Visual Classification on HR-Crime dataset

DAVID D. W. ELSKAMP, University of Twente, The Netherlands

Being able to detect anomalies in surveillance camera footage is essential for saving time on the otherwise time-consuming process which requires manual human detection. Recently researchers have made public the HR-Crime dataset which is a subset of a larger UCF-Crime dataset. The HR-Crime subset is available for automatic visual analysis of anomalies and consists of human-related crime scenes. In this paper, we will use this subset to detect human-related anomalies. We will be building a feature extraction pipeline using the latest technologies. And we will be presenting an implementation using two visual-based approaches for detecting anomalies in surveillance footage. One makes predictions one the whole scene whereas the other will make predictions based on human proposals in a scene. The results of our approaches were compared to the previously published skeleton extraction-based approach. Our approach turned out to be useful but not as good as previous techniques. There could still be improvements made to better the results using other techniques and improving the HR-Crime dataset. Lastly, the extracted features will be made publicly available and an appendix will give further insight into our model's prediction process.

Additional Key Words and Phrases: Surveillance videos, human-related anomaly detection, human-related crime detection

1 INTRODUCTION

Law enforcement agencies are always on the lookout for new technologies to aid their public safety efforts. The use of public safety cameras is one of the technologies that has been widely implemented throughout the last years. The idea behind using surveillance camera systems is that criminals will more likely refrain from participating in criminal activity when they know they are being watched [5].

In a lot of instances, security footage gets analyzed for solving crimes or violations and to later be used in court where lawyers and advocates use it as a strong piece of evidence [7]. The problem with this captured data is however that it is seldom looked at and gets saved on a server somewhere or is lost. This is the result of the tedious process of finding anomalies/crimes in surveillance videos, especially when a person manually does it. In addition, it is increasingly time-consuming and near impossible when multiple cameras capture footage 24/7. To be able to locate criminal activity in past footage or live footage is therefore a task hard to complete by humans. What if we could develop an artificial intelligence to automate this tedious process and be able to detect and categorise anomalies from surveillance camera footage through a machine learning model? The aim of this research is to attempt to develop two machine learning models that can automate this process using a visual-based approach. One model will be aimed at using bounding boxes around human subjects for classification whereas the other model will be trained on full frames. We aim at answering the following questions:

- **Research question 1:** *Can we use the visual information from the entire frames to accurately classify human-related anomalies?*
- **Research question 2:** *Can we use visual information by using bounding boxes around human subjects to accurately classify human-related anomalies?*
- **Research question 3:** *Are the results from using skeletal trajectories [1] a better option compared to using visual information or bounding boxes?*

The remaining parts of the paper are organised as follows: Section 2 describes related work and previous research. In Section 3 we will describe our methodology, dataset and feature extraction pipeline. In Section 4 we present a dataset analysis, our validation metrics and the results of our experiments. We elaborate and discuss these results in Section 5 and we come to our conclusions in Section 6. Lastly, we propose possible improvements for future work in Section 7.

2 RELATED WORK

2.1 Previous dataset

Previous research has already been conducted using an existing crime dataset to try and tackle this challenge. Researchers [1] used a publicly available UCF-Crime dataset [10] which consists of 950 crime-related videos and 950 normal videos. The crime/anomaly-related videos are categorised into 13 distinct categories. The categories are defined as:

Abuse, Arrest, Arson, Assault, Burglary, Explosion, Fighting, Road Accidents, Robbery, Shooting, Shoplifting, Stealing and Vandalism.

The researchers built a subset of the UCF-Crime dataset consisting of only human-related crime (HR-Crime) [1]. The HR-Crime dataset is a filtered version of the UCF-Crime dataset to be used for detecting human-related anomalies which we are interested in for this research. We will be using this HR-Crime dataset and it will be explained in more detail in Section 3.1.

2.2 Previous approach

This subset has so far been studied from a skeleton perspective, the pipeline they used can be seen in Figure 1. This pipeline shows how

Fig. 1. Feature extraction pipeline of HR-Crime dataset [1]. Given the frames of a robbery video, first human proposals from seperate frames were extracted. Second, body skeletons were detected. Finally, the skeletons accross multiple frames were tracked.

TScIT 37, July 8, 2022, Enschede, The Netherlands

the researchers extracted skeletons from the humans detected in a scene. It was based on their newly created HR-Crime dataset and the process was as follows: The researchers [1] first used YOLOv3 [3] to detect bounding boxes around all human subjects in a single frame, then extracted human body proposals out of those bounding boxes. These are also referred to as 'skeletons' and were then tracked over multiple frames. These results were used to train a machine learning model. The model was used to make predictions based on, by the model previously unseen surveillance footage, to determine whether an anomaly was detected and also which category of anomaly was being detected.

2.3 Previous results

The results from the trained model gave AUROC values varying between 0.43 and 0.73 [1] for different categories. For those works, the videos were split into train and test sets so the models could be trained with a subset and tested with another. In this work, we will work similarly to how Boekhoudt et al. [1] performed their research. This consists of using the same HR-Crime dataset for training our model, using the exact same train-test split to train and evaluate our model and using parts of the same methods they used like the bounding boxes. Contrary to their research we will however not focus on skeletons but use the bounding boxes provided by them. Besides that we will also experiment with a second method by using entire frames as a data source, more on this will be explained in Section 3.1.

