A Deep Learning Approach to Detect Pornography Videos in Educational Repositories

Pedro V. A. de Freitas[1], **Antonio J. G. Busson**[1], **Álan L. V. Guedes**[1], **Sérgio Colcher**[1]

{pedropva,busson,alan}@telemidia.puc-rio.br, colcher@inf.puc-rio.br

[1]TeleMídia/Pontíficia Universidade Católica do Rio de Janeiro (PUC-Rio)

Abstract. *A large number of videos are uploaded on educational platforms every minute. Those platforms are responsible for any sensitive media uploaded by their users. An automated detection system to identify pornographic content could assist human workers by pre-selecting suspicious videos. In this paper, we propose a multimodal approach to adult content detection. We use two Deep Convolutional Neural Networks to extract high-level features from both image and audio sources of a video. Then, we concatenate those features and evaluate the performance of classifiers on a set of mixed educational and pornographic videos. We achieve an F1-score of 95.67% on the educational and adult videos set and an F1-score of 94% on our test subset for the pornographic class.*

1. Introduction

More than 500 hours of video were uploaded to YouTube every minute during 2019.[1] This huge amount of data sharing pattern presents a challenge to the control of the type of content that is loaded to these video repositories. By allowing the upload of pornographic content from malicious users, content providers become exposed to legal issues. Youtube and other providers have already registered the upload of pornography content. In Brazil, the "Cicarely case" was an example that forced youtube to be blocked.[2] In our research, we are interested in helping to avoid scenarios where pornography can be uploaded to education channels, which might expose students, sometimes underage, to this kind of content.[3]. We define pornography as any media that contains explicit content, such as sexualized nudity or intercourse. On the other hand, the other type of video are the ones who do not have pornographic content in them, referred hereby as educational videos.

Controlling the type of content uploaded to video storage services requires an automatic analysis in an accurate and efficient way. Methods based on *Deep Learning* (DL) became the *state-of-the-art* in various segments related to automatic video analysis. More specifically, Convolutional Neural Networks (CNN) architectures, or ConvNets, have become the primary method used for audio-visual pattern recognition.

In this work, we created a CNN based model for video feature extraction and validate these video features experimenting with different baseline models to detect pornography. Then we evaluate the best model in a dataset created with videos sampled

[1]https://kinsta.com/blog/youtube-stats/

[2]http://g1.globo.com/Noticias/Tecnologia/0,,AA1412609-6174-363,00.html

[3]https://g1.globo.com/sp/sao-paulo/noticia/2020/06/19/professor-de-etec-na-zona-norte-de-sp-e-afastado-apos-se-masturbar-durante-aula-virt
ghtml

DOI: 10.5753/cbie.sbie.2020.1253

from the Brazilian RNP (National Research Network) repository video@RNP.[4] In our experimentation, the best model achieves a recall of 94.4% and an F1-score of 95.6% for pornography class.

Other papers ,such as [Freitas et al. 2019] and [Moreira et al. 2019], share our motivations and objectives, as described in Section 2. However, most of them do not use both audio and image for classification. Some use hand-crafted feature extraction methods, not the latest feature extraction CNN, which has been showing great potential in video recognition and classification. Our work uses two types of CNNs: one to extract image sequence features and the other to extract audio features. As we get one feature vector for each second of the video, we can approach the feature classification task as a time series classification, using a Recurrent Neural Networks (RNN) as a baseline. We also can combine those features to create a single feature vector for the entire video, which then is used as the input for other baseline classifiers. Our method uses a rather simpler approach for video classification and yet it still yields results significantly close to related works.

To present our proposal, this paper is organized as follows. Section 2 discusses the related work. Then, Section 3 explains the used model to detect pornography videos. We present our *dataset* in Section 4. Then, we present experiments and results in Section 5. Finally, Section 6 is devoted to our final remarks and future work.

