


FW: המשתמש איליה רחלבסקי עידכן את ההגשות שלו עבור המטלהדו"ח התקדמות ומצגת. 



Shery Edelson Cohen

איליה שלום, יש להעביר דוח התקדמות חתום ע"י המנחה, (אם האישור ניתן בי"ל יש להוסיף עמוד נוסף לעבודה- בו צילום מסך המיילים). בבקשה להחתיים ולהעלות שוב למודל אמתין ל



Rakhlevski Ilia

שלום אורן, האם אתה יכול לחתום או לתת אישור באימייל? תודה איליה



O.D [G]

היי,אני מאשר את ההגשה אורן



School of Software Engineering: Intelligent Systems

Detection of violence against children in videos

A project report submitted toward the degree of
Master of Science in Intelligent Systems

Student name: Ilia Rakhlevski

Supervisor: Dr. Oren Dinai

Advisor: Dr. Yehudit Aperstein

Date: 30/06/2022

Acknowledgments

I would like to thank the following people for helping with this project:

Thank you to my supervisor, Dr. Oren Dinai, for providing guidance and feedback throughout this project.

I would like to thank my Advisor: Dr. Yehudit Aperstein for her support and guidance during the running of this project.

I would particularly like to thank my wife and my son for their help with the videos creation.

Abstract

Violence against children is a world spread phenomenon. There are different forms of such violence: neglect, physical, sexual abuse. Violence against children has many negative consequences for physical, emotional, and psychosocial development. In this paper we are focused on physical violence and its detection.

This project proposes a physical violence detecting method based on indoor surveillance cameras. The cameras capture video streams, these streams are classified by using a deep-learning Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) based approach for violence detection by learning the detailed features in videos. CNN is used for feature extraction and LSTM is used to classify video based on those features. As a CNN feature extraction we use ResNet152V2 pre-trained model.

The novelty of this project is synthesized data. In this project we are focused on specific type of the violence of adults against children. There are no available data sets match of this situation and there are not enough relevant videos on the Internet and most of such videos are in low quality. So, we are going to use reborn dolls, which are very similar to real children.

We have reached various classification accuracies up to 86% for “violent” frames of the tested videos.

Table of Contents

Acknowledgments	3
Abstract	3
Abbreviations	5
List of Figures	6
List of Tables	6
Introduction	7
Literature Review	8
Contribution	10
Data Description	11
Methodology	16
Results	23
Discussion	26
References	28
Appendices	32
Project proposal	32
Project proposal presentation	40
Project progress report	57
Project progress presentation	88
Project final presentation	129
Project files description	160
Poster	162
תקציר	163

Abbreviations

biLSTM	Bidirectional Long Short-Term Memory
CNN	Convolution Neural Network
DNN	Deep Neural Network
FPS	Frames per second
GT	Ground Truth
IoU	Intersection over Union
LSTM	Long Short-Term Memory
mAp	Mean average precision
MNIST	Modified National Institute of Standards and Technology
MPEG	Moving Picture Experts Group
OKS	Object Keypoint Similarity
RELU	Rectified Linear Units
ResNet	Residual Neural Network
RGB	Red, Green, Blue
RNN	Recurrent Neural Network
SVM	Support Vector Machine

List of Figures

Figure 1. Example of frames sequence	11
Figure 2. Example of violent scenes	12
Figure 3. Example of non-violent scenes.....	12
Figure 4. Reborn baby dolls.....	13
Figure 5. Example of videos augmentation	14
Figure 6. Videos example for Convolutional LSTM experiment 2	16
Figure 7. Implemented model	19

List of Tables

Table 1. Results of the test videos run	24
---	----

Introduction

Violence against children is one of the biggest problems affecting families and societies. It happens all over the world, in all countries and societies; all too often it happens in the family.

Violence exists in schools, institutions and in the streets. In this project we will focus on the problem of reducing violent behavior in kindergarten. When violence occurs in a Kindergarten it often remains unknown to other people, because a child sometimes is afraid to talk about it with its parents or cannot speak about it because of his/her age.

This is a common phenomenon and appears often in the newspaper and courts.

Examples of recently reported cases of violent behavior in kindergarten [14] [15][16].

Our goal is development of an application that will help detect frames of violence in video streams. After the violence detection, an alert or a summary could be extracted to the adjacent security department (or to the parents) to yield an action.

The input of the application will be video stream - sequence of frames, that contains any interaction between an adult and children in kindergarten. Part of these sequences will contain violent behaviors.

The output of the application will be video stream compiled from the frames with violence detected in the input video streams.

Training and testing data will be video streams taken from the Internet or semi-synthetic created videos. Some parts of them contain scenes with violence while the other parts contain non-violent ones.

The quality of the solution is measured by the quantity of correctly detected violent frames. This can be measured by IoU (Intersection over Union [9]) of the predicted intervals vs the GT frames.

Literature Review

All the projects/articles that we have found are related to common violence and are not focused on a specific type of violence of adults against children. We learned the methods that were used in these approaches for possible use of them in our project.

In [13] discussed methods for violence detection. The authors paid more attention to the exploration of traditional detection methods ranging from the general interactional violence to crowd violence. They proposed to extract two kinds of low-level visual features (LHOG and LHOF) from the motion regions instead of extracting descriptors around the interest points. After that, the low-level features were processed under the traditional BoW framework and then predicted by SVM classifier. The data they used included 1268 videos from 3 datasets. The first dataset, Behave [26] – included 22 labeled video clips containing violence scenes. The second included 1000 videoclips collected from hockey games. The third one included 246 videos presenting crowd violence behavior. They have reached various classification accuracies between 94 and 100% from all these 3 datasets.

Nievas et al. in [8] assessed the performance of modern action recognition approaches for the recognition of fights in videos, movies and video-surveillance footage.

In this work they introduced a fight dataset and used two of the best action recognition methods that were then available (STIP [6] and MoSIFT [12]) to assess the performance of fight detection. The primary contribution of their paper was two-folded.

First, it was shown that one can construct a versatile and accurate fight detector using local descriptor approaches. Second, they presented a new dataset of hockey videos containing fights and demonstrated that their proposed approach can reliably detect violence in sports footage, even in the presence of camera motion. The methods that were used are HIK, STIP, SIFT [23]. The data they used included a 1000-video collection of NHL hockey games and 200-clips of scenes from action movies. These fight datasets have reached various classification accuracies up to 91%.

In [1] proposed a method for automatic violent behavior detection designed for video sensor

networks.

It consisted of a deep neural network followed by a time domain classifier. This allows separation of time domain and spatial processing.

In contrast with other approaches, the deep neural network input is fed exclusively with motion vector features extracted directly from the MPEG encoded video stream.

The novelty of their approach was represented by exclusively using, as input for the DNN, the motion features extracted from MPEG stream. Using the features embedded in MPEG stream lets them avoid optical flow computation. Methods that were used are DNN, MPEG flow vector and Time Domain Filter. The data they used included Behave [26] dataset and two clips with violence, they reached various classification accuracies up to 87%.

In a study by [10] they explored and dived deep into leveraging the potential of extracting salient features from the frames which then have been used in detecting violence in the videos. The authors have experimented with three pre trained ImageNet models VGG16, VGG19 and ResNet50. The extracted features from each of the frames have been fed into a fully connected network FCN. In another experiment the extracted features from 30 frames at a time and have been given to an LSTM network as an input sequence. They have constructed a CNN model as well to compare the saliency of the extracted features with other pre-trained models.

The features extracted by the ResNet50 pre-trained model proved to be more salient than the other models' classification, these features provided more accurate results. The data they used included the dataset had been collected from different video sharing website like YouTube and social networking platforms such as Facebook and Twitter. It included 110 videos for each class, 220 videos in total and they reached various classification accuracies up to 97%.

Contribution

In this project we are focused on a specific type of violence of adults against children. Often an adult beats a child and the child does not offer any resistance. There are no available data sets match this situation. It is very difficult to obtain a real data for privacy reasons. We are going to use both classic methods and state of the art ones. Also, we are going to create semi-synthetic data in a controlled environment.

Our approach for solving this problem is to build a model based on convolutional neural networks for videos. While developing we will try to use different architectures, both classic (for example a CNN with LSTM), and new ones (for example a CNN with Transformer [11]).

In the first phases of work, we would consider using transfer learning due to lack of large dataset for adult-child violence.

For training/testing data we are going to use videos from the Internet, movies, datasets (For example, UCF101/50 [25]) containing violent/non-violent scenes.

Also, we want to synthesize videos containing violence. For this purpose, are going to use reborn dolls [27], which are very similar to real children. This will be done since there are not enough relevant videos on the Internet and most of such videos are in low quality.

Therefore, we are going to create videos in which adult beats a doll that is similar to a child. We think that it is more important to detect/classify the motion of an adult that is beating a child since usually the child reacts relatively passively. Also, we will consider using pre-trained models for activity recognition.

The project will be developed on Python programming language using PyTorch, Keras and Tensorflow libraries.

Data Description

The data used in the project is video streams. See Figure 1.



Figure 1. Example of frames sequence

Format of video streams used in the project is MP4.

While a video stream processing the frames are grabbed from the video. It is created an array of the frames. This array serves as an input of the model.

The original video stream can have any frame size and FPS (frames per second). After converting of the video stream into array of the frames we change the frames size according to the model input. Current FPS is 30 but this parameter will be adjusted according to the results of the experiments so we can use part of the frames: each second, each third etc.

Video streams contain violent/non-violent scenes. At this stage we defined two types of violent behavior which will be detected - hitting a child by hand and by leg, see Figure 2. Extra types of violence can be added at late stages.



Figure 2. Example of violent scenes

Non-violent actions are the actions that do not contain the actions that are defined as violent.

It can be any action, but it should use the actions that are similar to violent ones.

See Figure 3. For example: dances, active games and sport.



Figure 3. Example of non-violent scenes

The videos are found on the Internet or synthetically created.

Because of lack of videos containing scenes of violence of adults against children we decided to create synthetical videos. For this purpose, we use reborn baby dolls – the dolls that are very similar to human infants [27]. See Figure 4. Using these dolls, we created both violent and non-violent videos.



Figure 4. Reborn baby dolls

Our goal to use such videos to train the model to detect the movements that can be considered as violence. The model must be trained to classify video according to violent movements only. Elements like background, clothes, beating adult or beaten child, lighting etc. must not have influence on classification results.

So, we make videos under these conditions. Different clothes of the humans and the dolls, background, interior, lighting, poses of the humans and the dolls, vertical angles. Most videos are synthetically created with the dolls, but also some real videos from the Internet will be added to train/test datasets. We will use short videos (20 frames, 30 fps) for training/testing. Videos must contain evident violent actions only.

During each video session must be made videos of both types: with violence and no violence. The last requirement is very important. The model must learn to classify videos according to

movements only. It means that during each video session, must be created videos containing both violent and non-violent movements, but the rest of the parameters must be the same. These parameters are background, interior, lighting, distance, people, dolls, clothes, poses etc. Because sometimes we do not have enough data, especially containing violence, it is possible to use data augmentation.

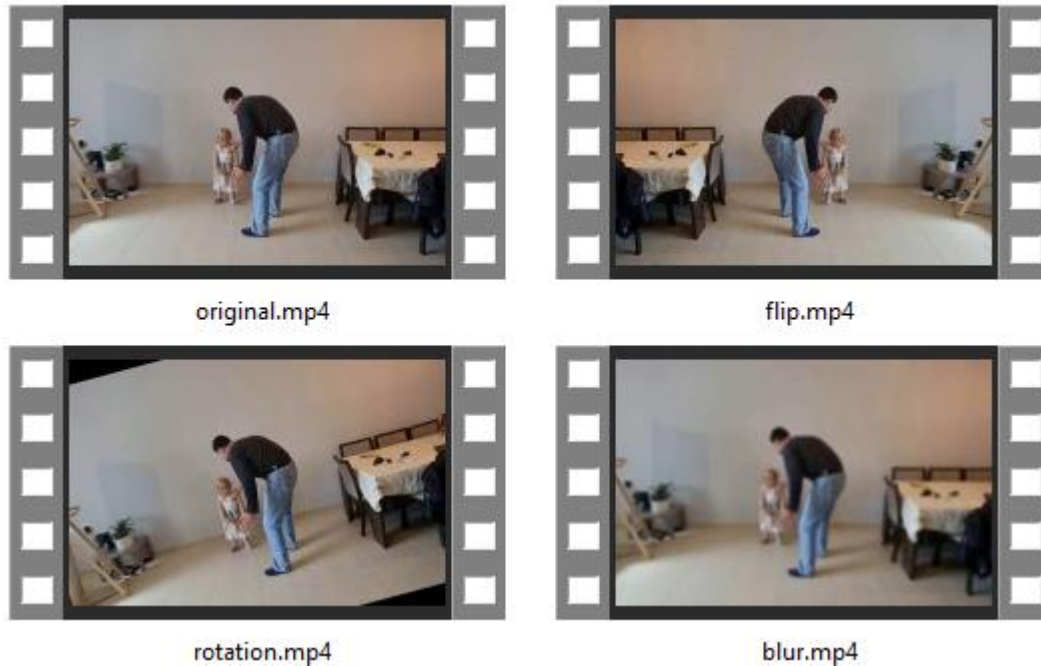


Figure 5. Example of videos augmentation

At this moment we use flipping for some videos.

At the later stages we can use other methods: changing brightness/contrast, blurring, reversing most non-violent videos, rotation. See Figure 5.

It is used for real videos (found on the Internet). Their number is limited.

At this stage we do not perform labeling of the data intended for training/validation.

We sort the video files according to their content: violent or non-violent and put them into the relevant directory. While loading they were labeled automatic according to the directory.

Labeling of the video files intended for testing will be performed at late stages.

Most of the videos that are used for training are synthetical. We used mobile phone camera to take videos. Video format: 1920x1080, 30 fps.

For each video session we perform several actions. Camera must be stable during the whole video session. Only humans/dolls movements can change. Each video session must contain both types of actions: with/no violence. Each video stream created during the video session is divided into small videos of size 20 frames. All the small videos are sorted according to their content (violent/non-violent).

We perform such video sessions many times with different background, interior, distance from the camera, clothes of the humans and the dolls, lighting.

We do it to improve the model generalization ability.

Methodology

We chose the CNN+LSTM. This is a combination of two architectures CNN and LSTM. First the input data is sent to convolution neural network and its output is sent as input to LSTM network.

The input of the network is a video, sequence of 2D images. The output is predicted class.

This architecture is a combination of CNN (convolutional neural network) [2], [3] and LSTM (Long Short-Term Memory).

CNN is used for feature extraction and LSTM is used to classify video based on those features.

So, here as we have multiple frames, we want to apply convolution operation at the same time to all the frames to learn features in each layer simultaneously, so that in the other deeper layers we learn other features on top of the previously learned features. It is using the accuracy metrics [19] that calculates how often predictions equal labels.

We performed an experiment with this architecture. For this experiment we used dataset of 340 videos with/without violence.

275 – train, 31 – validation, 34 – test. See Figure 6.



Figure 6. Videos example for Convolutional LSTM experiment 2

The input is video streams – sequence of 2D frames. All the frames were resized to 160X90 before processing.

The output is predicted class – with violence/no violence.

Train on 380 samples, validate on 43 samples, batch size is 2, number of epochs is 40 and the learning rate is 0.01.

Training/testing results

Test loss: 2.05 / Test accuracy: 0.80

	precision	recall	f1-score	support
0	0.73	0.57	0.64	14
1	0.83	0.91	0.87	33
accuracy			0.81	47
macro avg	0.78	0.74	0.75	47
weighted avg	0.80	0.81	0.80	47

This architecture can be used for violence detection without using of any extra technologies. We rated this architecture - high priority.

At this stage we continue with CNN+LSTM. For this architecture we can use pre-trained model. A pre-trained model is a model created and trained by someone else to solve a problem that is similar to ours.

In our project we use two algorithms. The first algorithm is assigned for real long videos processing. For testing and real videos classification we will use long videos processing with a windowing (sliding window) algorithm. The window length is 20 frames and the step is 5 frames. These values are approximate and will be corrected during the development process. The second algorithms is assigned for violent behavior detection. Using a technology CNN+LSTM that detects violence. It is invoked for each chunk of the data received from the sliding window algorithm. If violence is detected then the range of the frames sequence (start/end of the window) is stored. After the window sliding has been finished all the overlapped ranges are merged. For each range of frames create video as output.

During final architecture development we used advices from a blog of Andrej Karpathy “A Recipe for Training Neural Networks” [24].