3 METHODOLOGY

3.1 Dataset

The HR-Crime dataset has been made public by the researchers [1] so we can use it in our research. It uses the same 13 categories as the UCF-Crime dataset and consists of 789 human-related crime videos and 782 normal videos. It also consists of 239 testing videos with annotations [1] which we can use to evaluate our predictions later. Figure 3 shows examples of still frames in this dataset for the 13 different categories and gives a clear picture of what this dataset consists of.

The dataset is challenging to work with because the quality of videos is generally low (320x240px), dark and blurry because a lot are shot at night or in darker places as can be seen in various frames in Figure 3. Also, some videos are shot by infrared cameras at night resulting in a black and white image. Furthermore, a lot of videos contain black bars, timestamps, crops or other elements that might be confusing for a machine learning model. Besides that we are also dealing with overlapping classes e.g. a violent robbery can be classified as both a robbery and an assault in some cases, because it, for example, consists of people assaulting a cashier and then robbing the store.

3.2 Approach

For our approach we will use a quantitative analysis approach and come to conclusions using our metrics of evaluation described in section 4.2. We will conduct experiments on the HR-Crime dataset with newly trained models to find out if we can get comparable or improved performance in relation to previous research [1].

In this work, we will focus on using visual information, i.e. the frames as they look (we will either extract visual features or fine-tune networks) to make anomaly predictions. For this, we will be using the same train and test sets as in [1]. In addition, we will look into how the trajectories approached model divided the videos because we are interested in later comparing performance against those works. Also, we will extract visual features for both entire video frames as for separate areas of a frame that are of interest using bounding boxes around human subjects. The experiments can be divided into two methods to how we will classify this dataset:

3.2.1 Method 1. Classify frames directly. This uses the whole frame as the data source and also classifies them within the 13 categories. The steps for this will be first to subsample our video dataset to 1 frame per second for ease of training. Then run multiple configurations of a classification model to find the best results.

3.2.2 Method 2. Identify bounding boxes (previously detected by [1] with YOLOv3 [3]) around humans in a frame of a video and classify them within the 13 categories from the HR-Crime dataset. The steps for this will be to use the same subsampled frames from method 1. We will then combine them with the skeletal information provided by [1] and create bounding boxes around these skeletons. We will use the bounding boxes as input for our model and also apply data processing to possibly improve its results.

All in all our general pipeline is shown in Figure 2 to give a clear overview of these two methods. In this figure, we do not show the subsampling and we can also see the further steps taken on method 1 to find more information. This continuation of method 1 is described in Appendix A.1.

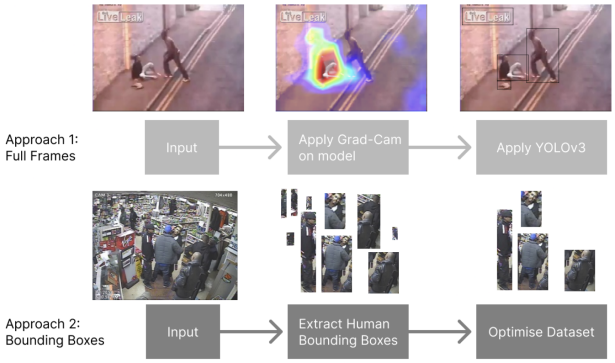
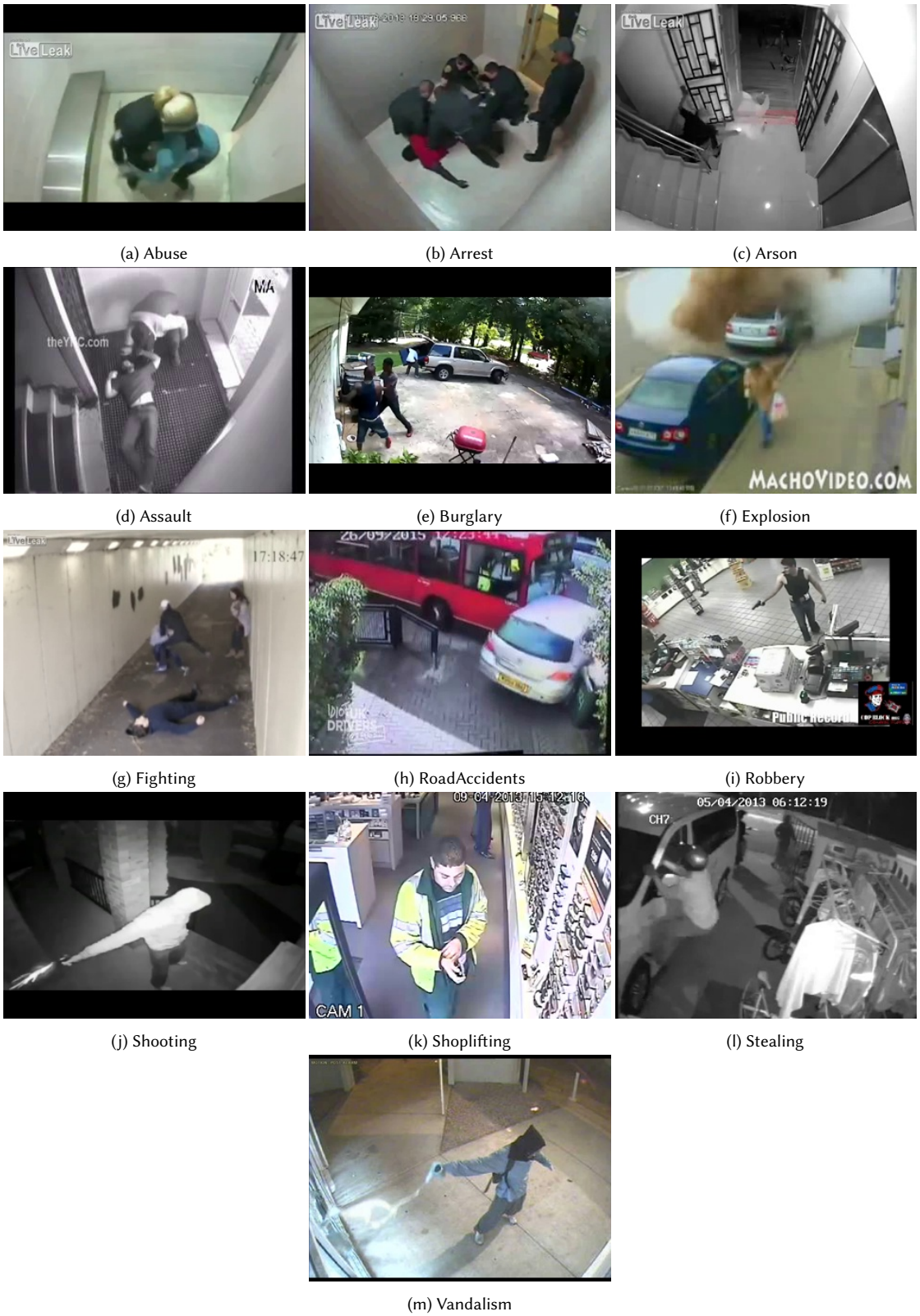


