburn, N. Kourtellis, I. Leontiadis, G. Stringhini, and M. Sirivianos, "Disturbed YouTube for kids: Characterizing and detecting inappropriate videos targeting young children," in *Proc. Int. AAAI Conf. Web Soc. Media*, 2020, pp. 522–533. [Online]. Available: https://ojs.aaai.org/index.php/ICWSM/article/view/7320/7174
[21] R. Kaushal, S. Saha, P. Bajaj, and P. Kumaraguru, "KidsTube: Detection, characterization and analysis of child unsafe content & promoters on YouTube," in *Proc. 14th Annu. Conf. Privacy, Secur. Trust (PST)*, Dec. 2016, pp. 157–164, doi: 10.1109/pst.2016.7906950.
[22] R. Tahir, F. Ahmed, H. Saeed, S. Ali, F. Zaffar, and C. Wilson, "Bringing the kid back into YouTube kids: Detecting inappropriate content on video streaming platforms," in *Proc. IEEE/ACM Int. Conf. Adv. Soc. Netw. Anal. Mining*, Aug. 2019, pp. 464–469, doi: 10.1145/3341161.3342913.

[23] A. Ulges, C. Schulze, D. Borth, and A. Stahl, "Pornography detection in video benefits (a lot) from a multi-modal approach," in *Proc. ACM Int. Workshop Audio Multimedia Methods Large-Scale Video Anal.*, 2012, pp. 21–26, doi: 10.1145/2390214.2390222.

[24] C. Caetano, S. Avila, S. Guimaraes, and A. D. A. Araújo, "Pornography detection using BossaNova video descriptor," in *Proc. 22nd Eur. Signal Process. Conf.*, 2014, pp. 1681–1685. [Online]. Available: https://ieeexplore.ieee.org/document/6952616

[25] L. Duan, G. Cui, W. Gao, and H. Zhang, "Adult image detection method base-on skin color model and support vector machine," in *Proc. Asian Conf. Comput. Vis.*, 2002, pp. 797–800. [Online]. Available: http://aprs.dictaconference.org/accv2002/accv2002_proceedings/Duan797.pdf

[26] C. Jansohn, A. Ulges, and T. M. Breuel, "Detecting pornographic video content by combining image features with motion information," in *Proc. 17th ACM Int. Conf. Multimedia*, 2009, pp. 601–604, doi: 10.1145/1631272.1631366.

[27] P. Zhou, Q. Ding, H. Luo, and X. Hou, "Violence detection in surveillance video using low-level features," *PLoS ONE*, vol. 13, no. 10, Oct. 2018, Art. no. e0203668, doi: 10.1371/journal.pone.0203668.

[28] M. B. Garcia, T. F. Revano, B. G. M. Habal, J. O. Contreras, and J. B. R. Enriquez, "A pornographic image and video filtering application using optimized nudity recognition and detection algorithm," in *Proc. IEEE 10th Int. Conf. Humanoid, Nanotechnol., Inf. Technol., Commun. Control, Environ. Manage. (HNICEM)*, Nov. 2018, pp. 1–5, doi: 10.1109/HNICEM.2018.8666227.

[29] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1725–1732, doi: 10.1109/cvpr.2014.223.

[30] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst. (NIPS)*, 2014, pp. 568–576. [Online]. Available: https://dl.acm.org/doi/10.5555/2968826.2968890

[31] Z. Wu, X. Wang, Y.-G. Jiang, H. Ye, and X. Xue, "Modeling spatial-temporal clues in a hybrid deep learning framework for video classification," in *Proc. 23rd ACM Int. Conf. Multimedia*, Oct. 2015, pp. 461–470, doi: 10.1145/2733373.2806222.

[32] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4694–4702, doi: 10.1109/CVPR.2015.7299101.

[33] J. P. Verma and S. Agrawal, "Big data analytics: Challenges and applications for text, audio, video, and social media data," *Int. J. Soft Comput., Artif. Intell. Appl.*, vol. 5, no. 1, pp. 41–51, Feb. 2016, doi: 10.5121/ijscai.2016.5105.