As mentioned above we continue with the CNN+LSTM architecture. See Figure 17. We will split video input into frames, then we use them as input of the pre-trained CNN model. The CNN model will learn both the appearance of invariant features and the local motion features. The output is fed to the LSTM layer. Lastly, the fully connected layer classifies the LSTM cell outputs to the categories.

As a pre-trained CNN model, we use one of the models implemented in the Keras Library. We chose the ResNet152V2 architecture [18].

Its accuracies are Top-1 - 0.780 and Top-5 - 0.942.

It is higher than the accuracy of many other architectures: VGG16/19, ResNet50/101/152. Some architectures have larger accuracy, but they require larger image input (299x299, 331x331). The architecture ResNet152V2 requires image input 224x224. In our proposed model, we chose the 224×224 model since it has fewer overall parameters and computational power.

In our project we use Bidirectional LSTM. Bidirectional LSTM, or biLSTM, is a sequence processing model that consists of two LSTMs: one taking the input in a forward direction, and the other in a backwards direction. BiLSTMs effectively increase the amount of information available to the network, improving the context available to the algorithm. According to [7] biLSTM improves accuracy when the dataset has imbalanced classes.

In our case the class “no violence” has much more possible motions than the class “with violence”. Experiments results in [4] show that biLSTM network has higher accuracy with less epochs.

Below is described a base model. It must be corrected for each new dataset. New layers can be added or the existing layers can be removed. Also, the number of the neurons in the layer can be changed.

As a base for implementation was taken an example from Keras tutorial [17].

Implemented model description, See figure 7.

- 1) input_1 - Input layer. The layer that receives the model input – short video stream 20 frames. Each frame is 224x224x3.
- 2) time_distributed - Time distributed layer that contains ResNet152V2 model implemented by Keras library. The layer receives the same input as the Input layer and sends to the ResNet152V2 model each frame.
- 3) bidirectional(lstm) - Bidirectional LSTM receives from the ResNet152V2 model extracted features and classifies them. Number of units in each LSTM is 4.
- 4) Dense - Dense layer (fully-connected layer), output layer. Results of classification. Number of outputs is 1. Activation function is Sigmoid.

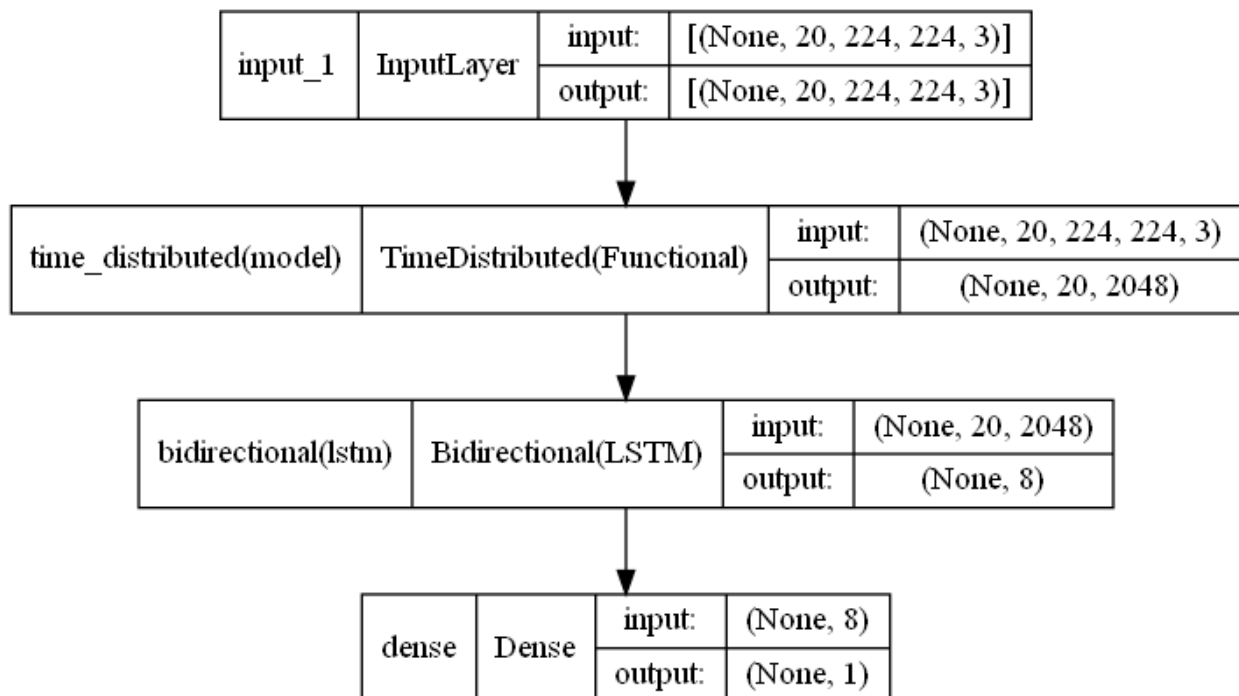


Figure 7. Implemented model

Before using the ResNet152V2 we must perform the data preprocessing.

According to Keras documentation [18] the ResNet152V2 receives as input images of size 224x224x3 and pixels values are scaled between -1 and 1. So we perform preprocessing of video data: resizing frames and pixels values scaling.

Transfer learning generally refers to a process where a model trained on one problem is used in some way on a second related problem. In our project we use the ResNet152V2 model with pre-trained weights to extract features from video frames.

Using the ResNet152V2 model.

- 1) Create the ResNet152V2 model without an output layer.
- 2) Load the weights for the pre-trained model (without output layer).
- 3) Freeze all the layers of the model that they will not be trained during the whole model training.
- 4) As the output layer is added GlobalAveragePooling2D layer. Global Average Pooling is a pooling operation designed to replace fully connected layers in classical CNNs. The idea is to generate one feature map for each corresponding category of the classification task. Instead of adding fully connected layers on top of the feature maps, we take the average of each feature map, and the resulting vector is fed directly into the “sigmoid” layer. In our case we use LSTM and full-connected layers before the “sigmoid” layer.

Overfitting is a big problem in deep learning. Also, in our project we need to overcome it.

There are several methods to reduce it. Increasing number of samples - creating extra videos.

To simplify the architecture - reducing number of layers or/and neurons. Data augmentation,

It is described in the section “Data Description”. Using Regularization methods [22].

We use some of them in our project - L1, L2 and Elastic Net regularization [21].

At this moment Elastic Net (combination L1 and L2) is used in the bidirectional (LSTM) and the dense (fully-connected layer) layers. Current λ (“lambda”) parameter is 0.000001 for both L1 and L2. This value is set experimentally.

Hyperparameters are set experimentally during model training on a small dataset containing samples that are very different from each other.

Number of layers and neurons is set experimentally. If the model is small, it can underfit.

In this case it has not enough capacity to learn the important features needed for classification. If the model is large enough it can overfit. Then the model learns superfluous features that are unnecessary. This parameter is set for the `bidirectional_1` (LSTM) layer – 4 units.

The learning rate (or step-size) is explained as the magnitude of change/update to model weights during the backpropagation training process. If it is too large it can cause the model to converge too quickly to a suboptimal solution. If it is too small it can cause the process to get stuck in local minimum. It is set to 0.0001 at the start of training. At the end stages of training it can be changed to 0.00001.

An Adam optimizer was used to train the network. Adam is a popular algorithm in the field of deep learning because it achieves good results fast [5]. All the experiments were performed with this optimizer only. It means that other optimizers should be tested.

The batch size is a number of samples processed before the model is updated. Current batch size is 64. Other values are set during training on more powerful machines.

The number of epochs should be increased until the validation loss function starts increasing. Current value of the epochs is 60-80.

At this stage there are two classes: no violence / with violence in our project. They are imbalanced. The “no violence” class has a lot more occurrences than the “with violence” one. It can cause high misclassification errors for the minority class. The training can be improved by giving different weights to both the majority and minority classes [20].

The formula to calculate this is

$$w_j = n_samples / (n_classes * n_samples_j)$$

w_j is the weight for each class (j signifies the class)

n_samples is the total number of samples or rows in the dataset

n_classes is the total number of unique classes in the target

n_samples_j is the total number of rows of the respective class

Model training process includes several steps.

- 1) It is taken as a base model or its modification.
- 2) We train this model.
- 3) During the model training we follow the parameters both “loss” and “accuracy”.

The training is performed as long as the “loss” parameter is decreasing and the

“accuracy” parameter is increasing (can be stable). Both increasing and decreasing movement can be sinusoidal.

- 4) The model is saved after each epoch. It gives us the opportunity to choose a desired model.
- 5) If the model stopped improving, then the learning rate should be updated and the training should continue.
- 6) If the model stopped improving finally, we stop the training.
- 7) We change the model: layers, number of neurons and repeat the steps 2-6. Our goal is finding the model with minimal capacity that learns the training set.
- 8) We try to use other methods of regularization to improve the model performance: L1/L2, dropout. Repeat the steps 2-6.

Testing of the model is performed on real videos. For processing real videos and their classification, we use the windowing algorithm. In order to measure the classification accuracy of videos containing “violent” frames the IoU (Intersection over Unit) method is used.

This parameter is more suitable for different models comparison.

Also, we use statistics on correct/incorrect predicted “violent” / “non-violent” frames.

For each video is created a file containing ranges of the “violent” frames.

After loading the file, a list of values is created. One cell of the list refers a specific frame (according to the index) and contains values: 0 – “non-violent” frame, 1 – “violent” frame.

A similar list is created during the video prediction. For two lists are calculated two new lists - overlap list and union list.

In the overlap list: for each cell its value is 1 if in both compared lists the corresponding cell has “1”. In the union list: for each cell its value is 1 if in the compared lists at least one corresponding cell has “1”.

We calculate the number of “1” for each list (overlap and union). Number of “1” in the overlap the list is divided into number of “1” in the union one. The received value is the prediction accuracy. This value is found in range [0, 1].

Results

In this section are described the result of test videos run. See Table 1. The table contains a summary of the testing. The full output is found in the file *Testing_run_results.txt*.

Video types	Statistics
Synthetical videos containing scenes with and without violence	<p>With Violence (actual): 1638 Correct predicted: 1398 (85.35 %) Incorrect predicted: 240 (14.65 %)</p> <p>No Violence (actual): 9914 Correct predicted: 8855 (89.32 %) Incorrect predicted: 1059 (10.68 %)</p>
Real videos with violence	<p>With Violence (actual): 1592 Correct predicted: 1390 (87.31 %) Incorrect predicted: 202 (12.69 %)</p> <p>No Violence (actual): 2045 Correct predicted: 693 (33.89 %) Incorrect predicted: 1352 (66.11 %)</p>
Real videos without violence	<p>With Violence (actual): 0 Correct predicted: 0 (100 %) Incorrect predicted: 0 (0 %)</p> <p>No Violence (actual): 30477 Correct predicted: 29120 (95.55 %) Incorrect predicted: 1357 (4.45 %)</p>
Videos that do not contain humans	<p>With Violence (actual): 0 Correct predicted: 0 (100 %) Incorrect predicted: 0 (0 %)</p> <p>No Violence (actual): 184 Correct predicted: 184 (100.0 %) Incorrect predicted: 0 (0.0 %)</p>
Summary – all types	<p>With Violence (actual): 3230 Correct predicted: 2788 (86.32 %) Incorrect predicted: 442 (13.68 %)</p> <p>No Violence (actual): 42620 Correct predicted: 38852 (91.16 %)</p>

	Incorrect predicted: 3768 (8.84 %)
--	------------------------------------

Table 1. Results of the test videos run

Video types - types of the video to be tested.

Statistics – statistic data, summaries of each type:

With Violence - actual quantity of frames containing violence.

No Violence - actual quantity of frames that do not contain violence.

Correct predicted - quantity of correct predicted frames.

Incorrect predicted - quantity of incorrect predicted frames.

For the testing purpose we used videos of several types.

Synthetical videos – they were created under similar conditions as the videos from the training dataset using reborn dolls containing scenes with violence and without one.

Real videos with violence – real videos containing scenes of violence against children.

Real videos without violence - real videos that do not contain scenes of violence. As a rule, we used real videos from kindergartens containing scenes of dances and dynamic games.

Videos that do not contain humans - videos that contain rooms with furniture only.

Total: 120 video files, 45850 – frames, ~25.5 minutes duration. 86 % “violent” and 91 % “non-violent” frames are correct predicted.

Synthetical videos: 85 % “violent” and 89 % “non-violent” frames are correct predicted.

Real videos with violence: 87 % “violent” and 33 % “non-violent” frames are correct predicted.

Real videos without violence: 95 % “non-violent” frames are correct predicted.

Videos with empty rooms: 100 % “non-violent” frames are correct predicted.

For “non-violent” frames the results are better. It can be explained by the fact that the quantity of the “non-violent” frames is larger than the quantity of the “violent” ones. It relates to both the training dataset and the testing one.

The worst results for the “violent” frames are shown by the videos: *Test_12.mp4*, *Test_73.mp4*, *Test_74.mp4*, *Test_75.mp4*, *Test_99.mp4*. No “violent” frames were predicted. In the case of the videos *Test_12.mp4* and *Test_73.mp4* we can explain it that the children were kicked by leg.

It can be caused due to lack of similar videos in the training dataset. Another reason for these

results can be low quality of the videos, except of the *Test_99.mp4*.

The worst results for the “non-violent” frames are shown by the video: *Test_9.mp4*, *Test_10.mp4*, *Test_28.mp4*, *Test_86.mp4*, *Test_87.mp4*, *Test_88.mp4*, *Test_89.mp4*, *Test_93.mp4*, *Test_96.mp4*, *Test_97.mp4*. In most cases (except of *Test_28.mp4*) is talked about videos containing violence scenes that are correct predicted. The “non-violent” frames are found near the “violent” ones and they are predicted as “violent” too. In the video *Test_28.mp4* there are no violence scenes, but humans’ movement is very similar to hits. For the real videos good results were received for “violent” frames. The best results for “violent” frames recognition were received for those videos where were used sequences of hits: several hits, quickly following one another. For example, *Test_7.mp4*, *Test_9.mp4*, *Test_13.mp4*, *Test_76.mp4*. On the other side these videos have bad results for “non-violent” frames. The “non-violent” frames that are located near the “violent” ones often are recognized as the “violent”. These videos are short and have a small number of “non-violent” frames. Thus, number of incorrect predicted “non-violent” frames is very high. Results of the testing are relative good, in the main for those videos that are similar to the videos from the dataset.

Discussion

There are differences between our project and the existing ones. We found on the Internet several examples of human actions recognition implementations. All these projects have two features.

In most cases it is enough to use one frame to perform recognition. We need to take one frame from a video to recognize a horse or bike on the frame. We do not need to recognize the action. Also, in all cases it talks about certain actions. Each action is one class.

Our project is different. We must recognize action. We need all the frames from a video for this purpose. We have two classes only. But the “violence” class can have many actions.

The “non-violence” class has an endless number of actions.

During the performing this project, we wanted to reach several goals. The essential thing is to make the model to learn the most important features.

According to our experience a model tries to use simple features to recognize a video. It can be furniture, clothes etc. Example: if the “violent” videos are created in one place and the “non-violent” videos are created in another place then the trained model will learn from the objects in the environments and not from the movements.

Our main task is that the model will learn from the human movement only. To meet this target, during each video session we created videos containing both violent and non-violent movements. The rest of the parameters (background, clothes, distance etc.) remained the same. Another important goal is to achieve generalization. The trained model must learn to recognize humans (reborn dolls) on the video and to classify their movement under different conditions: poses, clothes, gender, hairstyle etc. After many experiments we came to the conclusion that it will need thousands of videos to train the model so it would generalize well.

The dataset must contain not only synthetically generated videos but also real ones.

During the execution of the project, we encountered absence of real videos containing scenes of violence against children.

To continue this work, we need to increase the dataset. We must add new real videos containing scenes of violence against children.

Use more different reborn dolls and humans, clothes, backgrounds etc.

To create scenes with many humans and reborn dolls. Till now we have used two humans only (one human and one reborn doll).

To add extra augmentation techniques: saturation, changing brightness etc. Also, to apply the implemented methods to all of the videos in the dataset. At this moment it has not been done. It will increase the size of the dataset.