Fig. 2. Our feature extraction pipelines. Approach 1: First we train our model on the dataset of full images, then we apply Grad-CAM to find regions the model is predicting on, then we use those regions and apply object detection within them with YOLOv3. Approach 2: We extract human bounding boxes of video frames, and then we optimise the data to be used for training.

3.3 Feature Extraction

Afterwards, we will also make use of pre-trained networks and Convolutional Neural Networks (CNNs) [2] such as Squeezenet [4] and VGG [9] using PyTorch [6]. Then remove the softmax and



(a) Abuse (b) Arrest (c) Arson

(d) Assault (e) Burglary (f) Explosion

(g) Fighting (h) RoadAccidents (i) Robbery

(j) Shooting (k) Shoplifting (l) Stealing

(m) Vandalism

Fig. 3. From (a) to (m) examples of stills from the HR-Crime dataset: A guard excessively uses aggression on a detainee; cops arresting a person; a person setting fuel on the floor on fire; a person being assaulted in a stairway; people burglarizing a home; a car exploding; people fighting; a bus hitting a car; a person holding a cashier at gunpoint; a person shooting at a house; a person putting items in his sleeve; a person stealing a car; a person throwing paint at a shop.

extract features as the values of the last fully connected layer of the network. Finally, we will fine-tune our trained network using PyTorch [6]. In addition to these techniques, we will use a method to visualise what our model looks at when making predictions. We will be using Grad-CAM [8] for this. We will first use Grad-CAM [8] to give us heat maps of images showing where the model looks at. Following that, we will give insight into a possible approach that could be used to improve on this. This will be described and shown in Appendix A.1.

4 EXPERIMENTS

We conducted multiple experiments for our two approaches with varying results.

4.1 Dataset distribution and feature extraction

In these experiments the dataset we fed the model varied per approach. In the first approach, we first subsampled all the videos with 1 frame per second to scale down our otherwise big dataset to a more reasonable size and to possibly prevent overfitting. The distribution of this resulted dataset can be seen in Figure 4. This

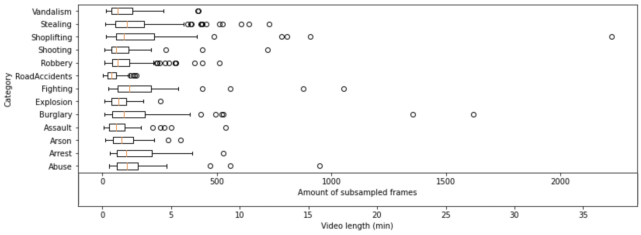


Fig. 4. Distribution of extracted frames and video length used for method 1.

shows that the category *RoadAccidents* consist of a lot shorter videos compared to *Shoplifting, Arrest, Burglary or Fighting*, for example. Notable is also the longest video which is a shoplifting video of 38 minutes. In addition to that, we can see what the results were of subsampling all our videos with one frame per second of video in Table 1. What becomes clear from the table is that our dataset is not evenly distributed. For example, *Fighting* occupies the largest percentage of our dataset with 15.8% contrary to *Explosions* with only 2.1%. This has to do with there being fewer explosion clips (26) compared to stealing clips (98). We can also look at the average frames per video to see that *RoadAccidents* clips are a lot shorter compared to *Shoplifting*. This is because the duration of the events is usually shorter resulting in a smaller amount of frames.