[34] Y.-G. Jiang, Z. Wu, J. Wang, X. Xue, and S.-F. Chang, "Exploiting feature and class relationships in video categorization with regularized deep neural networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 2, pp. 352–364, Feb. 2017, doi: 10.1109/TPAMI.2017.2670560.

[35] M. J. Jones and J. M. Rehg, "Statistical color models with application to skin detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 1999, pp. 274–280, doi: 10.1109/CVPR.1999.786951.

[36] T. Endeshaw, J. Garcia, and A. Jakobsson, "Classification of indecent videos by low complexity repetitive motion detection," in *Proc. 37th IEEE Appl. Imag. Pattern Recognit. Workshop*, Oct. 2008, pp. 1–7, doi: 10.1109/AIPR.2008.4906438.

[37] N. Rea, G. Lacey, R. Dahyot, and C. Lambe, "Multimodal periodicity analysis for illicit content detection in videos," in *Proc. 3rd Eur. Conf. Vis. Media Prod.*, 2006, pp. 106–114, doi: 10.1049/cp:20061978.

[38] Y. Liu, X. Wang, Y. Zhang, and S. Tang, "Fusing audio-words with visual features for pornographic video detection," in *Proc. IEEE 10th Int. Conf. Trust, Secur. Privacy Comput. Commun.*, Nov. 2011, pp. 1488–1493, doi: 10.1109/TRUSTCOM.2011.205.

[39] Y. Liu, Y. Yang, H. Xie, and S. Tang, "Fusing audio vocabulary with visual features for pornographic video detection," *Future Gener. Comput. Syst.*, vol. 31, pp. 69–76, Feb. 2014, doi: 10.1016/j.future.2012.08.012.

[40] V. M. T. Ochoa, S. Y. Yayilgan, and F. A. Cheikh, "Adult video content detection using machine learning techniques," in *Proc. 8th Int. Conf. Signal Image Technol. Internet Based Syst.*, Nov. 2012, pp. 967–974, doi: 10.1109/sitis.2012.143.

[41] S. Jung, J. Youn, and S. Sull, "A real-time system for detecting indecent videos based on spatiotemporal patterns," *IEEE Trans. Consum. Electron.*, vol. 60, no. 4, pp. 696–701, Nov. 2014, doi: 10.1109/TCE.2014.7027345.

[42] S. Tang, T.-S. Chua, J. Li, Y. Zhang, C. Xie, M. Li, Y. Liu, X. Hua, Y.-T. Zheng, and J. Tang, "Pornprobe: An LDA-SVM based pornography detection system," in *Proc. 17th ACM Int. Conf. Multimedia*, 2009, pp. 1003–1004, doi: 10.1145/1631272.1631490.

[43] S. Lee, W. Shim, and S. Kim, "Hierarchical system for objectionable video detection," *IEEE Trans. Consum. Electron.*, vol. 55, no. 2, pp. 677–684, May 2009, doi: 10.1109/TCE.2009.5174439.

[44] A. P. B. Lopes, S. E. F. D. Avila, A. N. A. Peixoto, R. S. Oliveira, M. D. M. Coelho, and A. D. A. Araújo, "Nude detection in video using bag-of-visual-features," in *Proc. XXII Brazilian Symp. Comput. Graph. Image Process.*, Oct. 2009, pp. 224–231, doi: 10.1109/sibgrapi.2009.32.

[45] S. Reddy, N. Srikanth, and G. Sharvani, "Development of kid-friendly YouTube access model using deep learning," in *Data Science and Security*. Singapore: Springer, 2021, pp. 243–250, doi: 10.1007/978-981-15-5309-7_26.

[46] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," in *Proc. 28th Int. Conf. Int. Conf. Mach. Learn. (ICML)*, 2011, pp. 689–696. [Online]. Available: https://dl.acm.org/doi/10.5555/3104482.3104569

[47] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 3104–3112. [Online]. Available: https://dl.acm.org/doi/10.5555/2969033.2969173

[48] J. Wehrmann, G. S. Simões, R. C. Barros, and V. F. Cavalcante, "Adult content detection in videos with convolutional and recurrent neural networks," *Neurocomputing*, vol. 272, pp. 432–438, Jan. 2018, doi: 10.1016/j.neucom.2017.07.012.