2. Related Works

Sing *at. al.*[Singh et al. 2019] proposes a fine-grained approach for child unsafe video representation and detection. One of its main objectives is to optimize detection on sparsely present child unsafe content and it does so by using a VGG16 Convolutional Neural Network (CNN) to encode each frame, at 1-second granularity, in 512 real values. Then an LSTM autoencoder is trained to output the sequence backward on those encoded frames. Once the LSTM autoencoder is trained, then a fully connected layer of neurons is used to fine-tune and classify each frame. The dataset used comprises of 109,835 short-duration video-clips extracted from four animes. The results for binary classification using safe and unsafe classes were 81% recall for unsafe and 80% recall for safe class. Although having similar objectives and also using a CNN based encoding method, our approach differs by using both visual and auditive features to encode a video. Their work uses 1 frame per second granularity and ours has the same encoding rate. The main differences between both works are on the dataset: Theirs consists of small clips of only anime videos. The one we use also has other types of videos such as live-action and other animations. In the dataset we use, the length of videos range from 6 seconds to 30 minutes.

Song *at. al.* [Song and Kim 2020] proposed a multimodal stacking scheme for quick and accurate online detection of pornographic content. Their work uses both visual and auditory features as input for their detection method. They use a VGG16 model and a bi-directional Recurrent Neural Network (RNN) to extract visual features and a combination of a Mel-scaled spectrogram followed by multilayered dilated convolutions to extract audio features. Using only the visual and auditory features, a video classifier and an audio classifier are trained, respectively. By using both features together, one fusion classifier is also trained. Then, these three component classifiers are combined

[4] `https://www.video.rnp.br`

in an ensemble scheme to reduce the false-negative errors and for faster detection. The proposed detection method yields a true positive rate of 95.40% and a false negative rate of 4.60% on the pornography class, totaling a recall for pornography class of 95.40%. The dataset used was the pornography-2k dataset [Moreira et al. 2016] plus examples of videos with only pornographic or non-pornographic audio collected by the authors. This work is similar to ours because it also uses a multimodal approach to pornography detection, it uses the same dataset as us, it has the same sampling rate of a frame for each second, and uses a deep learning method for extracting high-level features, which are then classified by one or more machine learning models. In contrast, use different feature extraction methods for image and audio. Also, diverging to their ensemble approach, we use a single model to classify the extracted features from our dataset.

Moreira *at.al.* [Moreira et al. 2019] has similar detection focuses as ours: Pornography and violence. Their method uses four multimodal classifiers, two for audio and two for image, those classifiers were fed features from multiple handcrafted feature extraction methods. Our uses Deep Learning to extract high-level features and uses one classifier for both aggregation and classification of the features. They proposed a method for sensitive scene localization which uses the output of four multimodal classifiers on snippets of the video, then creates a fusion vector at each second of the video. Finally, they test different classifiers on the fusion vector for each task: detecting pornography and detecting violence. Their best result on the pornography task was 90.75% accuracy and 93.53% on F2 metric. For the violent videos, they achieved 0.502 on the MAP2014 evaluation metric. Some differences between this work and ours are mainly its objectives: To detect if and at what time the pornographic video occurs and also identify violent content. While our only objective is to detect if there is or not pornographic content in a video. Other differences stand out as the dataset and the methods used for feature extraction and classification. We use a well known pornography dataset and investigate what results a deep learning-based approach to this problem can yield.

Liu *at. al.* [Liu et al. 2020] propose a multimodal approach to pornography detection, it uses audio-frames and visual-frames to create handcrafted low-level features based on, respectively on periodic patterns and salient regions. Once those features are extracted, they use k-means clustering to create audio and visual codebooks. Then, low-level audio and visual features of test videos are converted into mid-level semantic histograms via de audio or visual codebook. Finally, the histograms are concatenated to represent the video and a periodicity based video decision algorithm is used to fuse the classification results of multi-modal codebooks and the results of an SVM trained on the concatenated mid-level semantic features train set. The true positive rate of their approach achieves 96.7% while the false positive rate is about 10%. Liu *at. al.* detects pornography, and also use handcrafted features such as Region Of Interest (ROI) extraction and skin-color segmentation. Whereas our approach uses a fully automatic feature extraction method based on CNNs and our feature fusion method consists on just a concatenation followed by a classifier.