References

Academic Articles

1. Baba, M., Gui, V., Cernazanu, C., & Pescaru, D. (2019). A Sensor Network Approach for Violence Detection in Smart Cities Using Deep Learning. *Sensors*, 19(7), 1676.
<https://doi.org/10.3390/s19071676>
2. Bačanić Džakula, N. (2019). Convolutional Neural Network Layers and Architectures. *Sinteza 2019-International Scientific Conference on Information Technology and Data Related Research*, 445–451.
3. Ghosh, A., Sufian, A., Sultana, F., Chakrabarti, A., & De, D. (2020). Fundamental concepts of convolutional neural network. In *Recent Trends and Advances in Artificial Intelligence and Internet of Things* (pp. 519–567). Springer.
4. Graves, A., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM networks. *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, 4, 2047–2052.
5. Kingma, D. P., & Ba, J. (2017). Adam: A Method for Stochastic Optimization. *ArXiv:1412.6980 [Cs]*. <http://arxiv.org/abs/1412.6980>
6. Laptev, I. (2005). On Space-Time Interest Points. *International Journal of Computer Vision*, 64(2), 107–123. <https://doi.org/10.1007/s11263-005-1838-7>
7. Malki, Z., Atlam, E., Dagnew, G., Alzighaibi, A. R., Ghada, E., & Gad, I. (2020). Bidirectional Residual LSTM-based Human Activity Recognition. *Computer and Information Science*, 13(3), 1–40.
8. Nieves, E. B., Suarez, O. D., García, G. B., & Sukthankar, R. (2011). Violence detection in video using computer vision techniques. *International Conference on Computer Analysis of Images and Patterns*, 332–339.

9. Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., & Savarese, S. (2019). Generalized intersection over union: A metric and a loss for bounding box regression. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 658–666.
10. Sumon, S. A., Goni, R., Hashem, N. B., Shahria, T., & Rahman, R. M. (2020). Violence Detection by Pretrained Modules with Different Deep Learning Approaches. *Vietnam Journal of Computer Science*, 07(01), 19–40. <https://doi.org/10.1142/S2196888820500013>
11. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is All you Need. *Advances in Neural Information Processing Systems*, 30. <https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html>
12. Xu, L., Gong, C., Yang, J., Wu, Q., & Yao, L. (2014). Violent video detection based on MoSIFT feature and sparse coding. *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3538–3542. <https://doi.org/10.1109/ICASSP.2014.6854259>
13. Zhou, P., Ding, Q., Luo, H., & Hou, X. (2018). Violence detection in surveillance video using low-level features. *PLoS One*, 13(10), e0203668.

Articles

14. N12—השיפה: התעללות קשה בפעוטות בגן ילדים ברמת גן. (n.d.). Retrieved November 7, 2021, from https://www.mako.co.il/news-law/2020_q4/Article-852b6a71f058571027.htm
15. הגננת כרמל מעודה הורשעה בהתעללות ובתקיפת 11 פעוטות—משפט ופליילים—הארץ. (n.d.). Retrieved November 7, 2021, from https://www.haaretz.co.il/news/law/.premium-1.9360248?utm_source=App_Share&utm_medium=Android_Native&utm_campaign=Share

16. חדשות מעריב | "הייתה לי הרגשה" | חדשות מעריב. (n.d.). Retrieved November 7, 2021, from <https://www.maariv.co.il/news/law/Article-799453>
17. *Keras Tutorial => VGG-16 CNN and LSTM for Video Classification*. (n.d.). Retrieved November 22, 2021, from <https://riptutorial.com/keras/example/29812/vgg-16-cnn-and-lstm-for-video-classification>
18. *Keras documentation: ResNet and ResNetV2*. (n.d.). Retrieved November 24, 2021, from <https://keras.io/api/applications/resnet/#resnet152v2-function>
19. *Accuracy metrics*. (n.d.). Retrieved November 19, 2021, from https://keras.io/api/metrics/accuracy_metrics/#accuracy-class
20. How To Dealing With Imbalanced Classes in Machine Learning. (2020, October 6). *Analytics Vidhya*. <https://www.analyticsvidhya.com/blog/2020/10/improve-class-imbalance-class-weights/>
21. What are L1, L2 and Elastic Net Regularization in neural networks? (2020, January 21). *MachineCurve*. <https://www.machinecurve.com/index.php/2020/01/21/what-are-l1-l2-and-elastic-net-regularization-in-neural-networks/>
22. Regularization Techniques | Regularization In Deep Learning. (2018, April 19). *Analytics Vidhya*. <https://www.analyticsvidhya.com/blog/2018/04/fundamentals-deep-learning-regularization-techniques/>
23. Lerner, S. (2020, November 28). Implementing SIFT in Python. *Medium*. <https://lerner98.medium.com/implementing-sift-in-python-36c619df7945>
24. Karpathy, A., *A Recipe for Training Neural Networks*. (n.d.). Retrieved November 20, 2021, from <http://karpathy.github.io/2019/04/25/recipe/>

Code and Data

25. Soomro, K., Zamir, A. R., & Shah, M. (2012). UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild. *ArXiv:1212.0402 [Cs]*. <http://arxiv.org/abs/1212.0402>
26. *BEHAVE Interactions Test Case Scenarios*. (n.d.). Retrieved November 12, 2021, from [\(PDF\) The BEHAVE video dataset: ground truthed video for multi-person \(researchgate.net\)](#)
27. *Reborns*. (n.d.). Reborns. Retrieved November 13, 2021, from <https://www.reborns.com>

Appendices

Project proposal



Project proposal:

Detection of violence against children in videos

School of Software Engineering: Intelligent Systems

Student Name: Rakhlevski Ilia

Supervisor: Dr. Oren Dinai

Advisor: Dr. Yehudit Aperstein

Date: 14/12/2020

Motivation

Violence against children is one of the biggest problems affecting families and societies. It happens all around the world, in all countries and societies; all too often it happens in the family.

Violence exists in schools, institutions and in the streets. In this project we will focus on the problem of reducing violent behavior in Kindergarten. When violence occurs in a Kindergarten it often remains unknown to other people, because a child sometimes is afraid to tell about it to the parents or cannot speak about it because of his/her age. This is a common phenomenon and appears often in the newspaper and courts. Examples of the recently reported cases of violent behavior in Kindergarten [19, 20, 21, 22]. Our goal is development of an application that will help to detect frames of violence in video streams. After the violence detection, an alert or a summary could be extracted to adjacent security department (or to the parents) to yield an action.

Research Problem

The input of the application will be video stream - sequence of frames, that contains any interaction between an adult and children in Kindergarten. Part of these sequences will contain violent behaviors.

The output of the application will be localization of the detected frames with violence in the video streams.

Training and testing data will be video streams taken from the Internet, movies or semi-synthetic created videos. Some parts of them contain scenes with violence and the other parts contain non-violent ones.

The quality of the solution is measured by quantity of correct detected violent frames. This can be measured by IoU (Intersection over Union [18]) of the predicted intervals vs the GT frames.

Prior Work

All the projects/articles that we have found are related to common violence and are not focused on a specific type of violence of adults against children. We learned the methods that were used in these approaches for possible using them in our project.

In 2018, Zhou et al. [3] discussed methods for violence detection. The authors paid more attention to the exploration of traditional detection methods ranging from the general interactional violence to crowd violence. They proposed to extract two kinds of low-level visual features (LHOG and LHOF) from the motion regions instead of extracting descriptors around the interest points. After that, the low-level features were processed under the traditional BoW framework and then predicted by SVM classifier. The data they used included 1268 videos from 3 datasets. The first dataset, Behave [13] included 22 labeled video clips containing violence scenes. The second included 1000 video clips collected from hockey games. The third included 246 videos presents the crowd violence behavior and they reached various classification accuracies between 94 and 100%.

Nievas et al. in 2011 [4] assessed the performance of modern action recognition approaches for the recognition of fights in videos, movies and video-surveillance footage. In this work they introduced a fight dataset and used two of the best action recognition methods that were then available (STIP [15] and MoSIFT [16]) to assess the performance of fight detection. The primary contribution of their paper was two-folded. First, it was shown that one can construct a versatile and accurate fight detector using local descriptor approaches. Second, they presented a new dataset of hockey videos containing fights and demonstrated that their proposed approach can reliably detect violence in sports footage, even in the presence of camera motion. The methods that were used are HIK, STIP [15], SIFT [14]. The data they used included a 1000-video collection of NHL hockey games and smaller a 200-clip collection of scenes from action

movies and they reached various classification accuracies up to 91%.

In 2019, Baba et al. [1] proposed a method for automatic violent behavior detection designed for video sensor networks. It consisted of a deep neural network followed by a time domain classifier. This allows separation of time domain and spatial processing. In contrast with other approaches, the deep neural network input is fed exclusively with motion vector features extracted directly from the MPEG encoded video stream. The novelty of their approach was represented by exclusively using, as input for the DNN, the motion features extracted from MPEG stream. Using the features embedded in MPEG stream let them avoid optical flow computation. Methods that were used are DNN, MPEG flow vector and Time Domain Filter. The data they used included Behave [13] dataset and two clips with violence and they reached various classification accuracies up to 87%.

In a study by Sumon et al. in 2020 [2] they explored and dived deep into leveraging the potential of extracting salient features from the frames which then have been used in detecting violence in the videos. The authors have experimented with three pretrained ImageNet models VGG16, VGG19 and ResNet50. The extracted features from each of the frames have been fed into a fully connected network FCN. In another experiment the extracted features from 30 frames at a time and have been given to an LSTM network as an input sequence. They have constructed a CNN model as well to compare the saliency of the extracted features with other pre-trained models. The features extracted by the ResNet50 pre-trained model proved be more salient than the other models' classification on these features provided more accurate results. The data they used included the dataset had been collected from different video sharing website like YouTube and social networking platforms like Facebook and Twitter. It included 110 videos for each class it was 220 videos in total and they reached various classification accuracies up to 97%.

Project Novelty

In this project we are focused on specific type of the violence of adults against children. Often an adult beats a child and the child does not offer any resistance. There are no available data sets match this problem. It is very difficult to obtain a real data for private reasons. We are going to use both classic methods and state of the art ones. Also, we are going to create a semi synthetic data in a controlled environment.

Our Approach

Our approach for solving this problem is to build a model based on convolutional neural networks for videos. While development we will try to use different architectures, both classic (for example a CNN with LSTM), and new ones (for example a CNN with Transformer [12]). In the first phases of work we would consider using transfer learning due to lack of large dataset for adult-child violence.

For training/testing data we are going to use videos from the Internet, movies, datasets (for example, UCF101/50 [7]) containing violent/non-violent scenes. Also, we want to synthesize videos containing violence. For this purpose, are going to use reborn dolls [8], that are very similar to real children. This will be done since there is not enough relevant videos in the Internet and most of such videos are in low quality. Therefore, we are going to create videos in which adult beats a doll that is similar to a child. We think that it is more important to detect/classify the motion of an adult that is beating a child since usually the child reacts relatively passively. Also, we will consider using pre-trained models for activity recognition.

Planned model working:

1. First stage: detecting/tracking human objects in a video stream. Technologies: YOLO [6], OpenPose [5].
2. Second stage: analysis of the detected human objects. Example: their location in relation to each other, their sizes. The goal: to find humans objects that can take part in violence.

3. Third stage: check if violence exists between humans from the previous stage.
Can be used neural networks: CNN, LSTM, Transformers.

The project will be developed on Python programming language using PyTorch, Keras, Tensorflow libraries.

Workplan

Estimation: 33 weeks.

1. Preparing data.
 - Collecting videos. Estimation: 4 weeks.
 - Preprocessing of training/testing data. Estimation: 2-3 weeks.
 - Labeling data. Estimation: 2 weeks.
2. Choosing methodologies, quality metrics, designing models.
 - Evaluation of different methodologies done in the literature.
Estimation: 4-6 weeks.
 - Defining of quality metrics. Estimation: 1 week.
 - Cross validation split. Estimation: 2-3 weeks.
3. Choosing baseline model.
 - Choosing of technology for detecting/tracking human objects.
Estimation: 2-3 weeks.
 - Choosing of algorithms for analysis of the detected human objects.
Estimation: 3-4 weeks.
4. Training and testing the models.
 - Choosing Benchmarks. Estimation: 1 week.
 - Evaluation and improvements. Estimation: 4-6 weeks.
5. Final model training and evaluation.
 - Choosing final model. Estimation: 1 week.
 - Training and testing. Estimation: 2 weeks.
 - Final model evaluation. Estimation: 1 week.
6. Writing final project book. Estimation: 4 weeks.

Some of the items can be performed simultaneously. For example, in the stage 1 collecting videos, their preprocessing and labeling.

First Results

Investigated existing methods that can be useful for our development and projects that were implementing these methods: OpenPose [9], Optical flow [10], YOLO [11], CNN, RNN, LSTM and Transformers [12, 17].

Bibliography and References

1. Baba, M., Gui, V., Cernazanu, C. and Pescaru, D., 2019. A sensor network approach for violence detection in smart cities using deep learning. *Sensors*, 19(7), pp.1676.
2. Sumon, S.A., Goni, R., Hashem, N.B., Shahria, T. and Rahman, R.M., 2020. Violence Detection by Pretrained Modules with Different Deep Learning Approaches. *Vietnam Journal of Computer Science*, 7(01), pp.19-40.
3. Zhou, P., Ding, Q., Luo, H. and Hou, X., 2018. Violence detection in surveillance video using low-level features. *PLoS one*, 13(10), pp. e0203668.
4. Nievas, E.B., Suarez, O.D., García, G.B. and Sukthankar, R., 2011, August. Violence detection in video using computer vision techniques. In *International conference on Computer analysis of images and patterns*, Berlin and Heidelberg: Springer, pp. 332-339.
5. Cao, Z., Simon, T., Wei, S.E. and Sheikh, Y., 2017. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7291-7299.
6. YOLO: Real Time Object Detection:
<https://github.com/pjreddie/darknet/wiki/YOLO:-Real-Time-Object-Detection>
7. Soomro, K., Zamir, A.R. and Shah, M., 2012. A dataset of 101 human action classes from videos in the wild. *Center for Research in Computer Vision*, 2(11).
8. Reborn Dolls: <https://www.reborns.com/>
9. OpenPose: <https://github.com/spmallick/learnopencv/tree/master/OpenPose-Multi-Person>
10. Optical Flow: <https://github.com/chuanenlin/optical-flow>

11. Keras Yolo 3: <https://github.com/qqwweee/keras-yolo3>
12. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I., 2017. Attention is all you need. *Advances in neural information processing systems*, 30, pp.5998-6008.
13. BEHAVE database:
<http://groups.inf.ed.ac.uk/vision/BEHAVEDATA/INTERACTIONS/>
14. SIFT: <https://medium.com/@lerner98/implementing-sift-in-python-36c619df7945>
15. Laptev, I., 2005. On space-time interest points. *International journal of computer vision*, 64(2-3), pp.107-123.
16. Xu, L., Gong, C., Yang, J., Wu, Q. and Yao, L., 2014, May. Violent video detection based on MoSIFT feature and sparse coding. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3538-3542.
17. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S. and Uszkoreit, J., 2020. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv preprint arXiv:2010.11929*.
18. Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I. and Savarese, S., 2019. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 658-666.
19. Haaretz, 2020, "הגנת כרמל מעודה הורשעה בהתעללות ובתקיפת 11 פעוטות" https://www.haaretz.co.il/news/law/.premium-1.9360248?utm_source=App_Share&utm_medium=Android_Native&utm_campaign=Share
20. Maariv, 2020, "אמא לילד מהגן בו התעללו בפעוטות: הייתה לי הרגשה, חיפשתי סימנים" <https://www.maariv.co.il/news/law/Article-799453>
21. Ynet, 2020, "חשד לפרשת התעללות נוספת: סייעות בגן ילדים בחולון נעצרו" <https://www.ynet.co.il/news/article/rkCI511KGv>

22. Mako, 2020, "חשיפה: התעללות קשה בפעוטות בגן ילדים ברמת גן",
https://www.mako.co.il/news-law/2020_q4/Article-852b6a71f058571027.htm

Project proposal presentation

PROJECT PROPOSAL

Detection of violence against children in videos

Student Name: Rakhlevski Ilia

Supervisor: Dr. Oren Dinai

Advisor:

Motivation

2

- Violence against children is one of the biggest problems affecting families and societies. There many publications in mass media about this problem [5, 6].



- According to the World Health Organization estimation [11] up to 1 billion children aged 2–17 years, have experienced physical, sexual, or emotional violence or neglect in the past year.

Iliia Rakhlevski

Motivation (cont.)

3

- When violence occurs in a Kindergarten it often remains unknown to other people .
- Our goal is development of an application that will help to detect frames of violence in video streams.
- After the violence detection, an alert or a summary could be extracted to adjacent security department (or to the parents) to yield an action.