Besides that, the distribution of the second approach is shown in Figure 5 with black bars showing the distributions of the initial bounding boxes extracted and red bars showing the bounding boxes after pre-processing. For the second approach, we started with the same frames extracted by the first approach and combined them with the previously extracted human proposal bounding boxes [1]. The results of this can be seen in rows 4-6 in Table 1. It becomes clear from the table that shoplifting and robbery have the most bounding boxes extracted in total. This is a result of most of these videos occurring in crowded stores with relatively more human

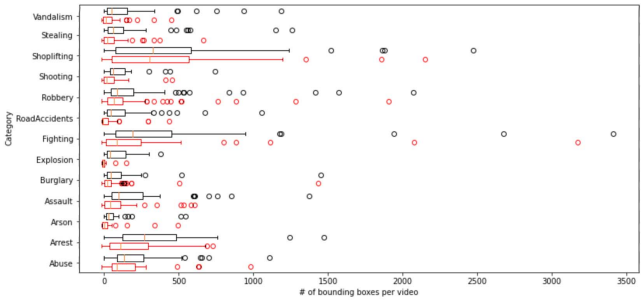


Fig. 5. Distribution of bounding boxes extracted per video category for method 2. Black bars indicate the number of bounding boxes extracted before pre-processing and the red bars indicate the same but after pre-processing i.e. removing images smaller than 25x25px

subjects. Therefore, more bounding boxes were extracted for these humans. This example illustrates the next problem that occurred. Which was that a lot of tiny images were being extracted because bounding boxes would be made around humans in the background of videos which most of the time had nothing to do with the actual crime being committed. These people are not interesting for our model and as a result, this introduced a lot of noise for training our model which we predicted would not be beneficial for the results of it. As a result, we experimented with removing small images that contained little data and settled on a minimum image size of 25x25 pixels because this gave us the best results and removed most tiny people in the background. The process of removing these images can also be seen in Approach 2 in Figure 2 and the resulted dataset statistics can be seen in the last two rows of Table 1. We can see that the dataset was reduced by about 50% (from 151164 frames to 100059) which removed a lot of noise and resulted in higher model accuracy.

4.2 Validation

Lastly, we will experiment with our model on the same test set that Boekhoudt et al. [1] used in their experiments. We will use the following evaluation metrics to evaluate our experiments:

- Accuracy, Precision, Recall, F-score, Weighted Accuracy, and Macro Accuracy: These will be used to get a basic idea of how well our model is performing in each category and how well our model performs in general.
- Area Under Receiver Operating Characteristic (AUROC): This will give us an idea of how well our model performs at class distinction.
- Confusion matrices for the best models: This will show us graphically what our model correctly classifies. Also, it shows which categories the model confuses generally and in which category it confuses it.

We will elaborate on our results using these metrics and compare them to the results from using a skeleton-based approach [1]. Using these metrics will give us clear quantifiable results that can clearly show which approach is favourable and thus enable us to answer our research questions.

Anomaly	Abuse	Arrest	Arson	Assault	Burglary	Explosion	Fighting	RoadAccidents	Robbery	Shooting	Shoplifting	Stealing	Vandalism	**Total**
Videos	38	42	48	47	96	26	39	68	145	46	50	98	46	**789**
Frames (1 per s)	5698	6430	4853	4022	15385	2072	7209	3195	13909	4579	10837	15466	4567	**98222**
Percentage of dataset	5.8	6.6	4.9	4.1	15.7	2.1	7.3	3.3	14.2	4.7	11.0	15.8	4.7	**100.0**
Average frames	149	153	101	86	160	80	185	47	96	100	217	158	99	-
Bounding boxes	9069	15323	3144	12958	9415	3762	18662	7904	22860	7911	20667	12210	7279	**151164**
Percentage of dataset	6.0	10.1	2.1	8.6	6.2	2.5	12.3	5.2	15.1	5.2	13.7	8.1	4.8	**100.0**
Boxes min (25x25)	7185	9309	2042	7550	7059	1205	12903	2304	17857	4568	18548	6707	2822	**100059**
Percentage of dataset	7.2	9.3	2.0	7.5	7.1	1.2	12.9	2.3	17.8	4.6	18.5	6.7	2.8	**100.0**

Table 1. Dataset statistics for both methods. The first two rows show the amount of video in the dataset and the amount of subsamples frames from the video dataset used in method 1. The last two rows show the percentage and average frames in the dataset. Rows 4-6 show the amount of extracted bounding boxes from the full frames and the percentage of the dataset used in method 2. The last two rows show the same data but with images smaller than 25x25 pixels removed.

4.3 Results

Following the feature extractions leading to our two datasets, we trained models using CNNs and pre-trained models in combination with kfold cross-validation to get the best results and minimize overfitting. We used our validation metrics as described in Section 4.2 to evaluate the best model. The results per method were as follows:

4.3.1 Results method 1: Visual information from entire frames. A model was trained multiple times in different ways to improve its performance. We experimented with pre-trained models as well as custom CNNs to find out what model would give the best results. In the first row of Table 2 the highest accuracy of this method can be seen. For the same train-test split as [1] our accuracy was 35% and when using 5-fold cross-validation with a CNN we obtained an accuracy of 54%, outperforming the previous result.