[49] M. Perez, S. Avila, D. Moreira, D. Moraes, V. Testoni, E. Valle, S. Goldenstein, and A. Rocha, "Video pornography detection through deep learning techniques and motion information," *Neurocomputing*, vol. 230, pp. 279–293, Mar. 2017, doi: 10.1016/j.neucom.2016.12.017.

[50] N. Aldahoul, H. A. Karim, M. H. L. Abdullah, and A. S. Ba Wazir, "An evaluation of traditional and CNN-based feature descriptors for cartoon pornography detection," *IEEE Access*, vol. 9, pp. 39910–39925, 2021, doi: 10.1109/ACCESS.2021.3064392.

[51] H. Yenala, A. Jhanwar, M. K. Chinnakotla, and J. Goyal, "Deep learning for detecting inappropriate content in text," *Int. J. Data Sci. Analytics*, vol. 6, no. 4, pp. 273–286, Dec. 2018, doi: 10.1007/s41060-017-0088-4.

[52] R. E. Trana, C. E. Gomez, and R. F. Adler, "Fighting cyberbullying: An analysis of algorithms used to detect harassing text found on YouTube," in *Proc. Int. Conf. Appl. Hum. Factors Ergonom.*, 2020, pp. 9–15, doi: 10.1007/978-3-030-51328-3_2.

[53] M. Dadvar and K. Eckert, "Cyberbullying detection in social networks using deep learning based models," in *Proc. Int. Conf. Big Data Analytics Knowl. Discovery*, 2020, pp. 245–255, doi: 10.1201/9781003134527-11.

[54] H. Mohaouchane, A. Mourhir, and N. S. Nikolov, "Detecting offensive language on Arabic social media using deep learning," in *Proc. 6th Int. Conf. Soc. Netw. Anal., Manage. Secur. (SNAMS)*, Oct. 2019, pp. 466–471, doi: 10.1109/snams.2019.8931839.

[55] S. Alshamrani, "Detecting and measuring the exposure of children and adolescents to inappropriate comments in YouTube," in *Proc. 29th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2020, pp. 3213–3216, doi: 10.1145/3340531.3418511.

[56] S. Alshamrani, M. Abuhamad, A. Abusnaina, and D. A. Mohaisen, "Investigating online toxicity in users interactions with the mainstream media channels on YouTube," in *Proc. CIKM Workshops*, 2020, pp. 1–6. [Online]. Available: http://ceur-ws.org/Vol-2699/paper39.pdf

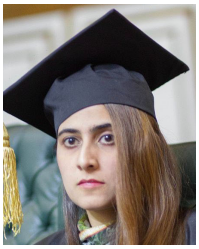
[57] E. Mariconti, G. Suarez-Tangil, J. Blackburn, E. De Cristofaro, N. Kourtellis, and I. Leontiadis, "'You know what to do' proactive detection YouTube videos targeted by coordinated hate attacks," in *Proc. ACM Hum.-Comput. Interact.*, vol. 3, pp. 1–21, Nov. 2019, doi: 10.1145/3359309.

[58] M. Gao, J. Jiang, L. Ma, S. Zhou, G. Zou, J. Pan, and Z. Liu, "Violent crowd behavior detection using deep learning and compressive sensing," in *Proc. Chin. Control Decis. Conf. (CCDC)*, Jun. 2019, pp. 613–625, doi: 10.1109/ccdc.2019.8832598.

[59] S. Alghowinem, "A safer YouTube kids: An extra layer of content filtering using automated multimodal analysis," in *Proc. SAI Intell. Syst. Conf.*, 2018, pp. 294–308, doi: 10.1007/978-3-030-01054-6_21.

[60] P. Vitorino, S. Avila, M. Perez, and A. Rocha, "Leveraging deep neural networks to fight child pornography in the age of social media," *J. Vis. Commun. Image Represent.*, vol. 50, pp. 303–313, Jan. 2018, doi: 10.1016/j.jvcir.2017.12.005.