Iliia Rakhlevski

Research problem

4

- The input of the application will be video stream - sequence of frames, that contains any interaction between an adult and children in Kindergarten.
- The output of the application will be localization of the detected frames with violence in the video streams.
- Training and testing data will be video streams taken from the Internet, movies or semi-synthetic created videos. Some parts of them contain scenes with violence and the other parts contain non-violent ones.
- The quality of the solution is measured by quantity of correct detected violent frames.

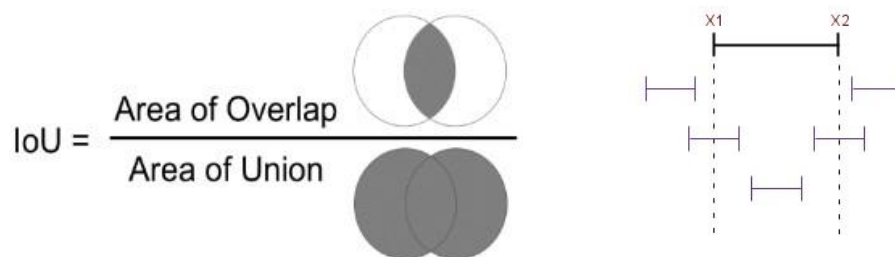
Iliia Rakhlevski

Research problem (cont.)

5

Intersection over Union

Intersection over Union is an evaluation metric used to measure the accuracy of an object detection.



Ilija Rakhlevski

Prior work

6

- All the projects/articles that we have found are related to common violence and are not focused on a specific type of violence of adults against children. [1, 2, 3, 4].
- In these projects/articles are used both classical methods (for example : SVM, CNN, LSTM) and modern ones (for examples: VGG 16, ResNet50).
- They reached various classification accuracies between 87 and 100%. Such high accuracies could be reached as a result of using very small datasets (not enough frames containing the scenes with violence) .

Iliia Rakhlevski

Project novelty

7

- In this project we are focused on specific type of the violence of adults against children.
- There are no available data sets match this problem.
- It is very difficult to obtain a real data for privacy and legal reasons.

Iliia Rakhlevski

Our approach

8

- A model based on convolutional neural networks for videos.
- We will try different architectures, both classic and new ones.
- Using transfer learning (due to lack of large dataset for adult -child violence).
- Using videos from:
 - ▣ Internet.
 - ▣ Movies.
 - ▣ Datasets containing violent/non -violent scenes.

Iliia Rakhlevski

Our approach (cont.)

9

- Synthesize videos containing violence.
Using reborn dolls, that are very similar to real children.
We assume that an adult performs most of motions, while a child does much less movements.



- Using pre-trained models for activity recognition
- Using PyTorch, Keras, Tensorflow libraries.

Iliia Rakhlevski

Our approach (cont.)

10

Planned model working:

- 1) Detecting/tracking human objects in a video stream.
Technologies: YOLO [7], OpenPose [8].
- 2) Analysis of the detected human objects.
Example: their location in relation to each other, their sizes.
The goal: to find humans objects that can take part in violence.
- 3) Check if violence exists between humans from the previous stage
Can be used neural networks: CNN, LSTM, Transformers [9].

Iliia Rakhlevski

Workplan

11

Total estimation: 33 weeks.

1) Preparing data. Estimation: 8-9 weeks.

- ▣ Collecting videos. Estimation: 4 weeks.
- ▣ Preprocessing of training/testing data. Estimation: 2-3 weeks.
- ▣ Labeling data. Estimation: 2 weeks.

2) Choosing methodologies, quality metrics, designing models. Estimation: 7-10 weeks.

- ▣ Evaluation of different methodologies done in the literature.
Estimation: 4-6 weeks.
- ▣ Defining of quality metrics. Estimation: 1 week.
- ▣ Cross validation split. Estimation: 2-3 weeks.

Iliia Rakhlevski

Workplan (cont.)

12

- 3) Choosing baseline model. Estimation: 5-7 weeks.
 - ▣ Choosing of technology for detecting/tracking human objects.
Estimation: 2-3 weeks.
 - ▣ Choosing of algorithms for analysis of the detected human objects.
Estimation: 3-4 weeks.
- 4) Training and testing various models. Estimation: 5-8 weeks.
 - ▣ Choosing Benchmarks. Estimation: 1 week.
 - ▣ Evaluation and improvements. Estimation: 4 -6 weeks.

Iliia Rakhlevski

Workplan (cont.)

13

5) Final model training and evaluation. Estimation: 4 weeks.

- ▣ Choosing final model. Estimation: 1 week.
- ▣ Training and testing. Estimation: 2 weeks.
- ▣ Final model evaluation. Estimation: 1 week.

6) Writing final project book. Estimation: 4 weeks.

Some of the items can be performed simultaneously. For example, in the stage 1 collecting videos, their preprocessing and labeling.

Iliia Rakhlevski

First results

14

Investigated existing methods that can be useful for our development and projects that were implementing these methods:

- OpenPose [8]
- Optical flow [10]
- YOLO [7]
- CNN
- RNN
- LSTM
- Transformers [9]

Iliia Rakhlevski

References

15

- 1) A sensor network approach for violence detection in smart cities using deep learning.
Baba, M., Gui, V., Cernazan, C. and Pescaru, D., 2019.
Sensors, 19(7), pp.1676.
- 2) Violence Detection by Pretrained Modules with Different Deep Learning Approaches
Sumon S.A., Goni, R., Hashem, N.B., Shahria, T. and Rahman, R.M., 2020.
Vietnam Journal of Computer Science, 7(01), pp.19-40.
- 3) Violence detection in surveillance video using low-level features.
Zhou, P., Ding, Q., Luo, H. and Hou, X., 2018.
PoS one, 13(10), pp. e0203668.
- 4) Violence detection in video using computer vision techniques.
Nievas, E.B., Suarez, O.D., García, G.B. and Sukthankar R., 2011, August.
In *International conference on Computer analysis of images and pattern*. Berlin and Heidelberg: Springer, pp.332-339.

Ilia Rakhlevski

References (cont.)

16

5) Ynet, 2020, "חשד לפרשת התעללות נוספת: סייעות בגן ילדים בחולון נעצרו"

<https://www.ynet.co.il/news/article/rkCl511KGv>

6) Maariv, 2020, "אמא לילד מהגן בו התעללו בפעוטות: הייתה לי הרגשה, חיפשתי סימנים"

<https://www.maariv.co.il/news/law/Article799453>

7) YOLO: Real Time Object Detection :

<https://github.com/pjreddie/darknet/wiki/YOLO:-Real-Time-Object-Detection>

8) OpenPose :

<https://github.com/spmallick/learnopencv/tree/master/OpenPose-Multi-Person>

Iliia Rakhlevski

References (cont.)

17

9) Attention is all you need.

Vaswani, A. et al. 2017.

Advances in neural information processing systems , 30, pp.5998 -6008.

10) Optical Flow:

<https://github.com/chuanenlin/optical-flow>

11) World Health Organization. Violence against children.

<https://www.who.int/news-room/fact-sheets/detail/violence-against-children#:~:text=Globally%2C%20it%20is%20estimated%20that,lifelong%20health%20and%20well-being>

Iliia Rakhlevski



Project progress report:

Detection of violence against children in videos

School of Software Engineering: Intelligent Systems

Student Name: Rakhlevski Ilia

Supervisor: Dr. Oren Dinai

Advisor: Dr. Yehudit Aperstein

Date: 25/07/2021

Table of changes in the original proposal

Section	Changes
Research Problem	The output of the application will be video stream compiled from the frames with violence detected in the input video streams instead of frames written to image files (this option still exists).
First Results	Sub-sections: Technologies Investigation, Data, Algorithms, Project code description have been added.

Motivation

Violence against children is one of the biggest problems affecting families and societies. It happens all around the world, in all countries and societies; all too often it happens in the family.

Violence exists in schools, institutions and in the streets. In this project we will focus on the problem of reducing violent behavior in kindergarten. When violence occurs in a Kindergarten it often remains unknown to other people, because a child sometimes is afraid to talk about it with its parents or cannot speak about it because of his/her age. This is a common phenomenon and appears often in the newspaper and courts. Examples of the recently reported cases of violent behavior in kindergarten [19, 20, 21, 22]. Our goal is development of an application that will help detect frames of violence in video streams. After the violence detection, an alert or a summary could be extracted to adjacent security department (or to the parents) to yield an action.

Research Problem

The input of the application will be video stream - sequence of frames, that contains any interaction between an adult and children in kindergarten. Part of these sequences will contain violent behaviors.

The output of the application will be video stream compiled from the frames with violence detected in the input video streams.

Training and testing data will be video streams taken from the Internet, movies or semi-synthetic created videos. Some parts of them contain scenes with violence and the other parts contain non-violent ones.

The quality of the solution is measured by quantity of correct detected violent frames. This can be measured by IoU (Intersection over Union [18]) of the predicted intervals vs the GT frames.

Prior Work

All the projects/articles that we have found are related to common violence and are not focused on a specific type of violence of adults against children. We learned the methods that were used in these approaches for possible using them in our project.

In 2018, Zhou et al. [3] discussed methods for violence detection. The authors paid more attention to the exploration of traditional detection methods ranging from the general interactional violence to crowd violence. They proposed to extract two kinds of low-level visual features (LHOG and LHOF) from the motion regions instead of extracting descriptors around the interest points. After that, the low-level features were processed under the traditional BoW framework and then predicted by SVM classifier. The data they used included 1268 videos from 3 datasets. The first dataset, Behave [13] - included 22 labeled video clips containing violence scenes. The second included 1000 video clips collected from hockey games. The third one included 246 videos presenting crowd violence behavior. They have reached various classification accuracies between 94 and 100% from all these 3 datasets.

Nievas et al. in 2011 [4] assessed the performance of modern action recognition approaches for the recognition of fights in videos, movies and video-surveillance

footage. In this work they introduced a fight dataset and used two of the best action recognition methods that were then available (STIP [15] and MoSIFT [16]) to assess the performance of fight detection. The primary contribution of their paper was twofold. First, it was shown that one can construct a versatile and accurate fight detector using local descriptor approaches. Second, they presented a new dataset of hockey videos containing fights and demonstrated that their proposed approach can reliably detect violence in sports footage, even in the presence of camera motion. The methods that were used are HIK, STIP [15], SIFT [14]. The data they used included a 1000-video collection of NHL hockey games and 200-clips of scenes from action movies. These fight datasets have reached various classification accuracies up to 91%.

In 2019, Baba et al. [1] proposed a method for automatic violent behavior detection designed for video sensor networks. It consisted of a deep neural network followed by a time domain classifier. This allows separation of time domain and spatial processing. In contrast with other approaches, the deep neural network input is fed exclusively with motion vector features extracted directly from the MPEG encoded video stream. The novelty of their approach was represented by exclusively using, as input for the DNN, the motion features extracted from MPEG stream. Using the features embedded in MPEG stream let them avoid optical flow computation. Methods that were used are DNN, MPEG flow vector and Time Domain Filter. The data they used included Behave [13] dataset and two clips with violence and they reached various classification accuracies up to 87%.

In a study by Sumon et al. in 2020 [2] they explored and dived deep into leveraging the potential of extracting salient features from the frames which then have been used in detecting violence in the videos. The authors have experimented with three pretrained ImageNet models VGG16, VGG19 and ResNet50. The extracted features from each of the frames have been fed into a fully connected network FCN. In another experiment

the extracted features from 30 frames at a time and have been given to an LSTM network as an input sequence. They have constructed a CNN model as well to compare the saliency of the extracted features with other pre-trained models. The features extracted by the ResNet50 pre-trained model proved to be more salient than the other models' classification, these features provided more accurate results. The data they used included the dataset had been collected from different video sharing website like YouTube and social networking platforms as Facebook and Twitter. It included 110 videos for each class, 220 videos in total and they reached various classification accuracies up to 97%.

Project Novelty

In this project we are focused on specific type of the violence of adults against children. Often an adult beats a child and the child does not offer any resistance. There are no available data sets match of this situation. It is very difficult to obtain a real data for private reasons. We are going to use both classic methods and state of the art ones. Also, we are going to create a semi synthetic data in a controlled environment.

Our Approach

Our approach for solving this problem is to build a model based on convolutional neural networks for videos. While development we will try to use different architectures, both classic (for example a CNN with LSTM), and new ones (for example a CNN with Transformer [12]). In the first phases of work, we would consider using transfer learning due to lack of large dataset for adult-child violence.

For training/testing data we are going to use videos from the Internet, movies, datasets (For example, UCF101/50 [7]) containing violent/non-violent scenes. Also, we want to synthesize videos containing violence. For this purpose, are going to use reborn

dolls [8], which are very similar to real children. This will be done since there is not enough relevant videos on the Internet and most of such videos are in low quality.

Therefore, we are going to create videos in which adult beats a doll that is similar to a child. We think that it is more important to detect/classify the motion of an adult that is beating a child since usually the child reacts relatively passively. Also, we will consider using pre-trained models for activity recognition.

Planned model working:

4. First stage: detecting/tracking human objects in a video stream. Technologies: YOLO [6], OpenPose [5].
5. Second stage: analysis of the detected human objects. Example: their location in relation to each other, their sizes. The goal: to find humans objects that can take part in violence.
6. Third stage: check if violence exists between humans from the previous stage. Can be used neural networks: CNN, LSTM, Transformers.

The project will be developed on Python programming language using PyTorch, Keras, Tensorflow libraries.

Workplan

Estimation: 33 weeks.

7. Preparing data.
 - Collecting videos. Estimation: 4 weeks.
 - Preprocessing of training/testing data. Estimation: 2-3 weeks.
 - Labeling data. Estimation: 2 weeks.
8. Choosing methodologies, quality metrics, designing models.
 - Evaluation of different methodologies done in the literature. Estimation: 4-6 weeks.
 - Defining of quality metrics. Estimation: 1 week.
 - Cross validation split. Estimation: 2-3 weeks.
9. Choosing baseline model.
 - Choosing of technology for detecting/tracking human objects. Estimation: 2-3 weeks.

- Choosing of algorithms for analysis of the detected human objects. Estimation: 3-4 weeks.
10. Training and testing the models.
 - Choosing Benchmarks. Estimation: 1 week.
 - Evaluation and improvements. Estimation: 4-6 weeks.
 11. Final model training and evaluation.
 - Choosing final model. Estimation: 1 week.
 - Training and testing. Estimation: 2 weeks.
 - Final model evaluation. Estimation: 1 week.
 12. Writing final project book. Estimation: 4 weeks.

Some of the items can be performed simultaneously. For example, in the stage 1 collecting videos, their preprocessing and labeling.

First Results

Investigated existing methods that can be useful for our development and projects that were implementing these methods: OpenPose [9], Optical flow [10], YOLO [11], CNN, RNN, LSTM and Transformers [12, 17].

Technologies Investigation

In this section is described technologies investigation. We want to find out which technologies, methods, libraries can be useful for our goals.

YOLO

Object detection library [11]. YOLO object detector is used to detect objects in both images and video streams. See Figure 1.

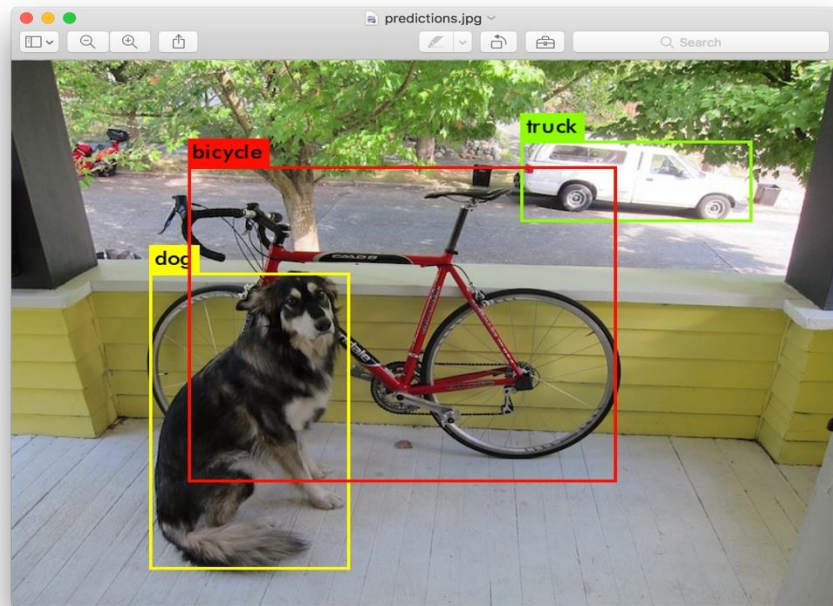


Figure 1. YOLO example

Advantages:

Can detect human objects on a stream, their sizes (in pixels) and locations (in pixels) on the image. It can help to detect if humans are found on the image, their relative sizes and relative distance between them, also to localize the region(s) where some actions between humans can occur.