Model (CNN)	[1] split acc.	kfold acc.	std
Single-frame	35%	54%	3.8%
Bounding Boxes	15%	18%	1.9%
Bounding Boxes min25x25	26%	37%	1.5%

Table 2. Accuracy for methods 1 and 2. The second column for the model with the same train-test split as Boekhoudt et al. [1] and the third column for 5 fold K-fold accuracy. Also the last column shows the standard deviation of the kfold accuracy.

Following these results, the best model was evaluated in more depth resulting in an answer to our metrics of evaluation in Table 3. This table shows that our precision, recall and f1-score vary per category and our AUROC values lie between 0.50 and 0.78 for the categories *Explosion and Shoplifting* respectively. Lastly to be able to see how well method 1 performed on different classes the confusion matrix is shown in Figure 6.

Figure 6 gives us an indication of how well our model performs depending on the predicted classes (a diagonal dark blue line is ideal).

category	**precision**	**recall**	**f1-score**	**AUROC**
Abuse	0.60	0.46	0.52	0.66
Arrest	0.55	0.44	0.49	0.63
Arson	0.69	0.21	0.33	0.56
Assault	0.31	0.21	0.25	0.63
Burglary	0.48	0.76	0.59	0.71
Explosion	0.60	0.20	0.30	0.50
Fighting	0.53	0.71	0.61	0.70
RoadAccidents	0.26	0.36	0.30	0.51
Robbery	0.51	0.51	0.51	0.68
Shooting	0.86	0.22	0.35	0.52
Shoplifting	0.77	0.76	0.77	0.78
Stealing	0.53	0.67	0.59	0.74
Vandalism	0.61	0.21	0.31	0.60
accuracy			**0.54**	
macro avg	**0.56**	**0.44**	**0.45**	
weighted avg	**0.56**	**0.54**	**0.52**	

Table 3. Validation metrics for the best model of method 1 (higher is generally better).

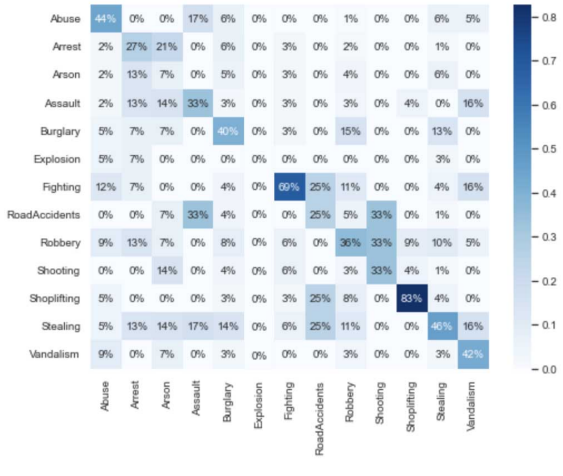


Fig. 6. Confusion matrix for method 1.

4.3.2 Results method 2: Visual information from bounding boxes around human subjects. For the second method, a model was trained in 4 configurations to find out which configuration would give the best results. These configurations were as follows: 2 configurations for bounding boxes and 2 configurations for bounding boxes with data pre-processing applied. The last two rows of Table 2 show the accuracy per model and we can conclude that the model with 5-fold cross-validation and data pre-possessing has resulted in the highest accuracy of 37%. In Table 4 we can see this model in more detail.

category	**precision**	**recall**	**f1-score**	**AUROC**
Abuse	0.33	0.44	0.37	0.68
Arrest	0.37	0.36	0.37	0.65
Arson	0.26	0.14	0.18	0.57
Assault	0.23	0.21	0.22	0.58
Burglary	0.36	0.30	0.32	0.63
Explosion	0.00	0.00	0.00	0.50
Fighting	0.43	0.48	0.46	0.69
RoadAccidents	0.20	0.10	0.13	0.54
Robbery	0.42	0.49	0.45	0.66
Shooting	0.13	0.02	0.04	0.51
Shoplifting	0.49	0.48	0.48	0.69
Stealing	0.22	0.27	0.24	0.60
Vandalism	0.15	0.07	0.09	0.53
accuracy			**0.37**	
macro avg	**0.28**	**0.26**	**0.26**	
weighted avg	**0.36**	**0.37**	**0.36**	

Table 4. Validation metrics for the best model of method 2 (higher is generally better).

The AUROC values also vary per category and the category *Explosion* has values that are 0.00 which is interesting. More on this will be explained in Section 5.2. Also, the AUROC values lie between 0.50 and 0.69 for *Explosion and Fighting* respectively. Furthermore, the confusion matrix of method 2 can be seen in Figure 7.

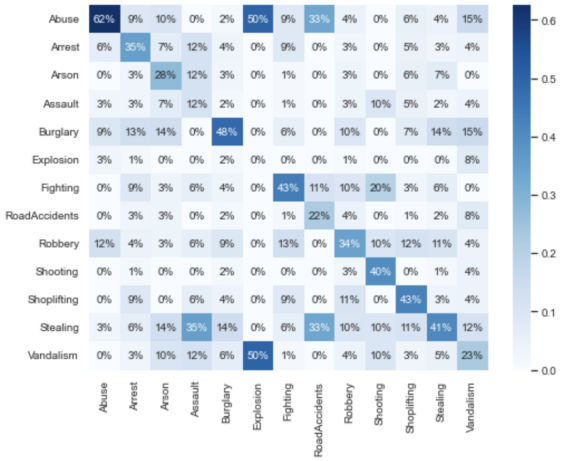


Fig. 7. Confusion matrix for method 2.