[61] A. Ishikawa, E. Bollis, and S. Avila, "Combating the elsagate phenomenon: Deep learning architectures for disturbing cartoons," in *Proc. 7th Int. Workshop Biometrics Forensics (IWBF)*, May 2019, pp. 1–6, doi: 10.1109/iwbf.2019.8739202.
[62] S. Singh, R. Kaushal, A. B. Buduru, and P. Kumaraguru, "KidsGUARD: Fine grained approach for child unsafe video representation and detection," in *Proc. 34th ACM/SIGAPP Symp. Appl. Comput.*, Apr. 2019, pp. 2104–2111, doi: 10.1145/3297280.3297487.
[63] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255, doi: 10.1109/CVPR.2009.5206848.
[64] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," 2019, *arXiv:1905.11946*.
[65] S. Abu-El-Haija, N. Kothari, J. Lee, P. Natsev, G. Toderici, B. Varadarajan, and S. Vijayanarasimhan, "YouTube-8M: A large-scale video classification benchmark," 2016, *arXiv:1609.08675*.
[66] K. Soomro, A. Roshan Zamir, and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," 2012, *arXiv:1212.0402*.
[67] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: A large video database for human motion recognition," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2556–2563, doi: 10.1109/iccv.2011.6126543.
[68] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman, "The kinetics human action video dataset," 2017, *arXiv:1705.06950*.
[69] L. Wolf, T. Hassner, and I. Maoz, "Face recognition in unconstrained videos with matched background similarity," in *Proc. CVPR*, Jun. 2011, pp. 529–534, doi: 10.1109/cvpr.2011.5995566.
[70] M. Kim, S. Kumar, V. Pavlovic, and H. Rowley, "Face tracking and recognition with visual constraints in real-world videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8, doi: 10.1109/cvpr.2008.4587572.
[71] A. Bermingham, M. Conway, L. McInerney, N. O'Hare, and A. F. Smeaton, "Combining social network analysis and sentiment analysis to explore the potential for online radicalisation," in *Proc. Int. Conf. Adv. Soc. Netw. Anal. Mining*, Jul. 2009, pp. 231–236, doi: 10.1109/asonam.2009.31.
[72] S. Guadarrama, N. Krishnamoorthy, G. Malkarnenkar, S. Venugopalan, R. Mooney, T. Darrell, and K. Saenko, "YouTube2Text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2712–2719, doi: 10.1109/iccv.2013.337.
[73] J. Xu, T. Mei, T. Yao, and Y. Rui, "MSR-VTT: A large video description dataset for bridging video and language," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5288–5296, doi: 10.1109/cvpr.2016.571.
[74] N. Ketkar, "Introduction to keras," in *Deep Learning With Python*. Berkeley, CA, USA: Springer, 2017, pp. 97–111, doi: 10.1007/978-1-4842-2766-4_7.
[75] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, and J. Dean, "TensorFlow: A system for large-scale machine learning," in *Proc. 12th USENIX Symp. Oper. Syst. Des. Implement. (OSDI)*, 2016, pp. 265–283. [Online]. Available: https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf
[76] Z. Xu, Y. Yang, and A. G. Hauptmann, "A discriminative CNN video representation for event detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1798–1807, doi: 10.1109/cvpr.2015.7298789.
[77] J. You and J. Korhonen, "Attention boosted deep networks for video classification," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2020, pp. 1761–1765, doi: 10.1109/ICIP40778.2020.9190996.



KANWAL YOUSAF received the B.Sc. (Hons.) and M.Sc. degrees in Software Engineering from the University of Engineering and Technology (UET), Taxila, in 2010 and 2013, respectively, where she is currently pursuing the Ph.D. degree. She is also working as a Lecturer at UET, Taxila. Her research interests include deep learning, artificial neural networks and machine learning.

TABASSAM NAWAZ received the Ph.D. degree from the University of Engineering and Technology (UET), Taxila. He is currently serving as a Professor and the Head of the Software Engineering Department, UET, Taxila. His research interests include advanced databases, and object-oriented design and analysis.