Although most of the aforementioned works share our motivations and objectives, most of them do not use both audio and image for classification. Some use hand-crafted feature extraction methods, not the latest feature extraction CNN, which has been showing great potential in video recognition and classification.

3. Pornography Detection Model

Our CNN-based pornography detector is composed of two modules. The first one is what researchers call the *backbone*, which acts as the feature extractor from which the whole model draws its discriminating power. The second module, the *classifier*, operates over the extracted features from the backbone to aggregate and classify them. The architecture of our pornography detector is illustrated in Fig. 1.

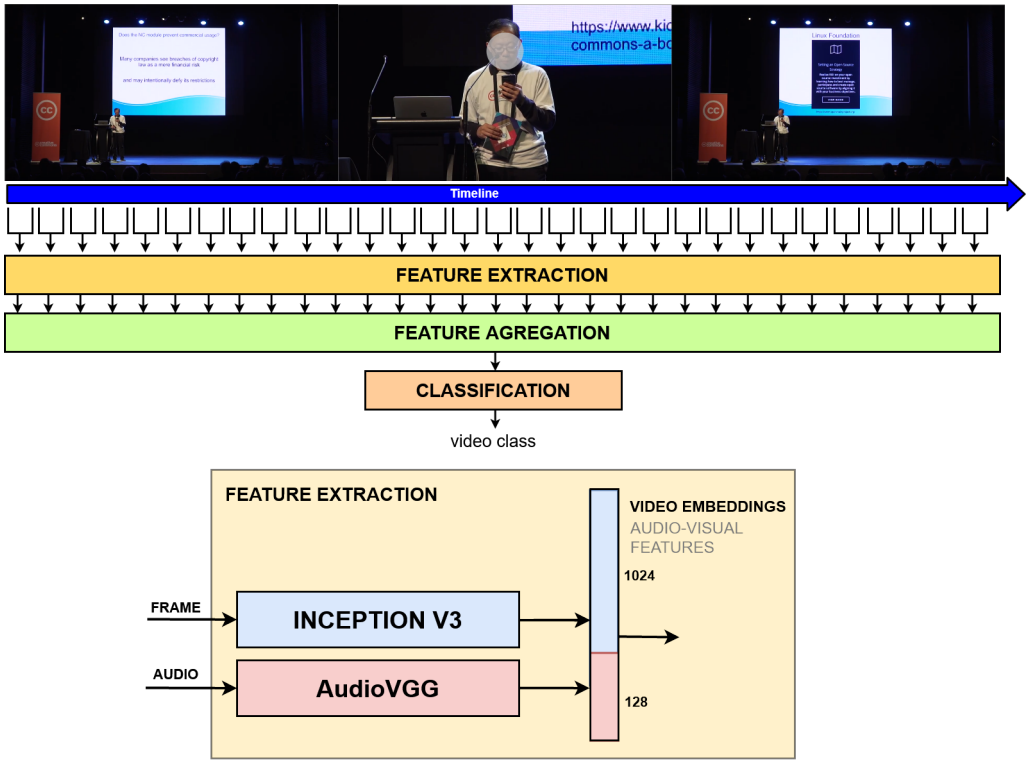


Figura 1. Bimodal-based architecture for Pornography video classification

We opted for a bi-modal approach that uses two backbones to extract the audio and image features because it showed good results for the YouTube8m dataset challenge for multi-label classification of videos [Abu-El-Haija et al. 2016, Lee et al. 2018]. Once we have extracted the features from a video, we use a shallow model to perform the video classification.