5 DISCUSSION

In addition to the results of Section 4.3 we will now give an analysis of our findings and what we can take away from these results.

5.1 Discussion method 1

For method 1 the best model configuration was the 5-fold CNN because this minimised overfitting and worked best for our dataset with an average accuracy of 53%. Besides that, we can deduct from Table 3 that classes such as *Fighting, Shoplifting* and *Burglary* have the highest f1-score and thus indicating that our model performs well in these classes. Contrary to classes such as *Assault, Explosion, Shooting and Vandalism*, confirming our predictions that these classes would be harder to predict due to actions being less clear and/or classes overlapping. What is also interesting is that the resulting AUROC values are comparable to those using a skeleton-based approach like Boekhoudt et al. [1]. In their research, the AUROC values were between 0.43 and 0.73 and ours are between 0.50 and 0.78.

In previous research *Arson and Explosion* had the lowest AUROC values of 0.43 and 0.47 [1] respectively. On contrary, our AUROC values in these instances are a bit higher: 0.56 and 0.50 indicating our class separation is more effective. This is however still low indicating that these two classes are hard to separate. This is likely a result of the training data of explosions containing a lot of frames that are not actually anomalous because explosions mostly span only a few frames. As a result, our model trains on a lot of normal frames without an explosion taking place making it very hard to predict this class right, this could be improved in the future as described in Section 7. In addition, this could also be a result of the explosion dataset consisting of only 2.1% of the dataset as seen in Table 1.

Continuing on this, we can also see a relation between the distribution of our dataset in Table 1 and the AUROC values in Table 3. We can see that categories that are less represented result in lower AUROC values. For example, *Explosion and RoadAccidents* have the lowest representation in our dataset with a percentage of the dataset being 2.1% and 3.3% respectively. As a result, the AUROC values are also low: 0.50 and 0.51. This could indicate that the distribution of our dataset has a direct or indirect effect on our model's performance.

Besides that, the confusion matrix also confirms our predictions that e.g. *Robbery* has a lot of false positives and false negatives with classes such as *Shoplifting* and *Stealing* and vice versa. This is not very surprising as these classes are all about stealing an object in different environments. As humans, we could hardly even separate these categories when looking at a single frame like our model does. We normally use the context of multiple frames to make this decision. Therefore, it is not surprising that our model has a hard time separating these classes.

All in all, the results of this first method using full frames give results lower than the skeletal approach [1] because our accuracy is lower. This is likely due to our dataset containing noise such as video intros, timestamps and logos which confuses our model (more on this in Appendix A.1). The skeletal approach [1] removes a lot of this noise and makes it likely better at making predictions. Besides

that, the dataset is also barely annotated which could greatly improve performance because classes such as *Explosion* span just a few frames. In general, the performance of the model is still reasonable with 53% accuracy and could be used for further implementation when for example combined with using the information of multiple frames. More on this in Section 7.

5.2 Discussion method 2

For method 2 we see results that show a lower performance compared to method 1. When looking at the AUROC values in Table 4 we can conclude a few things. First, is that we generally have more AUROC values close to 0.50 compared to the AUROC values of method 1 as seen in Table 3 and thus indicating that method 2 has a harder time with the distinction of classes compared to method 1. Second, we can also see a relation with the dataset distribution as shown in Table 1 and the AUROC values of classes such as *Explosion, RoadAccidents* and *Arson*. These classes have the lowest AUROC values and this likely is related to the dataset distribution being only 1.2%, 2.3% and 2.0% respectively after pre-processing as shown in the bottom row in Table 1. Because these classes are poorly represented in the dataset they also result in poor performance in our model which could be improved in the future by altering our bounding boxes dataset. This is similar to what we found in method 1 and could be generally seen as a possible improvement of the dataset.

Second, when looking at the confusion matrix we ideally want to see a dark blue diagonal line indicating that the model easily predicts a class right. This is however hardly the case in Figure 7 and compared to Figure 6 it also is more spread and shows less of a blue line like in Figure 6. This indicates a poor class distinction and confirms our predictions based on the generally low AUROC values close to 0.50 as shown in Table 4. Lastly, the class *Explosion* has zeros in its cross-section meaning it predicted 0 images right. This again is likely a result of it being poorly represented in the dataset and trained on.

Furthermore, this reduction in performance compared to method 1 is likely related to a lot of human bounding boxes being extracted of humans that are not actively engaging in the crime scene and the low quality of the footage. Besides that also a lot of tiny images were extracted of people in the far background as shown in step 2 of approach 2 in Figure 2. Even after pre-processing, this turned out to still have introduced a lot of images of people that were not interesting for our model. Besides that the uneven distribution of the resulting dataset also caused a few classes to perform worse than others.

All in all, the pre-processing of our dataset did improve this method by increasing the accuracy from 18% to 37%. However, in general, this approach is not the most useful approach when looking at other non-human indicators of what is happening in a scene. This is also why a class such as *Explosion* performs very poor. This method does not look at a fire or a big blast but more at how the humans react to it because that is what it is trained on. In addition, a lot of people are looked at that are not actually engaging actively in a scene which introduces a lot of noise.