Disadvantages:

- Sometimes a human is incorrect detected or not detected at all. See Figure 2.
- If two or more humans are very close, they can be detected as one human. See Figure 3.



Figure 2. YOLO does not detect humans

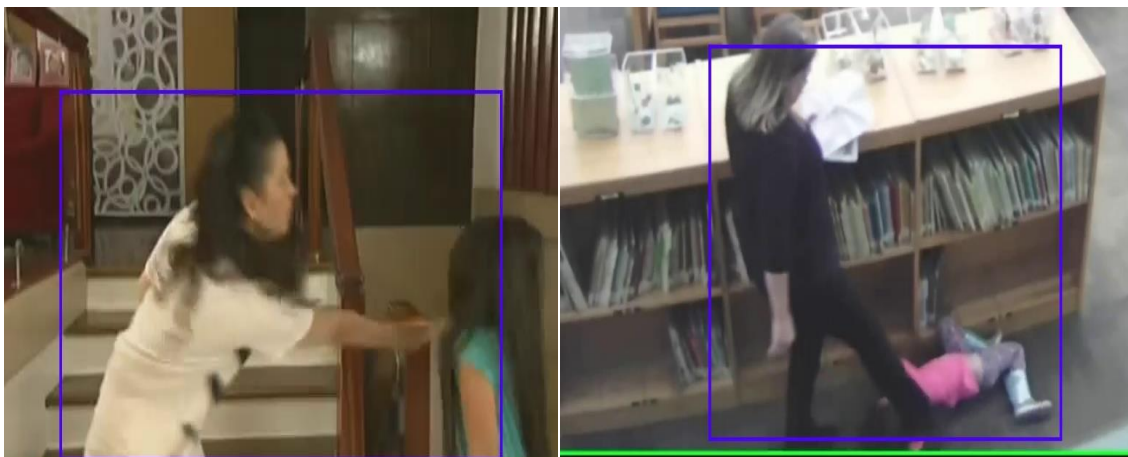


Figure 3. YOLO detects two persons as one

Conclusion:

This technology can be used to detect human objects and the region in which can occur actions between humans. It can help to detect not only the frames including violence, but also, the regions in the frames containing violence. For our goal it can be used in combination with other technology only.

Optical flow

Optical flow [10] is defined as the apparent motion of individual pixels on the image plane. See Figure 4.

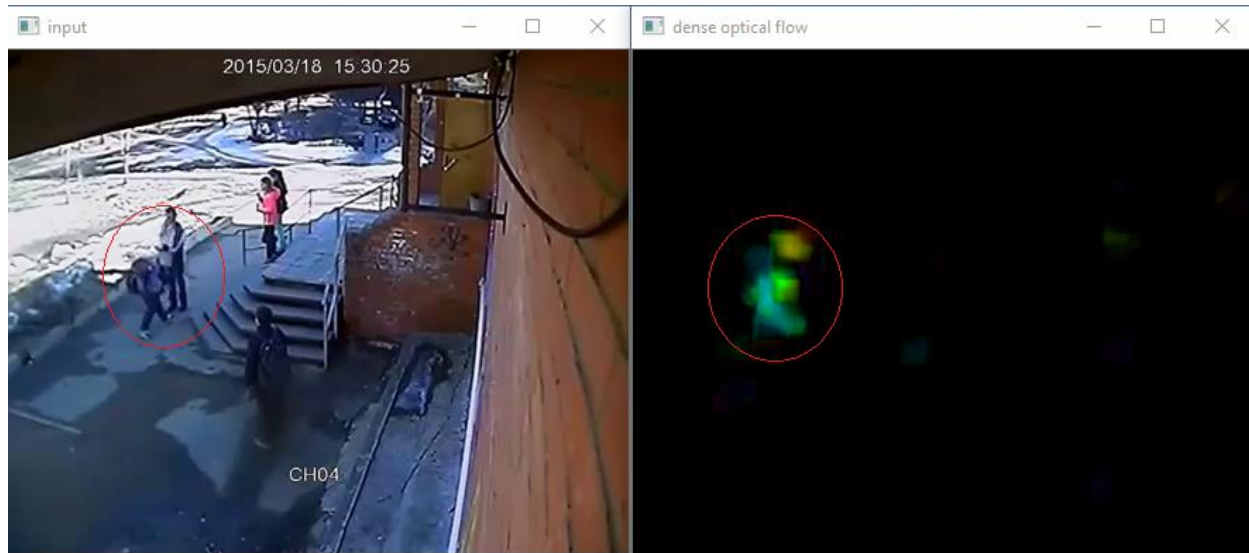


Figure 4. Optical flow. An adult man hits a child

Advantages:

Detects objects motion and its speed.

Disadvantages:

- Camera must be absolutely stationary.
- Detects all moving objects, not only humans.
- It does not detect an object if it does not move. Example: beaten child.
- Detects noise. Example: a camera is not qualitative; it can have a lot of digital noise.

Conclusion:

It can be used to detect regions where are performed fast motions that can be considered as hits. For our goal it can be used in combination with other technologies only.

OpenPose

OpenPose [9] has represented the **first real-time multi-person system to jointly detect human body, hand, facial, and foot key points on single images**. See Figure 5.

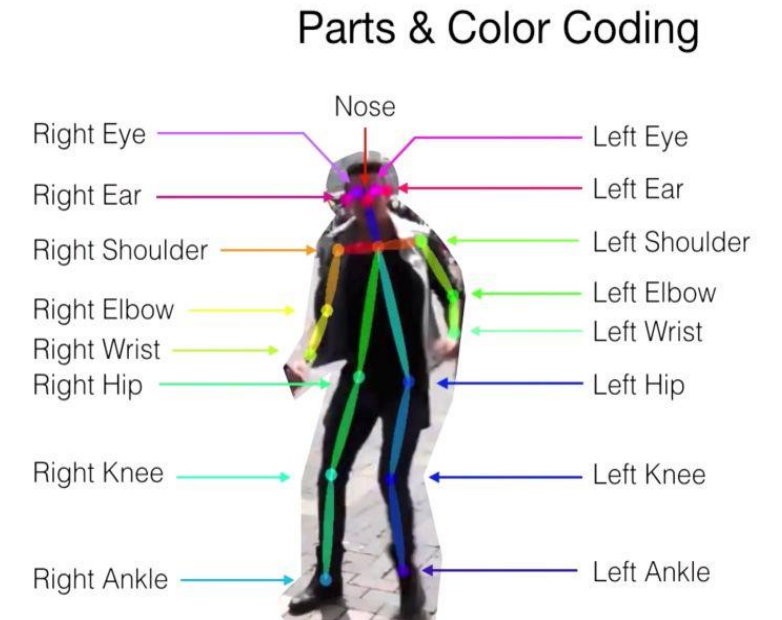


Figure 5. OpenPose detects human key points

Advantages:

Detects human objects, legs and hands, also their sizes and locations.

Hits that performed as a rule by legs and hands can be easily detected and analyzed.

See Figure 6.

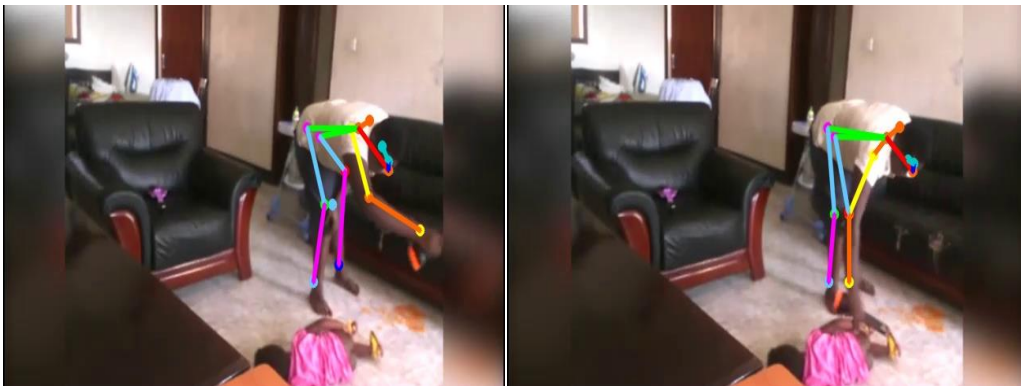


Figure 6. OpenPose detects key points on a woman body that beating a child

Disadvantages:

- Works very slow. Each frame is processed 4-7 seconds.
- Human body is not always detected. See Figure 6: the lying child is not detected.
- A key point is not always detected (even if it is visible). See Figure 7.

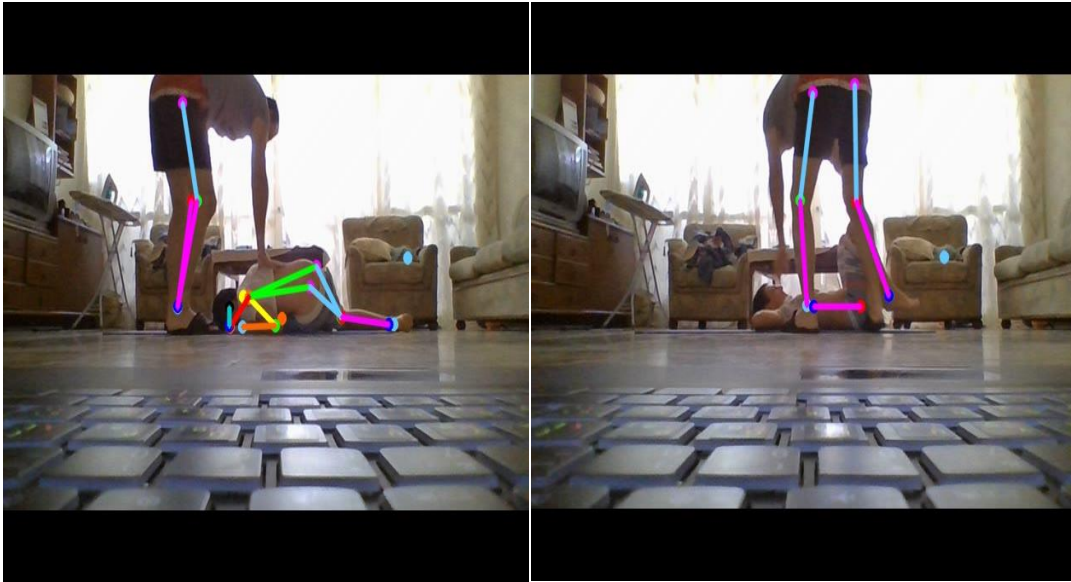


Figure 7. Left hand is visible, but not detected (right image). Right hand is visible, but not detected (left image)

Conclusion:

Because of time of processing can be used for short videos. It can be used to detect human objects and the region in which an action between humans occurs. It can help to detect movement of hands and legs that can be considered as hits.

ConvLSTM (ConvLSTM2D)

Convolutional LSTM [42]. It is similar to an LSTM layer, but the input transformations And recurrent transformations are both convolutional. According to some sources it can

Be used for:

- Feature extraction and classification [23, 24].
- Video frame prediction [25].
- Human activity recognition [26, 27].

Experiment 1:

We have used the code and the data from the example [27]. The application uses videos divided into 100 classes [28] and creates/trains a model that classifies videos (part or all of them). We have changed the code a little. Some bags have been fixed.

Only two classes (from 100) were selected for classification: biking and horse riding, because they are similar. See Figure 8. The video format of the data - 320X240, 30 fps.

While video processing the frames were compressed to 120X90. Only 80 first frames from each video were processed. Our model was trained on 189 videos and validated on 48 videos with 40 epochs. The classification accuracy is up to 90%.



Figure 8. Two classes for Convolutional LSTM experiment 1

Train/test results:

Train on 189 samples, validate on 48 samples

	precision	recall	f1-score	support
0	0.90	0.83	0.86	23
1	0.90	0.95	0.92	37
accuracy	0.90	60		
macro avg	0.90	0.89	0.89	60

weighted avg 0.90 0.90 0.90 60

Experiment 2:

For this experiment we used dataset of 340 videos with/without violence:

275 – train, 31 – validation, 34 – test. See Figure 8.



Figure 8. Videos example for Convolutional LSTM experiment 2

Train/test results:

Train on 275 samples, validate on 31 samples

	precision	recall	f1-score	support
0	0.82	0.60	0.69	15
1	0.74	0.89	0.81	19
accuracy	0.76	34		
macro avg	0.78	0.75	0.75	34
weighted avg	0.77	0.76	0.76	34

Advantages:

It can be used for violence detection without using of some extra technologies.

Disadvantages:

It detects frames containing violence only. In order to localize the regions (optional) with

violence, extra technologies or algorithms must be used.

Conclusion:

This architecture can be used for violence detection without using of any extra technologies. For localization of region containing violence must be used other methods/algorithms.

Transformers (Self-Attention)

This technology [12, 17] can be used for:

- Object detection [29].
- Video classification [30].
- Image classification [31].
- Image generation [17].

Advantages:

It can be used for violence detection without using of some extra technologies.

Disadvantages:

It detects frames containing violence only. In order to localize the regions with violence extra technologies or algorithms must be used.

Conclusion:

This architecture can be used for violence detection without using of any extra technologies. But selected implementations must be well tested. It should give to this technology lower priority. For localization of region containing violence must be used other methods/algorithms.

Mask R-CNN

It is object detector [43] is used to detect objects in both images and video streams.

See Figure 9.



Figure 9. Mask R-CNN

Advantages:

Can detect human objects on a stream, their sizes (in pixels) and locations (in pixels) on the image. It can help to detect if humans are found on the image, their relative sizes and relative distance between them, also to localize the region(s) where some actions between humans can occur.

Disadvantages:

Sometimes a human is incorrectly detected, not detected at all or two or more humans are detected as one. See the examples above.

Conclusion:

This technology can be used to detect human objects and the region in which can occur actions between humans. It can help to detect not only the frames including violence, but also, the regions in the frames containing violence. For our goal it can be used in combination with other technology only.

Conv3D

This architecture [44] can be used for video processing [39].

Experiment 1:

We have used the code and the data from the example [37]. The author developed an application that recognized 3D digits from the 3D MNIST [38]. This is a 3D version of MNIST database of handwritten digits. See figure 10.

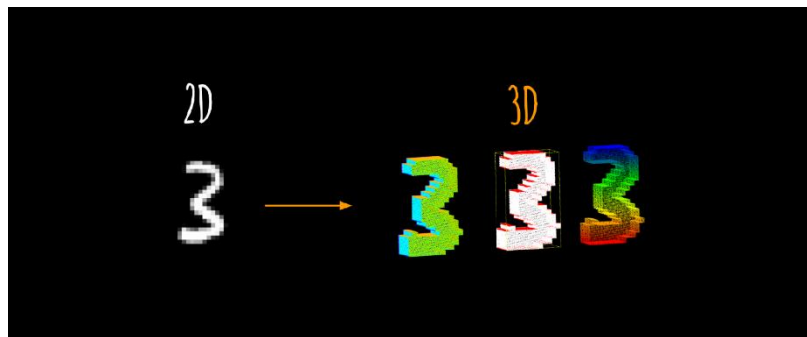


Figure 10. 3D MNIST

Accuracy of the original code is 67%. After of changes of some parameters we have succeeded to reach accuracy up to 76%.

Train/test results:

Train on 8000 samples, validate on 2000 samples

Test loss: 0.9244766535758973 / Test accuracy: 0.7689999938011169

Experiment 2:

For this experiment we used the same dataset that we used for ConvLSTM

Experiment 2. During the experiment were changed several parameters:

- Number videos in the dataset.
- Number and size of filters.
- Number of layers.

The best result that was received:

Train on 76 samples, validate on 20 samples

Test loss: 0.7750303149223328 / Test accuracy: 0.5

	precision	recall	f1-score	support
0	0.45	1.00	0.62	10
1	1.00	0.14	0.25	14
accuracy	0.50	24		
macro avg	0.73	0.57	0.44	24
weighted avg	0.77	0.50	0.41	24

Advantages:

It can be used for violence detection without using of some extra technologies.

Disadvantages:

It detects frames containing violence only. In order to localize the regions with violence extra technologies or algorithms must be used.

Conclusion:

The accuracy received in the experiments is very low.

CNN+LSTM (Conv2D+LSTM)

This is a combination of two architectures CNN and LSTM. First the input data is sent to convolution neural network and its output is sent as input to LSTM network.