In the remainder of this section, we detail the embeddings extractor and the algorithms used for classification. First, in Section 3.1, we present the CNN used as the backbone for feature extraction. Then, in Section 3.2, we present the algorithms used for classification.

3.1. Video Embeddings Extractor

CNNs tend to learn low-level features (*e.g.*, in the visual domain: edges, corners, contours) at their first layers. At the intermediate and final layers, the combination of these filters helps to extract more complex features, resulting in a vector of continuous values, referred to as *embeddings*, that might be used for classification and other tasks. In this work, we use two benchmark CNNs to extract both image and audio *embeddings* by using a transfer learning technique [Tan et al. 2018].

Based in the work of Abu-El-Haija *et al.* [Abu-El-Haija et al. 2016], we decode each video at 1 frame-per-second and feed an InceptionV3 network [Szegedy et al. 2016]. The network weights were trained in the ImageNet[5] dataset to extract image embeddings. We also make use of AudioVGG [Hershey et al. 2017] with its weights pre-trained in the Audioset[6] dataset to extract the audio embeddings. Next, we apply Principal Component Analysis (PCA) [Wold et al. 1987] to reduce the dimensions of the image embeddings and to generate a one-dimensional vector of size 1024 and 128 for audio embeddings. Finally, we concatenate both image and audio embeddings extracted in the current frame and audio window to compose the final embeddings as a sequence of the same size of the number of seconds of the video, with each timestep having 1,152 features. Each of CNNs used (AudioVGG and InceptionV3) were used exacly as published by their authors, the only modification made is that the classification layers were removed in both CNNs to obtain their respective embeddings.

Notice that, with this approach, the video is transformed into a time series, and to use it in non-sequential models (*e.g.* SVM, KNN, and MLP), we need to turn this sequence into a single feature vector that represents the whole video. In our setting, we did that by taking the average, standard deviation, min, and max values for each feature to represent the entire video. In summary, we turn the sequence of features with size n and shape n by 1,152 into a single feature with shape 1 by 4,608.

3.2. Classifiers

To do the feature classification task, we experimented both sequential models, which use extracted video embeddings in a time series format, and non-sequential ones, which use a single aggregated embeddings vector. We tested both approaches because we wanted to investigate if a more compact format such as the single embeddings vector could yield results as good or better than the full feature sequence data. As an example, one can think of a long video that has a pornography scene in just one second out of its entirety. In a non-sequential representation of the extracted features, this single pornography second could vanish among the other non-pornographic ones. Whereas, in a sequential representation, the embedding vectors of each second of the video would not be aggregated and thus could be analysed step by step. Although a sequential representation contains possibly much more redundant data than the non-sequential one, it could give the sequential classification model a important edge of detail over the less granular non-sequential ones.

For the sequential classification model, we chose the Long Short-Term Memory (LSTM) [Hochreiter and Schmidhuber 1997] networks. It has been a commonly used time series classification baseline model.

For the non-sequence models, we chose Support Vector Machines (SVM) [Cortes and Vapnik 1995], K-Nearest Neighbors (KNN) [Peterson 2009], and Multilayer Perceptron (MLP) [Haykin et al. 2009]. Among all of the experimented models, the *Support Vector Machine (SVM)* is the most used in the related literature. It is a classification model in which the data is mapped into a higher dimension input space, where an optimal separating hyper-plane is constructed. The second model, *K-Nearest Neighbors (KNN)* uses distance measure between training samples so that the k-nearest

[5] `http://www.image-net.org/`
[6] `https://research.google.com/audioset/`

neighbors always belong to the same class, while samples from different classes are separated by a large margin. It was chosen because it used also by related work, although it is a simple classification method. The third model is the *Multilayer-perceptron (MLP)*, which contains layers of nodes: an input layer, an output layer and various hidden layers in between. This one was selected because it is also commonly used as a final classifier on deep neural networks. Lastly, *Long short-term memory (LSTM)*, different than the feed-forward neural networks, process the entire sequences of data using feedback connections.