6 CONCLUSION

In this paper, we discussed two methods of classifying anomalous events in surveillance footage using a visual approach. Throughout the paper, we compared the results of both methods to the works of Boekhoudt et al. [1]. By comparing our results we were able to give answers to our research questions:

- **Research question 1:** *Can we use the visual information from the entire frames to accurately classify human-related anomalies?*

When using visual information of entire frames we got an accuracy of 54% indicating that this is a technique that could indeed be useful to classify anomalies especially when it is used in combination with multiple frames or better datasets with less noise or better annotations.

- **Research question 2:** *Can we use visual information by using bounding boxes around human subjects to accurately classify human-related anomalies?*

When using bounding boxes it turned out that this gives reasonable results but is less accurate than method 1 or previous research [1]. When trying to improve this with data pre-processing this did improve the method but is not a very accurate way of making predictions with an accuracy of 37%. Because this method does not use objects in a scene this useful information is gone and likely contributes to the model not performing very well. Also just like method 1 the data and dataset distribution had a lot of issues that contributed to low performance.

- **Research question 3:** *Are the results from using skeletal trajectories [1] a better option compared to using visual information or bounding boxes?*

In general, using skeletal trajectories gives higher results than our single frame analysis. However, the skeletal trajectories approach used tracking of trajectories. If we would use some sort of multi-frame analysis our results would likely be higher or comparable.

7 FUTURE WORK

Both approaches could use improvements to make them perform better on this classification problem. For both methods, improvements could be made to the use of the HR-Crime dataset. First, the dataset itself could be extended to contain more data for classes that currently have little anomaly data. This could be done by gathering more video data for less represented classes. Second, the train videos could be annotated to mark anomalous events on a frame scale. As a result, the model can be trained on more precise data about anomalous events and can ignore long videos or video in -and outros. Third, some sort of voting or frame tracking could be implemented for both methods to improve accuracy when feeding multiple frames to the model.

In addition, improvements can be made to method 1 to improve its performance. Data augmentation or frame cropping could be used to test performance on frames without timestamps, logos or black crop bars because our model uses this as information as seen in Appendix A.1 and is not favourable.

Furthermore, for method 2 improvements could be made relating to useful objects in a scene. Now it only uses humans as a source of

information, while cars, weapons, etc. could be objects that contain information about a scene. If we would use this information in some way we would likely be able to better predict a scene. In addition, we could come up with an approach that only detects bounding boxes of human subjects actively participating in the scene to narrow down our dataset to important data (more on this in Appendix A.1). In addition, it could also track humans through frames to better build connections between the actions of subjects.

REFERENCES

[1] Kayleigh Boekhoudt, Alina Matei, Maya Aghaei, and Estefanía Talavera. 2021. HR-Crime: Human-Related Anomaly Detection in Surveillance Videos. In *International Conference on Computer Analysis of Images and Patterns*. Springer, 164–174.

[2] Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, et al. 2018. Recent advances in convolutional neural networks. *Pattern recognition* 77 (2018), 354–377.

[3] Zhanchao Huang, Jianlin Wang, Xuesong Fu, Tao Yu, Yongqi Guo, and Rutong Wang. 2020. DC-SPP-YOLO: Dense connection and spatial pyramid pooling based YOLO for object detection. *Information Sciences* 522 (2020), 241–258.

[4] Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. 2016. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. *arXiv preprint arXiv:1602.07360* (2016).

[5] Nancy G La Vigne, Samantha S Lowry, Joshua A Markman, and Allison M Dwyer. 2011. Evaluating the use of public surveillance cameras for crime control and prevention. *Washington, DC: US Department of Justice, Office of Community Oriented Policing Services. Urban Institute, Justice Policy Center* (2011), 1–152.

[6] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* 32 (2019).

[7] Sukanya Pillay. 2005. Video as evidence. *Video for change. A guide for advocacy and activism* (2005), 209–232.

[8] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*. 618–626.

[9] Abhronil Sengupta, Yuting Ye, Robert Wang, Chiao Liu, and Kaushik Roy. 2019. Going deeper in spiking neural networks: VGG and residual architectures. *Frontiers in neuroscience* 13 (2019), 95.

[10] Waqas Sultani, Chen Chen, and Mubarak Shah. 2018. Real-world anomaly detection in surveillance videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 6479–6488.

A APPENDICES

A.1 Appendix A.1 Grad-CAM Analysis

In this appendix, we will analyse and interpret our model's decision process by presenting what it looks at to determine its results.

Appendix A.1: Grad-CAM Analysis

DAVID D. W. ELSKAMP, University of Twente, The Netherlands

1 INTRODUCTION

In this appendix, we will analyse our model using Gradient-weighted Class Activation Mapping (Grad-CAM) [2]. For each category, we will present examples of what our model looks at for single full frames. In general, we will only be applying Grad-CAM [2] to method 1 because this gives us the most interesting information. Besides that, we will also further analyse these results and elaborate on what we can learn from them and present possible improvements and/or solutions.