Experiment:

For this experiment we used the same dataset that we used for ConvLSTM

Experiment 2.

Train/test report:

Train on 380 samples, validate on 43 samples

Test loss: 2.0506185775107526 / Test accuracy: 0.8085106611251831

	precision	recall	f1-score	support
0	0.73	0.57	0.64	14
1	0.83	0.91	0.87	33
accuracy	0.81	47		
macro avg	0.78	0.74	0.75	47
weighted avg	0.80	0.81	0.80	47

Advantages:

It can be used for violence detection without using of some extra technologies.

Disadvantages:

It detects frames containing violence only. In order to localize the regions with violence extra technologies or algorithms must be used.

Conclusion:

This architecture can be used for violence detection without using of any extra technologies. For localization of region containing violence must be used other methods/algorithms.

Investigation Conclusion

According to selected algorithms we need two types of technologies:

1. Detection humans on a frame, their locations and sizes.
2. Detection violence in sequence of frames.

Candidates for the first type:

1. YOLO
2. OpenPose
3. Mask R-CNN

Candidates for the second type:

4. ConvLSTM (ConvLSTM2D)
5. Transformers
6. CNN+LSTM (Conv2D+LSTM)
7. Conv3D
8. OpenPose
9. Optical flow

First type technology selection:

The OpenPose detects human body key points, but we do not need them for the selected algorithms. Also, it is very slow (4-7 seconds/frame). So, we reject it.

Two other technologies remain: Mask R-CNN and YOLO.

They make the same work, but Mask R-CNN performs not only detection, but also object segmentation. According to the article [36] Mask R-CNN architecture has significantly higher accuracy in determining classes in the video stream.

At the same time, in all experiments, the performance on the same hardware platform for the YOLO architecture turned out to be in three times higher than for Mask R-CNN. In the articles [40, 41] is said that processing speed of Mask R-CNN is lower than of YOLO3. These two parameters (accuracy and performance) must be tested before choosing of appropriate technology for our goal.

Second type technology selection:

The Transformers is relative new technology, basically tested in text processing. There is not any official implementation of the Transformers that can be used for video processing in any known libraries (like Keras, Torch). There are several implementations of the Transformers for image/video processing in the GitHub [33, 34, 35], but we cannot know how reliable they are so they must be tested.

While the architecture ConvLSTM is implemented in the Keras, well documented and widely used. According to the article [32] the Transformers do not outperform the CNN+LSTM. So, at this stage we reject it.

The Conv3D technology demonstrated very low results. Rejected.

The Optical flow method has many disadvantages and can be used for our purpose in combination with other methods and technologies only. At this stage is rejected.

The OpenPose is very slow. So, it cannot be used for long videos processing. Rejected.

Both the CNN+LSTM and the ConvLSTM technologies must be compared for performance and accuracy.

Chosen technologies:

- 1) Detection humans on a frame – YOLO, Mask R-CNN.
- 2) Detection violence in sequence of frames – ConvLSTM, CNN+LSTM.

Data

Data is used in our project is video streams containing violent/non-violent scenes.

The videos are found on the Internet or synthetically created. See *Synthetical videos* section. Video stream is a frames sequence, see Figure 12. We process video stream as an array of frames.

Synthetical videos

Because of lack of videos containing scenes of violence of adults against children we decided to create synthetical videos. For this purpose, we use reborn baby dolls – the dolls that are very similar to human infants [8].

See Figure 11.



Figure 11. Reborn baby dolls



Figure 12. Example of frames sequence

Requirements for videos:

Our goal to use such videos to train the model to detect the movements that can be considered as violence. The model must be trained to classify video according to violent movements only. Such elements like background, clothes, beating adult or beaten child, lighting etc. must not have influence on classification results.

So, we make videos under conditions:

- Different clothes of the humans and the dolls.
- Different background, interior.
- Different lighting.
- Different poses of the humans and the dolls.
- Different vertical angle.
- Most of videos are synthetical created with the dolls, but also some real videos from the Internet will be added to train/test datasets.
- We will use short videos (1-2 seconds, 30 fps) for training/testing.
- Videos must contain evident violent actions only.

Types of violent actions:

At this stage we defined three types of violent behavior which will be detected.

- Hitting a child by hands.
- Hitting a child by legs.

Extra types of violence can be added at late stages.

Non-violent actions:

Non-violent actions are the actions that do not contain the actions are defined as violent. It can be any actions, but it should use the actions that are similar to violent ones. For example:

- Dances
- Active games
- Sport

Data augmentation:

Because sometimes we do not have enough data, especially containing violence, it is possible to create a new data from the old one. It means we create new videos from the existing ones. We flip (mirror) the existing videos horizontally. See Figure 13.

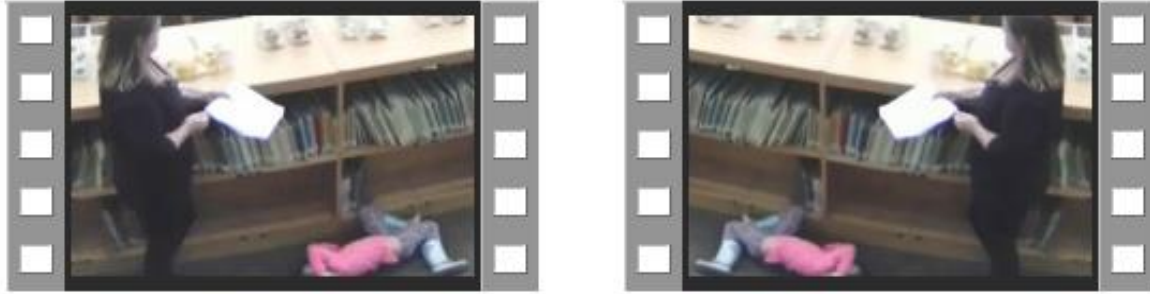


Figure 13. Example of mirrored videos

Data labeling

At this stage we do not perform labeling of the data intended for training/validation.

We sort the video files according to their content: violent or non-violent and put them into relevant directory. While loading they are labeled automatic according to the directory.

Labeling of the video files intended for testing will be performed at late stages.

Algorithms

In this section are described algorithms that can be used in our project.

Algorithms for violent behavior detection

- 1) Using a technology, for example ConvLSTM/CNN+LSTM, that can detect violence.

Advantages:

Only one technology is used.

Disadvantages:

In this case we find only frames containing violent behavior. If we need to find a region with humans we need to use other methods.

Conclusion:

The algorithm can be used for our goals.

- 2) Using a technology, OpenPose [9] or some other similar, that detect humans and their body key points. Following the body key points movements can help detect both violence and region containing this violence.

Advantages:

Only one technology is used.

Disadvantages:

- The technology is very slow (4-7 second/frame). Cannot be used for long videos.
- The second problem: a key point is not always detected, even when the relevant part of body is visible. It can cause an algorithm failure. Combination with other technologies does not solve this problem.

Conclusion:

The algorithm is rejected because of disadvantages.

- 3) There is a fact: Humans are not always found in a room in which a camera is installed.

So, we can use two-stages algorithm:

- Using a technology like YOLO [11] infrequently, 1-2 frame(s) per second(s), we can detect if humans are found on a video.
- If they are found, we apply the second technology like ConvLSTM/CNN+LSTM to detect violence.

Advantages:

This algorithm works faster when humans are not always present in the room.

Disadvantages:

If humans are always or most of time present in the room, it may work slower.

Conclusion:

The algorithm can be used for our goals.

We continue working with algorithms 1, 2. They must be tested for speed and accuracy.

While using a technology that detects humans and their location it is not obligatory to use every frame. We can use 1-2 frames per second for this purpose. In this way we could improve performance. It must be checked.

Also, while using the technology that detects violence, we should not use all of the frames. Maybe 10-15 frames per second, it must be checked and it can also improve performance.

We need to test each technology and define which of them is “heavier”. According to the results we should choose the algorithm in which a “heavier” technology is used less often and an “easier” technology is used more often. A “heavier” technology means that this technology takes longer time to process the same quantity of data.

Videos for training/testing will be short: 1-2 seconds, 30 frames per second.

Algorithm for real videos processing

For testing and real videos classification we will use long videos processing with windowing algorithm. The window length is 120-150 frames (4-5 seconds) and the step is 30-90 frames (1-3 seconds). These values are approximate and will be precise during of the development process.

Project code description

In this section is provided a description of the files (modules) containing the code implementing the project.

settings.py:

In this module are defined global parameters that are used in several modules.

For example: list of classification classes, height/width of the processed images etc.

main.py:

Main module of the project. In this module can be performed all the actions defined in the other modules.

utils.py:

This module contains some utilities. At this moment there is the function that sets CPU/GPU mode for the model training.

frames_processing.py:

Video and frames processing for further training, testing, classification.

- Getting video parameters.
- Loading frames from video files.
- Writing frames to video files.
- Writing frames to image files.
- Resizing, padding frames.
- Extracting certain frames.
- Extracting specific region from the frames.
- Creating blank frames.
- Creating padded frames sequence – adding blank frames to the existing ones.
- Flipping video files for data augmentation purpose.

classification.py:

Video stream classification:

- Frames sequence classification.
- Windowing.
- Models testing.

conv_lstm2d.py:

Creating/training a model based on the architecture “ConvLSTM2D”.

conv2d_lstm.py:

Creating/training a model based on combination of the architectures “Conv2D” and “LSTM”.

conv3d.py:

Creating/training a model based on the architecture “Conv3D”.

yolo.py:

Object detection using YOLO3 technology.

geometry.py:

Calculating distance between rectangles. It is used to calculate distance between regions containing human objects received from YOLO image processing.

Bibliography and References

23. Baba, M., Gui, V., Cernazanu, C. and Pescaru, D., 2019. A sensor network approach for violence detection in smart cities using deep learning. *Sensors*, 19(7), pp.1676.
24. Sumon, S.A., Goni, R., Hashem, N.B., Shahria, T. and Rahman, R.M., 2020. Violence Detection by Pretrained Modules with Different Deep Learning Approaches. *Vietnam Journal of Computer Science*, 7(01), pp.19-40.
25. Zhou, P., Ding, Q., Luo, H. and Hou, X., 2018. Violence detection in surveillance video using low-level features. *PLoS one*, 13(10), pp. e0203668.
26. Nievas, E.B., Suarez, O.D., García, G.B. and Sukthankar, R., 2011, August. Violence detection in video using computer vision techniques. In *International conference on Computer analysis of images and patterns*, Berlin and Heidelberg: Springer, pp. 332-339.
27. Cao, Z., Simon, T., Wei, S.E. and Sheikh, Y., 2017. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7291-7299.
28. YOLO: Real Time Object Detection:
<https://github.com/pjreddie/darknet/wiki/YOLO:-Real-Time-Object-Detection>
29. Soomro, K., Zamir, A.R. and Shah, M., 2012. A dataset of 101 human action classes from videos in the wild. *Center for Research in Computer Vision*, 2(11).
30. Reborn Dolls: <https://www.reborns.com/>

31. OpenPose: <https://github.com/spmallick/learnopencv/tree/master/OpenPose-Multi-Person>
32. Optical Flow: <https://github.com/chuanenlin/optical-flow>
33. Keras Yolo 3: <https://github.com/qgwweee/keras-yolo3>
34. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I., 2017. Attention is all you need. *Advances in neural information processing systems*, 30, pp.5998-6008.
35. BEHAVE database:
<http://groups.inf.ed.ac.uk/vision/BEHAVEDATA/INTERACTIONS/>
36. SIFT: <https://medium.com/@lerner98/implementing-sift-in-python-36c619df7945>
37. Laptev, I., 2005. On space-time interest points. *International journal of computer vision*, 64(2-3), pp.107-123.
38. Xu, L., Gong, C., Yang, J., Wu, Q. and Yao, L., 2014, May. Violent video detection based on MoSIFT feature and sparse coding. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3538-3542.
39. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S. and Uszkoreit, J., 2020. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv preprint arXiv:2010.11929*.
40. Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I. and Savarese, S., 2019. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 658-666.
41. Haaretz, 2020, "הגננת כרמל מעודה הורשעה בהתעללות ובתקיפת 11 פעוטות"
https://www.haaretz.co.il/news/law/.premium-1.9360248?utm_source=App_Share&utm_medium=Android_Native&utm_campaign=Share
42. Maariv, 2020, "אמא לילד מהגן בו התעללו בפעוטות: הייתה לי הרגשה, חיפשתי סימנים"
<https://www.maariv.co.il/news/law/Article-799453>

43. Ynet, 2020, "חשד לפרשת התעללות נוספת: סייעות בגן ילדים בחולון נעצרו"
<https://www.ynet.co.il/news/article/rkCI511KGv>
44. Mako, 2020, "חשיפה: התעללות קשה בפעוטות בגן ילדים ברמת גן"
https://www.mako.co.il/news-law/2020_q4/Article-852b6a71f058571027.htm
45. Hu, W. S., Li, H. C., Pan, L., Li, W., Tao, R., & Du, Q. (2019). Feature extraction and classification based on spatial-spectral ConvLstm neural network for hyperspectral images. *arXiv preprint arXiv:1905.03577*.
46. Hu, W. S., Li, H. C., Pan, L., Li, W., Tao, R., & Du, Q. (2020). Spatial-spectral feature extraction via deep ConvLSTM neural networks for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 58(6), 4237-4250.
47. Video Frame Prediction with Keras:
<https://srirangatarun.wordpress.com/2018/07/09/video-frame-prediction-with-keras/>
48. LSTMs for Human Activity Recognition Time Series Classification:
<https://machinelearningmastery.com/how-to-develop-rnn-models-for-human-activity-recognition-time-series-classification/>
49. Video Classification in Keras using ConvLSTM: <https://thebinarynotes.com/video-classification-keras-convlstm/>
50. UCF101 - Action Recognition Data Set:
<https://www.crcv.ucf.edu/data/UCF101.php>
51. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. (2020, August). End-to-end object detection with transformers. In *European Conference on Computer Vision* (pp. 213-229). Springer, Cham.
52. Wang, X., Girshick, R., Gupta, A., & He, K. (2018). Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7794-7803).
53. Parmar, N., Vaswani, A., Uszkoreit, J., Kaiser, L., Shazeer, N., Ku, A., & Tran, D. (2018, July). Image transformer. In *International Conference on Machine Learning* (pp. 4055-4064).

54. Li, Jason and Helen Qiu (2019). Comparing Attention-based Neural Architectures for Video Captioning.
55. Non-local Neural Networks for Video Classification:
<https://github.com/facebookresearch/video-nonlocal-net>
56. Transformer for Action Recognition in PyTorch: <https://github.com/Axe--/ActionBERT>
57. Vision Transformer – Pytorch: <https://github.com/lucidrains/vit-pytorch>
58. Dorrer, M. G., & Tolmacheva, A. E. (2020, November). Comparison of the YOLOv3 and Mask R-CNN architectures' efficiency in the smart refrigerator's computer vision. In *Journal of Physics: Conference Series* (Vol. 1679, No. 4, p. 042022).
59. A simple Conv3D example with TensorFlow 2 and Keras:
<https://www.machinecurve.com/index.php/2019/10/18/a-simple-conv3d-example-with-keras/>
60. 3D MNIST – A 3D version of MNIST database of handwritten digits:
<https://www.kaggle.com/daavoo/3d-mnist>
61. Hou, R., Chen, C., & Shah, M. (2017). An end-to-end 3d convolutional neural network for action detection and segmentation in videos. *arXiv preprint arXiv:1712.01111*.
62. Hurtik, P., Molek, V., Hula, J., Vajgl, M., Vlasanek, P., & Nejezchleba, T. (2020). Poly-YOLO: higher speed, more precise detection and instance segmentation for YOLOv3. *arXiv preprint arXiv:2005.13243*.
63. Buric, M., Pobar, M., & Ivasic-Kos, M. (2018, December). Ball detection using YOLO and Mask R-CNN. In *2018 International Conference on Computational Science and Computational Intelligence (CSCI)* (pp. 319-323). IEEE.
64. ConvLSTM2D: https://keras.io/api/layers/recurrent_layers/conv_lstm2d/
65. Mask R-CNN:
https://github.com/matterport/Mask_RCNN/releases/download/v2.0/mask_rcnn_coco.h5
66. Conv3D: https://keras.io/api/layers/convolution_layers/convolution3d/

FINAL PROJECT PROGRESS

Detection of violence against children in videos

Student Name: Rakhlevski Ilia

Supervisor: Dr. Oren Dinai

Advisor: Dr. Yehudit Aperstein

Changes in the project proposal

2

The output of the application will be video stream compiled from the frames with violence detected in the input video streams instead of frames written to image files (this option still exists).