In next the Section, we present our author-made dataset. Then, Section 5.3 shows the performance of the aforementioned models on the classification of feature vectors extracted from this dataset.

4. Dataset

We use two datasets: one for training and validation, and one for testing. The training dataset was the *NDPI pornography-2k* dataset [Moreira et al. 2016] that contains 1,000 pornography videos and 1,000 non-pornography videos. Those non-pornography videos are comprised of "hard" and "easy" videos according to the likelihood of misclassification. Some examples of "hard" videos are those with high amounts of exposed skin, such as swimming and sumo fighting videos. For testing, we compiled a set of videos, called *RNP+Pornography set*. It consist of 1,976 videos from different Brazilian Federal institutions hosted in the RNP educational video repository[7] combined with a sample of 1,976 pornographic videos from the Xvideos Database.[8]

5. Experiment

In this experiment, our objective is to attest to the quality of our video *embeddings*. We evaluate the performances of popular baseline classifiers over the video *embeddings* that were extracted from a subset of the original dataset described in Section 4. Then we test the best performing classifier in the *RNP+Pornography* set. In the next subsections, we discuss the experiment setup, used metrics, and our findings. In Subsection 5.1 we describe the training configuration for each model. Next, in Subsection 5.2 we describe the evaluation metrics. And finally, in Subsection 5.3 we present our empirical findings.

5.1. Setup

We shuffled and then split the Pornography-2k dataset into approximately 90%, 5%, and 5% for the training, validation, and test sets, respectively. Each of those sets also has equally balanced examples of each class. That is why we selected only YouTube videos for the educational class and porn videos for the pornography class. Each set was structured as a collection of batches, each batch with 20 samples. For the classifiers, we use the following hyper-parameters:

1. *SVM* hyper-parameters: C: 1.0, decision function shape: 'ovr', degree: 3, gamma: 'scale', kernel: 'rbf', max iterations: -1, random state: None, shrinking: True and tolerance: 0.001.

[7] https://video.rnp.br
[8] https://info.xvideos.com/db

2. *KNN* hyper-parameters: leaf size 30, used the *minkowski* metric, k = 5, p = 2 and uniform weights.
3. *MLP* hyper-parameters: three hidden layers, the first one with 2000 neurons and the second and third ones with 1000 neurons, *RelU* activation, *xavier* initialization, *adam* optimizer, 0.001 learning rate, and cross entropy loss function.
4. *LSTM* hyper-parameters: two hidden layers, each with 10 neurons, *tanh* activation, *adam* optimizer, 0.001 learning rate, and cross entropy loss function.

5.2. Metrics

We evaluate the models by the Precision (P), Recall (R) and F1-score for educational and pornography classes:

$$P = \frac{TP}{TP + FP} \quad (1) \qquad R = \frac{TP}{TP + FN} \quad (2) \qquad F1 = \frac{2 \times P \times R}{P + R} \quad (3)$$

Where TP, TN, FP, and FN denote the examples that are true positives, true negatives, false-positives, and false negatives, respectively. The F1-score, defined in Equation 3, measures how precise the classifier by the harmonic mean between Precision (Equation 1) and Recall (Equation 2). The F1-score represents an overall performance metric, while the precision and recall metrics can give insights on where the classifier model is doing better.

5.3. Results

Having each model fitted to the train set, we proceeded to validate each of their results on the validation set, which yielded the values of the F1 score for each class and also the mean the F1 score between both classes presented in Table 1.