2 GRAD-CAM RESULTS

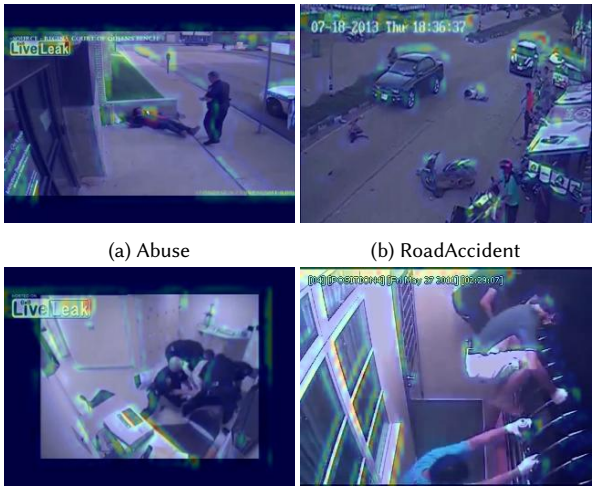
To begin, we applied Grad-CAM [2] to our model from method 1. In Figure 1 we present examples of results for every category.

2.1 Results per category

A green to red gradient indicates where and how concentrated our model looks at an image. For these categories we can take away a few things: First, we can generally see that our model is paying attention to mostly the same things as we humans would. Some clear examples of this are Figure 1f and Figure 1j. They both show a big red/orange zone around the subject or activity related to the crime. For the explosion, we clearly see a red and orange area around the explosion. Likewise for the shooting subject where Grad-CAM [2] shows a green to red area around the subject and also an area around the muzzle flash. This indicates that our model does indeed look at regions that are related to the crime and are important to get information from.

2.2 Analysis of problems

When looking at multiple frames analysed by Grad-CAM [2] we can see a few interesting results that confirm the problems we experienced in the performance of our model. In Figure 2 we can see examples of images in 4 categories where our model uses the wrong regions of interest to classify a crime. In all 4 images, we can see that things such as timestamps, logos (LiveLeak in this case) and other text in images confuse our model. This is unsatisfactory because we do not want our model to learn based on logos or text that are common in some categories. Moreover, we can also see in Figure 2c that it is not only looking at the logo but also at the regions in the black bars. This indicates that it is looking at data that is useless and could have been changed in the dataset.



(a) Abuse (b) RoadAccident

(c) Arrest (d) Burglary

Fig. 2. Examples of unsatisfactory attention zones from Grad-CAM [2]. From (a) to (d): An officer abusing a detainee, a car crash with a motorcycle, a man being arrested, people climbing a fence for a burglary.

In addition to that, we can also see in Figure 1 that our model is also sensitive to bright white lines or objects in general as for example seen in Figure 1k where the white floor lines are brightly green. This can also be seen in Figure 1a where the white lamp at the top of the frame is looked at. Why this is happening is not exactly clear to us, however, it could have something to do with brightly lit areas being easier to classify since our dataset consists of a lot of badly lit environments. It could also have something to do with the timestamps and logos again since these are also generally white and could have slightly confused our model.

3 FINDINGS AND POSSIBLE IMPROVEMENTS

Our findings from using Grad-CAM [2] confirm the predictions we had in the discussion part of the paper are likely true. Since Grad-CAM [2] shows that our model pays attention to logos, timestamps and cropping we could use more data augmentation and possibly change our dataset to prevent this. This could be done by adding more videos without visual distractions or removing existing videos from our dataset. In addition, we could possibly identify these regions with something such as YOLOv3 [1] and then blur them to prevent our model from paying much attention to it. All of this combined would probably increase the accuracy of our model.

In addition, this could possibly also be used to improve method 2. As we found in our discussion, our model trains on a lot of data of people that are not actively engaging in a crime or in the scene in general. However, for those people bounding boxes are still extracted and used. It would be better to remove these people from our dataset and only train on bounding boxes around human subjects that actively engage in a crime or a scene. If we would combine

TScIT 37, July 8, 2022, Enschede, The Netherlands

(a) Abuse (b) Arrest (c) Arson (d) Assault

(e) Burglary (f) Explosion (g) Fighting (h) RoadAccidents

(i) Robbery (j) Shooting (k) Shoplifting (l) Stealing

(m) Vandalism

Fig. 1. From (a) to (m) examples of what our model looks at from using Grad-CAM [2]: A woman being abused, a man being arrested, people lighting a truck on fire, a man being assaulted in an elevator, guys burglarizing a house, an explosive exploding, people fighting, a car flipped after a crash, a robbery at gunpoint, a shooting at a house, a shoplifter in a store, a car being stolen, a car being vandalised with paint.

the areas extracted by Grad-CAM [2] with YOLOv3 [1] we could possibly extract bounding boxes only in these important regions. This could be done by simply leaving out bounding boxes outside of the important regions. Moreover, this could also be used to improve the dataset for method 1 if we would for example blur parts outside the important regions. That way we could improve our datasets and probably remove a lot of noise.

REFERENCES

[1] Zhanchao Huang, Jianlin Wang, Xuesong Fu, Tao Yu, Yongqi Guo, and Rutong Wang. 2020. DC-SPP-YOLO: Dense connection and spatial pyramid pooling based YOLO for object detection. *Information Sciences* 522 (2020), 241–258.

[2] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*. 618–626.