Iliia Rakhlevski

Technologies investigation

3

In this section are described technologies investigation results.

We want to find out which technologies, methods, libraries can be useful for our goals.

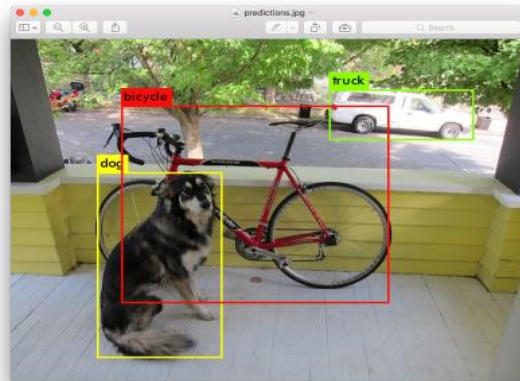
Iliia Rakhlevski

Technologies investigation (cont.)

4

YOLO

Object detection library. YOLO object detector is used to detect objects in both images and video streams



Iliia Rakhlevski

Technologies investigation (cont.)

5

YOLO

Advantages:

Can detect human objects on a stream, their sizes (in pixels) and locations (in pixels) on the image.

Disadvantages:

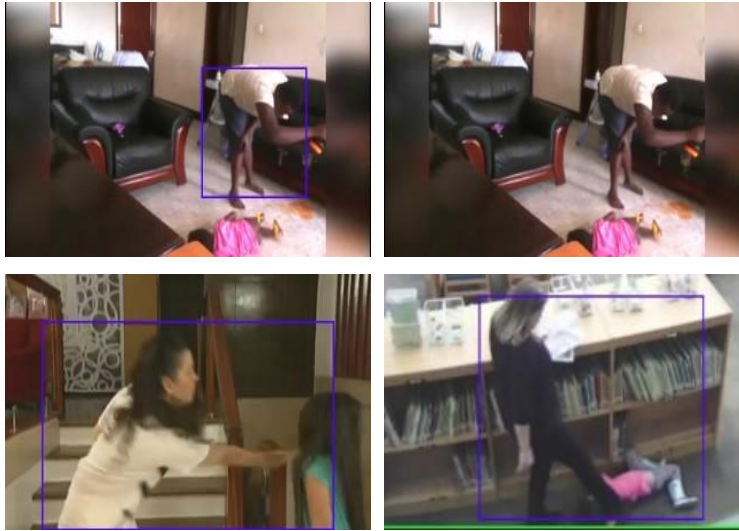
- Sometimes a human is incorrect detected or not detected at all.
- If two or more humans are very close, they can be detected as one human

Iliia Rakhlevski

Technologies investigation (cont.)

6

YOLO



Iliia Rakhlevski

Technologies investigation (cont.)

7

YOLO

Conclusion:

This technology can be used to detect human objects and the region in which can occur actions between humans. For our goal it can be used in combination with other technology only.

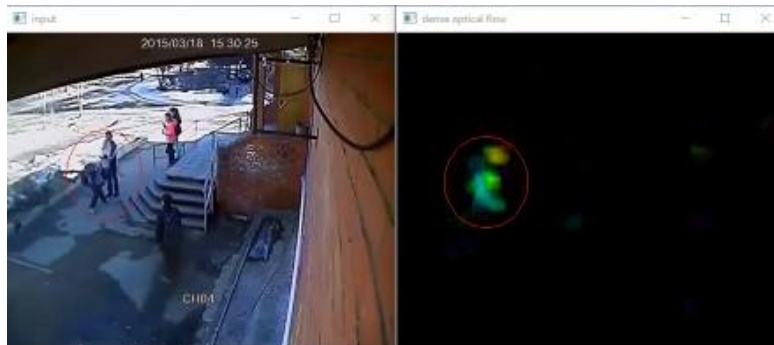
Iliia Rakhlevski

Technologies investigation (cont.)

8

Optical flow

Optical flow is defined as the apparent motion of individual pixels on the image plane.



Iliia Rakhlevski

Technologies investigation (cont.)

9

Optical flow

Advantages:

Detects objects motion and its speed.

Disadvantages:

- Camera must be absolutely stationary.
- Detects all moving objects, not only humans.
- It does not detect an object if it does not move. Example: beaten child.
- Detects noise. Example: a camera is not qualitative; it can have a lot of digital noise.

Iliia Rakhlevski

Technologies investigation (cont.)

10

Optical flow

Conclusion:

It can be used to detect regions where are performed fast motions that can be considered as hits. For our goal it can be used in combination with other technologies only.

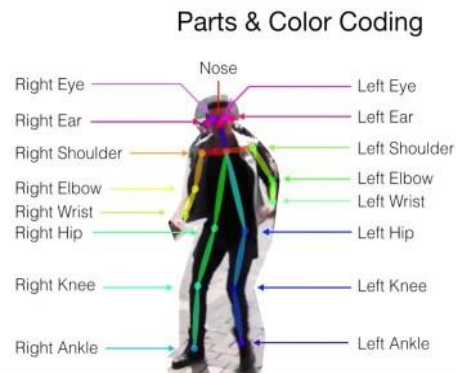
Iliia Rakhlevski

Technologies investigation (cont.)

11

OpenPose

OpenPose has represented the first real-time multi-person system to jointly detect human body, hand, facial, and foot key points on single images.



Iliia Rakhlevski

Technologies investigation (cont.)

12

OpenPose

Advantages:

Detects human objects, legs and hands, also their sizes and locations.

Hits that performed as a rule by legs and hands can be easily detected and analyzed.



Iliia Rakhlevski

Technologies investigation (cont.)

13

OpenPose

Disadvantages:

- Works very slow. Each frame is processed 47 seconds.
- Human body is not always detected.
- A key point is not always detected (even if it is visible).

Ilia Rakhlevski

Technologies investigation (cont.)

14

OpenPose

Example: Left hand is visible, but not detected (right image). Right hand is visible, but not detected (left image)



Ilya Rakhlevski

Technologies investigation (cont.)

15

OpenPose

Conclusion:

Because of time of processing can be used for short videos. It can be used to detect human objects and the region in which an action between humans occurs. It can help to detect movement of hands and legs that can be considered as hits.

Iliia Rakhlevski

Technologies investigation (cont.)

16

ConvLSTM(ConvLSTM2D)

Convolutional LSTM. It is similar to an LSTM layer, but the input transformations and recurrent transformations are both convolutional. It can be used for:

- Feature extraction and classification.
- Video frame prediction.
- Human activity recognition.

Iliia Rakhlevski

Technologies investigation (cont.)

17

ConvLSTM(ConvLSTM2D)

Experiment 1:

Two classes are selected for classification: biking and horse riding.

The video format of the data - 320X240, 30 fps.

While video processing the frames were compressed to 120X90.

The model was trained on 189 videos and validated on 48 videos with 40 epochs.



Iliia Rakhlevski

Technologies investigation (cont.)

18

ConvLSTM(ConvLSTM2D)

Train/test results:

Train on 189 samples, validate on 48 samples

	precision	recall	f1-score	support
0	0.90	0.83	0.86	23
1	0.90	0.95	0.92	37
accuracy	0.90	0.90	0.90	60
macro avg	0.90	0.89	0.89	60
weighted avg	0.90	0.90	0.90	60

Iliia Rakhlevski

Technologies investigation (cont.)

19

ConvLSTM(ConvLSTM2D)

Experiment 2:

For this experiment we used dataset of 340 videos with/without violence:

275 – train, 31 – validation, 34 – test.



Iliia Rakhlevski

Technologies investigation (cont.)

20

ConvLSTM(ConvLSTM2D)

Train/test results:

Train on 275 samples, validate on 31 samples

	precision	recall	f1-score	support
0	0.82	0.60	0.69	15
1	0.74	0.89	0.81	19
accuracy	0.76	34		
macro avg	0.78	0.75	0.75	34
weighted avg	0.77	0.76	0.76	34

Iliia Rakhlevski

Technologies investigation (cont.)

21

ConvLSTM (ConvLSTM2D)

Advantages:

It can be used for violence detection without using of some extra technologies.

Disadvantages:

It detects frames containing violence only. In order to localize the regions (optional) with violence extra technologies or algorithms must be used.

Conclusion:

This architecture can be used for violence detection without using of any extra technologies. For localization of region containing violence must be used other methods/algorithms.

Ilia Rakhlevski

Technologies investigation (cont.)

22

Transformers (Self-Attention)

This technology can be used for:

- Object detection.
- Video classification.
- Image classification.
- Image generation.

Advantages:

It can be used for violence detection without using of some extra technologies.

Iliia Rakhlevski

Technologies investigation (cont.)

23

Transformers (Self Attention)

Disadvantages:

It detects frames containing violence only. In order to localize the regions with violence extra technologies or algorithms must be used.

Conclusion:

This architecture can be used for violence detection without using of any extra technologies. For localization of region containing violence must be used other methods/algorithms.

Iliia Rakhlevski

Technologies investigation (cont.)

24

Mask R-CNN

It is object detector is used to detect objects in both images and video streams.



Iliia Rakhlevski

Technologies investigation (cont.)

25

Mask R-CNN

Advantages:

Can detect human objects on a stream, their sizes (in pixels) and locations (in pixels) on the image.

Disadvantages:

Sometimes a human is incorrect detected, not detected at all or two or more humans are detected as one.

Conclusion:

This technology can be used to detect human objects and the region in which can occur actions between humans. For our goal it can be used in combination with other technology only.

Iliia Rakhlevski

Technologies investigation (cont.)

26

Conv3D

This architecture can be used for video processing.

Experiment 1:

We have used the code and the data from the example. The author developed an application that recognized 3D digits from the 3D MNIST. This is a 3D version of MNIST database of handwritten digits.



Iliia Rakhlevski

Technologies investigation (cont.)

27

Conv3D

Accuracy of the original code is 67%. After of changes of some parameters we have succeeded to reach accuracy up to 76%.

Train/test results:

Train on 8000 samples, validate on 2000 samples

Test loss: 0.9244766535758973 / Test accuracy: 0.7689999938011169

Iliia Rakhlevski

Technologies investigation (cont.)

28

Conv3D

Experiment 2:

For this experiment we used the same dataset that we used for ConvLSTM

Experiment 2. During the experiment were changed several parameters:

- Number videos in the dataset.
- Number and size of filters.
- Number of layers.

Train on 76 samples, validate on 20 samples

Test loss: 0.7750303149223328 / Test accuracy: 0.5

Iliia Rakhlevski

Technologies investigation (cont.)

29

Conv3D

Advantages:

It can be used for violence detection without using of some extra technologies.

Disadvantages:

It detects frames containing violence only. In order to localize the regions with violence extra technologies or algorithms must be used.

Conclusion:

The accuracy received in the experiments is very low.

Iliia Rakhlevski

Technologies investigation (cont.)

30

CNN+LSTM (Conv2D+LSTM)

This is a combination of two architectures CNN and LSTM. First the input data is sent to convolution neural network and its output is sent as input to LSTM network.

Experiment:

For this experiment we used the same dataset that we used ~~Conv~~LSTM Experiment 2.

Train/test report:

Train on 380 samples, validate on 43 samples

Test loss: 2.0506185775107526 / Test accuracy: 0.8085106611251831

	precision	recall	f1-score	support
0	0.73	0.57	0.64	14
1	0.83	0.91	0.87	33
accuracy	0.81	47		
macro avg	0.78	0.74	0.75	47
weighted avg	0.80	0.81	0.80	47

Iliia Rakhlevski

Technologies investigation (cont.)

31

CNN+LSTM (Conv2D+LSTM)

Advantages:

It can be used for violence detection without using of some extra technologies.

Disadvantages:

It detects frames containing violence only. In order to localize the regions with violence extra technologies or algorithms must be used.

Conclusion:

This architecture can be used for violence detection without using of any extra technologies. For localization of region containing violence must be used other methods/algorithms.

Iliia Rakhlevski

Technologies investigation (cont.)

32

Investigation Conclusion

We need two types of technologies:

- Detection humans on a frame, their locations and sizes.
- Detection violence in sequence of frames.

Chosen technologies:

- Detection humans on a frame– YOLO, Mask R-CNN.
- Detection violence in sequence of frames– ConvLSTM, CNN+LSTM.

Iliia Rakhlevski

Data

33

- Data is used in our project is video streams containing violent /non-violent scenes.
- The videos are found in the Internet or synthetically created by using reborn baby dolls.



Iliia Rakhlevski

Data (cont.)

34

Requirements for videos

The model must be trained to classify video according to violent movements only.

Such elements like: background, clothes, beating adult or beaten child, lighting etc. must not have influence on classification results.

Types of violent actions:

- Hitting a child by hands.
- Hitting a child by legs.

Non-violent actions

- Dances
- Active games
- Sport

Iliia Rakhlevski

Data (cont.)

35

Data augmentation:

- Because sometimes we do not have enough data, especially containing violence, it is possible to create a new data from the old one.
- We flip (mirror) the existing videos horizontally



Iliia Rakhlevski

Data (cont.)

36

Data labeling

At this stage we do not perform labeling of the data intended for training/validation.

We sort the video files according to their content: violent or nonviolent and put them into relevant directory. While loading they are labeled automatic according to the directory.

Labeling of the video files intended for testing will be performed at late stages.

Iliia Rakhlevski

Algorithms

37

Algorithms for violent behavior detection:

1st algorithm

Using a technology, for example ConvLSTM/CNN+LSTM, that can detect violence.

Advantages:

Only one technology is used.

Disadvantages:

In this case we find only frames containing violent behavior. If we need to find a region with humans we need to use other methods.

Conclusion:

The algorithm can be used for our goals.

Iliia Rakhlevski

Algorithms

38

2nd algorithm

Using a technology, OpenPose or some other similar, that detect humans and their body key points. Following the body key points movements can help to detect both violence and region containing this violence.

Advantages:

Only one technology is used.

Disadvantages:

- The technology is very slow (4-7 second/frame). Cannot be used for long videos.
- The second problem: a key point is not always detected, even the relevant part of body is visible. It can cause the algorithm failure. Combination with other technologies does not solve this problem.

Conclusion:

The algorithm is rejected because of disadvantages.

Iliia Rakhlevski

Algorithms

39

3rd algorithm

There is a fact: Humans are not always found in a room in which a camera is installed. So, we can use two-stages algorithm:

- Using a technology like YOLO infrequently, 1-2 frame(s) per second(s), we can detect if humans are found on a video.
- If they are found then we apply the second technology like ConvLSTM/CNN+LSTM to detect violence.

Advantages:

This algorithm work faster when humans are not always present in the room.

Disadvantages:

If humans are always or most of time present in the room then it can work slower.

Conclusion:

The algorithm can be used for our goals.

Ilia Rakhlevski

Algorithms

40

We continue working with algorithms 1, 3. They must be tested for speed and accuracy.

While using the technology that detects violence, we should not use all the frames. Maybe 10-15 frames per second.

While using a technology that detects humans we can use 1-2 frames per second for this purpose.

Videos for training/testing will be short: 1-2 seconds, 30 frames per second.

Iliia Rakhlevski

Algorithms

41

Algorithm for real videos processing

- For testing and real videos classification we will use long videos processing with windowing algorithm.
- The window length is 120-150 frames (4-5 seconds) and the step is 30-90 frames (1-3 seconds).
- These values are approximate and will be precise during of the development process.

Iliia Rakhlevski

FINAL PROJECT

Detection of violence against children in videos

Student Name: Rakhlevski Ilia

Supervisor: Dr. Oren Dinai

Advisor: Dr. Yehudit Aperia

Motivation

2

- Violence against children is one of the biggest problems affecting families and societies. There many publications in mass media about this problem.



- According to the World Health Organization estimation up to 1 billion children aged 2 – 17 years, have experienced physical, sexual, or emotional violence or neglect in the past year.

Iliia Rakhlevski

Motivation (cont.)

3

- When violence occurs in a Kindergarten it often remains unknown to other people.
- Our goal is development of an application that will help to detect frames of violence in video streams.
- After the violence detection, an alert or a summary could be extracted to adjacent security department (or to the parents) to yield an action.

Iliia Rakhlevski

Research problem

4

- The input of the application will be video stream - sequence of frames, that contains any interaction between an adult and children in Kindergarten.
- The output of the application will be localization of the detected frames with violence in the video streams.
- Training and testing data will be video streams taken from the Internet, movies or semi-synthetic created videos. Some parts of them contain scenes with violence and the other parts contain non-violent ones.
- The quality of the solution is measured by quantity of correct detected violent frames.