Model	F1-score-educational	F1-score-pornography	Mean-F1
SVM	86,49%	83,15%	84,82%
KNN	96,15%	95,83%	95,99%
MLP	97,03%	96,97%	97,00%
LSTM	98,99%	99,01%	**99,00%**

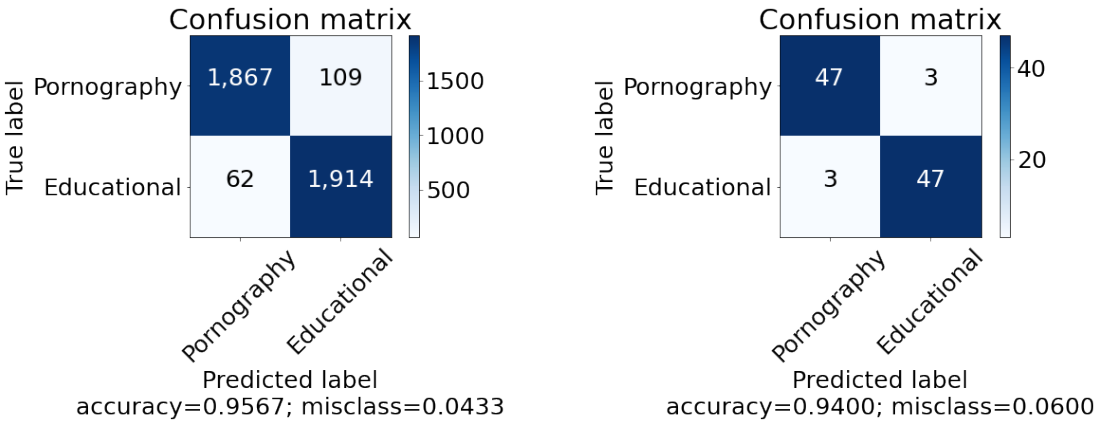
Tabela 1. The values for each metric used in the validation of the models trained on the pornography-2k dataset.

It can be noted that the SVM model had a performance significantly lower than the KNN model, which is unexpected because the KNN model is a much simpler model. Nevertheless, it is possible that the feature extractor learned a feature vector distribution that favors the KNN model. We also did not make an extensive parameter optimization on the SVM model and it is possible be that with a different set of parameters the SVM model would outperforms the KNN model.

In relation to the KNN model, the MLP model shows an increase of 0.88% and 1.14% F1-score for, respectively, the educational video class and the pornography class.

It is also noticeable that the LSTM model yielded an increase of F1-score of 1.96% and 2.04% for each class in relation to the MLP model.

As the LSTM model was the best performing model, we selected it for further evaluation on the test subset and on the RNP+Pornography set. The confusion matrix for the LSTM model on each set used is shown in Figure 2.



(a) RNP+Pornography videos. **(b) Pornography-2k videos subset.**

Figura 2. Confusion matrix for each test set.

On the *RNP+Pornography* set confusion matrix, one can notice that the number of Pornography videos that were predicted to be educational is higher than the number of pornography videos that were predicted as educational. That means that the recall for pornography videos is lower than the recall for educational videos. We strive to reduce both false positive and false negative values, but, in an educational environment, it is more important to detect pornography than it is for an educational video to be mislabeled as pornography. Because of that, one can conclude that recall is also a good metric to evaluate our model and by doing so we can compare to our related works.

On the test set confusion matrix the model achieved a balanced amount of misclassification for both classes. It also performed worse than on the *RNP+Pornography* set, which can be explained by looking into the exact kind of video in both test sets: one is a subset from the pornography-2k dataset, a dataset that was made to have "easy" and "hard" videos to classify. In its turn, the *RNP+Pornography* is a set of videos sampled from an educational platform, which is not necessarily "hard" to classify. As for the collected pornography, it is a random sample from an adult video website, that neither is necessarily "hard" to classify.

On both sets, the LSTM model performed under the expected range of our used metrics. For the test set, which had 100 videos, it yielded a precision of 94.00%, a recall of 94.00%, and an F1 score of 94.00% for pornography class. On the *RNP+Pornography* set, which had 3,952 videos, it had a precision of 96.79%, recall of 94.48%, and a F1 score of 95.62% for the pornography class.