Iliia Rakhlevski

Prior work

5

- All the projects/articles that we have found are related to common violence and are not focused on a specific type of violence of adults against children.
- In these projects/articles are used both classical methods (for example: SVM, CNN, LSTM) and modern ones (for examples: VGG16, ResNet50).
- They reached various classification accuracies between 87 and 100%. Such high accuracies could be reached as a result of using very small datasets (not enough frames containing the scenes with violence) .

Iliia Rakhlevski

Project novelty

6

- In this project we are focused on specific type of the violence of adults against children.
- There are no available data sets match this problem.
- It is very difficult to obtain a real data for privacy and legal reasons.

Iliia Rakhlevski

Data description

7

- Format of video streams used in the project is MP4.
- While a video stream processing the frames are grabbed from the video. It is created an array of the frames. This array serves as an input of the model.
- After converting of the video stream into array of the frames we change the frames size according to the model input.

Iliia Rakhlevski

Data description (cont.)

8

- Video streams contain violent/non-violent scenes.
- At this stage we defined two types of violent behavior which will be detected - hitting a child by hand and by leg .
- Non-violent actions are the actions that do not contain the actions that are defined as violent.

Iliia Rakhlevski

Data description (cont.)

9

Violent scenes



Iliia Rakhlevski

Data description (cont.)

10

Non-violent scenes



Iliia Rakhlevski

Data description (cont.)

11

- Because of lack of videos containing scenes of violence of adults against children we decided to create synthetical videos.
- Using reborn baby dolls – the dolls that are very similar to human infants.



Iliia Rakhlevski

Data description (cont.)

12

- Our goal to use such videos to train the model to detect the movements that can be considered as violence.
- The model must be trained to classify video according to violent movements only.
- Elements like background, clothes, beating adult or beaten child, lighting etc. must not have influence on classification results.
- During each video session must be made videos of both types: with violence and no violence.
- Using data augmentation.

Iliia Rakhlevski

Data description (cont.)

13

Labeling

- Training:

We sort the video files according to their content: violent or non-violent and put them into the relevant directory.

- Testing:

For each video is created a file containing ranges of the “violent” frames.

Iliia Rakhlevski

Data description (cont.)

14

Training data creation

For each video session we perform several actions:

- Camera must be stable during the whole video session.
- Only humans/dolls movements can change.
- Each video session must contain both types of actions: with/no violence.
- Each video stream created during the video session is divided into small videos of size 20 frames.
- All the small videos are sorted according to their content (violent/non -violent).

Iliia Rakhlevski

Methodology

15

Architecture

- We chose the CNN+LSTM. This is a combination of two architectures CNN and LSTM.
- The input of the network is a video, sequence of 2D images.
- The output is predicted class.
- CNN is used for feature extraction and LSTM is used to classify video based on those features.

Iliia Rakhlevski

Methodology (cont.)

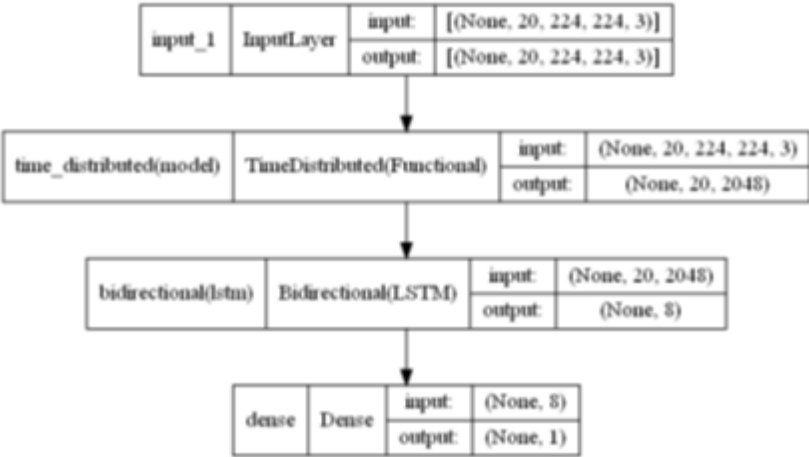
17

Architecture Implementation

- Using Keras library.
- Using a pre-trained CNN model ResNet152V2.
- Input - short video stream 20 frames. Each frame is 224x224x3 and pixels values are scaled between -1 and 1.
- Output – predicted class. 0 – violent, 1 – non-violent.
- Using Bidirectional LSTM. It can improve accuracy when the dataset has imbalanced classes.

Iliia Rakhlevski

Methodology (cont.)



Iliia Rakhlevski

Methodology (cont.)

19

Transfer learning

Using the ResNet152V2 model.

- Create the ResNet152V2 model without an output layer.
- Load the weights for the pre-trained model (without output layer).
- Freeze all the layers of the model that they will not be trained during the whole model training.
- As the output layer is added GlobalAveragePooling2D layer.

Iliia Rakhlevski

Methodology (cont.)

20

Overfitting prevention

- To simplify the architecture - reducing number of layers or/and neurons.
- Increasing number of samples - creating extra videos.
- Data augmentation.
- Using Regularization: Elastic Net (combination L1 and L2).

Iliia Rakhlevski

Methodology (cont.)

21

Hyperparameters

Hyperparameters are set experimentally during model training.

- Number of layers and neurons . This parameter is set for the bidirectional LSTM layer – 4 units.
- Learning rate. It is set to 0.0001 at the start of training. At the end stages of training it can be changed to 0.00001.
- Adam optimizer was used to train the network.
- The batch size is 64.
- Class weight parameter. It is calculated (for class j):
$$w_j = n_samples / (n_classes * n_samples_j)$$

Iliia Rakhlevski

Methodology (cont.)

22

Model training process

1. We train the created/trained model.
2. During the model training we follow the parameters both “loss” and “accuracy”.
3. If the model stopped improving, then the learning rate should be updated and the training should continue.
4. If the model stopped improving finally, we stop the training.
5. We change the model: layers, number of neurons, methods of regularization and repeat the steps 1-4.

Iliia Rakhlevski

Methodology (cont.)

23

Testing

- Testing of the model is performed on real videos.
- Using the windowing algorithm for videos processing.
- Classification accuracy measurement:
 - IoU (Intersection over Unit) is used for videos containing both “violent”/”non - violent” frames.
 - Also, we use statistics on correct/incorrect predicted “violent” / ”non -violent” frames.
- For each video is created a file containing ranges of the “violent” frames.

Iliia Rakhlevski

Methodology (cont.)

24

For the testing purpose we used videos of several types :

- Synthetical videos – they were using reborn dolls containing scenes with violence and without one.
- Real videos with violence – real videos containing scenes of violence .
- Real videos without violence - real videos that do not contain scenes of violence.
- Videos that do not contain humans - videos that contain rooms with furniture only.

Total: 120 video files, 45850 – frames, ~25.5 minutes duration.

Ilia Rakhlevski

Results

25

Summary – all types	<p>With Violence (actual): 3230 Correct predicted: 2788 (86.32 %) Incorrect predicted: 442 (13.68 %)</p> <p>No Violence (actual): 42620 Correct predicted: 38852 (91.16 %) Incorrect predicted: 3768 (8.84 %)</p>
---------------------	---

Iliia Rakhlevski

Results (cont.)

26

Synthetical videos containing scenes with and without violence	<p>With Violence (actual): 1638 Correct predicted: 1398 (85.35 %) Incorrect predicted: 240 (14.65 %)</p> <p>No Violence (actual): 9914 Correct predicted: 8855 (89.32 %) Incorrect predicted: 1059 (10.68 %)</p>
Real videos with violence	<p>With Violence (actual): 1592 Correct predicted: 1390 (87.31 %) Incorrect predicted: 202 (12.69 %)</p> <p>No Violence (actual): 2045 Correct predicted: 693 (33.89 %) Incorrect predicted: 1352 (66.11 %)</p>

Iliia Rakhlevski

Results (cont.)

27

Real videos without violence	<p>With Violence (actual): 0 Correct predicted: 0 (100 %) Incorrect predicted: 0 (0 %)</p> <p>No Violence (actual): 30477 Correct predicted: 29120 (95.55 %) Incorrect predicted: 1357 (4.45 %)</p>
Videos that do not contain humans	<p>With Violence (actual): 0 Correct predicted: 0 (100 %) Incorrect predicted: 0 (0 %)</p> <p>No Violence (actual): 184 Correct predicted: 184 (100.0 %) Incorrect predicted: 0 (0.0 %)</p>

Iliia Rakhlevski

Results (cont.)

28

- 86 % “violent” and 91 % “non -violent” frames are correct predicted.
- Synthetical videos: 85 % “violent” and 89 % “non -violent” frames are correctly predicted.
- Real videos with violence: 87 % “violent” and 33 % “non -violent” frames are correctly predicted.
- Real videos without violence: 95 % “non-violent” frames are correctly predicted.
- Videos with empty rooms: 100 % “non -violent” frames are correctly predicted.

Iliia Rakhlevski

Results (cont.)

29

- For “non-violent” frames the results are better.
- The “non-violent” frames are found near the “violent” ones and they are predicted as “violent” too.
- The best results for “violent” frames recognition were received for those videos where were used sequences of hits: several hits, quickly following one another.
- Results of the testing are relative good, in the main for those videos that are similar to the videos from the dataset.

Iliia Rakhlevski

Discussion

30

Differences between our project and the existing ones

Other projects:

- In most cases it is enough to use one frame to perform recognition.
- In all cases it talks about certain actions. Each action is one class.

Our projects:

- We must recognize action. We need all the frames from a video for this purpose.
- We have two classes only, but the “violence” class can have many actions.
The “non-violence” class has an endless number of actions.

Iliia Rakhlevski

Discussion

31

Our goals

- Our main task is that the model will learn from the human movement only. To meet this target, during each video session we created videos containing both violent and non-violent movements.
- Another important goal is to achieve generalization. The trained model must learn to recognize humans (reborn dolls) on the video and to classify their movement under different conditions: poses, clothes, gender, hairstyle etc.

Iliia Rakhlevski

Discussion

32

Further work

To continue this work, we need to increase the dataset.

- To add new real videos containing scenes of violence against children.
- Using more different reborn dolls and humans, clothes, backgrounds etc. to create scenes with many humans and reborn dolls.
- Using extra augmentation techniques: saturation, changing brightness etc.

Iliia Rakhlevski

Project files description

In this section is provided a description of the files (modules) containing the code implementing the project.

settings.py

In this module are defined global parameters that are used in several modules.

List of classification classes, height/width of the processed images etc.

utils.py

This module contains some utilities. The function that sets CPU/GPU mode for the model training and the function that returns date/time string.

frames_processing.py

Implementation of video and frames processing for further training, testing, classification. Getting video parameters, loading frames from video files, writing frames to video files, writing frames to image files, resizing, padding frames, extracting certain frames, extracting specific region from the frames, creating blank frames, creating padded frames sequence – adding blank frames to the existing ones, flipping, blurring, reversing, rotating video files for data augmentation purpose.

classification.py

Implementation of video stream classification.

Frames sequence classification, windowing, models testing.

conv2d_lstm.py

Creating/training a model based on combination of the architectures “Conv2D” and “LSTM”.

Training_run_results.txt

Summary of the model training.

Testing_run_results.txt

Output of the reference videos testing.

conv2d_lstm_model.h5

The file contains the trained model.

Development tools that were used for the project implementation.

Language: Python 3.9

Libraries: keras/tensorflow, pandas, matplotlib, scikit-learn, numpy, opencv-python.

Development environments: Spider 3.3.6, PyCharm 2021.3.1.

Cloud platforms: Google Colaboratory, Kaggle.

The code for training is found in the file *conv2d_lstm.py*. The train dataset must be located in the same directory as this file. The dataset directory is called "video_data" and must be found in the same directory as the code files. Open the development environment and run this file.

The code for classification is found in the file *classification.py*. Be sure that the model file *conv2d_lstm_model.h5* is located in the same directory. In the function *testing* set the correct path to the directory containing the tested video files. In the “main” function set the lines

```
lst = [i for i in range(1, 120 + 1)]  
testing(model, lst, False)
```

uncommented. Open the development environment and run this file.

Poster

Ilya Rokhlevski
Advisor: Dr. Oren Dinal
Consultant: Dr. Yehudit Aperstein

Intelligent Systems

AFEKA TEL-AVIV ACADEMIC COLLEGE OF ENGINEERING

Detection of violence against children in videos

In this project we propose a method for detecting violence against children, using video streams captured by cameras. We built a model based on Convolutional Neural Network and Long Short-Term Memory for video classification.

Violence against children is a world spread phenomenon - it occurs around the world, in every country and every society; all too often it happens within the family and in many different forms such as neglect, physical and sexual abuse etc. Violence against children can have dire consequences for the child's physical, emotional and psychosocial development. In this paper we focused on physical violence in kindergarten and its detection. When violence occurs in a Kindergarten it often remains unknown since the child either afraid to tell his parents about it or simply cannot speak yet due to his/her age.

This project proposes a physical violence detecting method based on indoor surveillance cameras. The cameras capture video streams, these streams are classified by using a deep-learning Convolution Neural Network (CNN) and Long Short-Term Memory (LSTM) based approach for violence detection by learning the detailed features in the videos. CNN used for feature extraction while LSTM is used to classify the video, based on those features. As a CNN feature extraction we use ResNet152V2 pre-trained model.

The novelty of this project is synthesized data. In this project we focused on specific type of violence of adults against children. There are no available data sets match this situation and not enough relevant videos on the Internet, and the existing ones are very low quality. Thus, we decided to use reborn dolls which are very similar to real children.



For the testing purpose we used videos which created under similar conditions as the videos from the training dataset. We have reached various classification accuracies up to 70% for "violent" frames of the tested videos. The results of the tests were relatively good, but only for those videos similar to the ones from the dataset. We need to improve the generalization and for this purpose, we have to increase the training dataset.

Our main task is that the model will learn from the human movement only. To meet this target during each video session, we created videos containing both violent and non-violent movements. The rest of the parameters (background, clothes, distance etc.) remained the same.

Another important goal is to achieve generalization. The trained model must learn to recognize humans (reborn dolls) on the video and classify their movement under different conditions: poses, clothes, gender, hairstyle etc.

After numerous experiments we came to the conclusion that in order to train the model so it can generalize well - there is a need for thousands of videos. The dataset must contain not only synthetically generated videos but also real ones. To continue this work, we need to increase the dataset.

תקציר

אלימות נגד ילדים היא תופעה עולמית. יש צורות שונות של אלימות כאלו: הזנחה, התעללות פיזית, מינית. לאלימות נגד ילדים יש הרבה השלכות שליליות על התפתחות פיזית, רגשית ופסיכו-סוציאלית. בפרויקט זה אנחנו מתמקדים באלימות הפיזית ובזיהויה. פרויקט זה מציע שיטת זיהוי אלימות פיזית המבוססת על מעקב בתוך מצלמות. הן מצלמות זרמי וידאו, זרמים אלה מסווגים על ידי שימוש בלמידה עמוקה מבוססת Convolution Neural Network (CNN) ו Long Short-Term Memory (LSTM) גישה לזיהוי אלימות על ידי לימוד התכונות המפורטות בסרטונים. CNN משמש עבור חילוף תכונות ו-LSTM משמש לסיווג וידאו על סמך תכונות אלו. בתור תכונה של CNN חילוף אנו משתמשים במודל מיומן מראש של ResNet152V2. החידוש בפרויקט זה הוא נתונים מסונתזים. בפרויקט זה אנו מתמקדים בסוג ספציפי של אלימות של מבוגרים כלפי ילדים. אין מערכי הנתונים זמינות למצב זה ואין מספיק סרטונים רלוונטיים באינטרנט ורוב הסרטונים האלה נמצאים ברמת איכות נמוכה. אז, אנחנו הולכים להשתמש בבובות (reborn dolls) הדומות מאוד לילדים אמיתיים. הגענו לדיוקי זיהוי שונים של עד 86% עבור הפריימים "אלימות" של סרטונים שנבדקו.



אפקה המכללה האקדמית להנדסה בתל-אביב
TEL-AVIV ACADEMIC COLLEGE OF ENGINEERING

בית הספר להנדסת תוכנה

החוג למערכות תבניות

שם העבודה:

זיהוי אלימות כלפי ילדים בסרטונים

חיבור על עבודת גמר למילוי חלקי של הדרישות לקבלת

תואר M.Sc. במערכות תבניות

שם הסטודנט: איליה רחלבסקי

שם המנחה: ד"ר אורן דינאי

יועץ מומחה: ד"ר יהודית אפרשטיין

תאריך הגשה: 30.06.2022