6. Final remarks

In this work, we a presented classifier model to detect pornography content in education repositories. Our results show that the feature extraction using deep learning yield high

accuracy even in shallow models. On our validation tests (*pornography-2k*), the best performing model was the LSTM, which archived an F1 score of 98.99% for educational class and 99,01% for the pornography class. On the author-made *RNP+Pornography* set, it had an F1 score of 95,72% for educational class and 95,62% for the pornography class. Finally, on the test subset, it had a 94,00% F1 score for both classes.

We must highlight the fact that the LSTM model outperformed all the other non-sequential models. One can conclude that even though to aggregate embedding from all seconds of the video is a space efficient approach, having a sequential model to run on the embeddings as a time series can lead to a more granular, and thus, effective outcome. However, the main drawback of the the LSTM model was its recall rate, it was 96,86% for educational videos but 94,48% for pornography class. In the context of improper content detection, the recall metric for pornography content is usually more important than the recall metric for educational content.

As future work, to address this drawback we plan to search for a specialized loss function to focus on the recall for pornography class. Moreover, we also plan redo the experiments using cross-validation and inserting more difficult pornography videos (*e.g.* animation pornography) and sensitive content from other contexts, such as violence and gore.

Referências

Abu-El-Haija, S., Kothari, N., Lee, J., Natsev, P., Toderici, G., Varadarajan, B., and Vijayanarasimhan, S. (2016). Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.

Freitas, P. V. A. d., dos Santos, G. N. P., Busson, A. J. G., Álan L. V. Guedes, and Colcher, S. (2019). A baseline for nsfw video detection in e-learning environments. In *Anais Principais do XXV Simpósio Brasileiro de Multimídia e Web*, pages 357–360, Porto Alegre, RS, Brasil. SBC.

Haykin, S. S. et al. (2009). *Neural networks and learning machines/Simon Haykin.* New York: Prentice Hall,.

Hershey, S., Chaudhuri, S., Ellis, D. P., Gemmeke, J. F., Jansen, A., Moore, R. C., Plakal, M., Platt, D., Saurous, R. A., Seybold, B., et al. (2017). Cnn architectures for large-scale audio classification. In *2017 ieee international conference on acoustics, speech and signal processing (icassp)*, pages 131–135. IEEE.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8).

Lee, J., Reade, W., Sukthankar, R., Toderici, G., et al. (2018). The 2nd youtube-8m large-scale video understanding challenge. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0.

Liu, Y., Gu, X., Huang, L., Ouyang, J., Liao, M., and Wu, L. (2020). Analyzing periodicity and saliency for adult video detection. *Multimedia Tools and Applications*, 79(7):4729–4745.

Moreira, D., Avila, S., Perez, M., Moraes, D., Testoni, V., Valle, E., Goldenstein, S., and Rocha, A. (2016). Pornography classification: The hidden clues in video space–time. *Forensic science international*, 268:46–61.

Moreira, D., Avila, S., Perez, M., Moraes, D., Testoni, V., Valle, E., Goldenstein, S., and Rocha, A. (2019). Multimodal data fusion for sensitive scene localization. *Information Fusion*, 45:307–323.

Peterson, L. E. (2009). K-nearest neighbor. *Scholarpedia*, 4(2):1883.

Singh, S., Kaushal, R., Buduru, A. B., and Kumaraguru, P. (2019). KidsGUARD: Fine Grained Approach for Child Unsafe Video Representation and Detection. In *Proceedings of the 34th Annual ACM Symposium on Applied Computing*.

Song, K. and Kim, Y.-S. (2020). An enhanced multimodal stacking scheme for online pornographic content detection. *Applied Sciences*, 10(8):2943.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., and Liu, C. (2018). A survey on deep transfer learning. In *International Conference on Artificial Neural Networks*, pages 270–279. Springer.

Wold, S., Esbensen, K., and Geladi, P. (1987). Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52.