Automotive Interior Sensing - Towards a Synergetic Approach between Anomaly Detection and Action Recognition Strategies

Pedro Augusto
Faculty of Engineering of the University of Porto
Email: up201503495@fe.up.pt

Jaime S. Cardoso
INESCTEC and University of Porto
Email: jaime.cardoso@inesctec.pt

Joaquim Fonseca
Bosch Car Multimedia
Email: Joaquim.Fonseca2@pt.bosch.com

Abstract—With the appearance of Shared Autonomous Vehicles there will no longer be a driver responsible for maintaining the car interior and well-being of passengers. To counter this, it is imperative to have a system that is able to detect any abnormal behaviors, more specifically, violence between passengers. Traditional action recognition algorithms build models around known interactions but activities can be so diverse, that having a dataset that incorporates most use cases is unattainable. While action recognition models are normally trained on all the defined activities and directly output a score that classifies the likelihood of violence, video anomaly detection algorithms present themselves as an alternative approach to build a good discriminative model since usually only non-violent examples are needed. This work focuses on anomaly detection and action recognition algorithms trained, validated and tested on a subset of human behavior video sequences from Bosch’s internal datasets. The anomaly detection network architecture defines how to properly reconstruct normal frame sequences so that during testing, each sequence can be classified as normal or abnormal based on its reconstruction error. With these errors, regularity scores are inferred showing the predicted regularity of each frame. The resulting framework is a viable addition to traditional action recognition algorithms since it can work as a tool for detecting unknown actions, strange/violent behaviors and aid in understanding the meaning of such human interactions.

Index Terms—Action Recognition, Anomaly detection, Computer Vision, Deep Learning.

I. INTRODUCTION

Driverless cars are emerging as a technology that allows a more accessible and dynamic form of mobility and are expected to revolutionize daily transportation. One of the conceivable services based on driverless cars is shared autonomous vehicles (SAVs). This service could merge cabs, carsharing, and ridesharing systems into a singular transportation mode, [1]. However, without the driver’s interior supervision, the users of this service may encounter unpleasant situations that could threaten their well-being and comfort such as littering, smoking and worst of all - violent events. For these reasons, an intelligent system is needed to detect violent behaviors and interactions. With the help of cameras placed in the interior of the vehicles, all the rides can be monitored and processed by an adequate violence detection algorithm that detects violent interactions and triggers pre-established counter-measures. Numerous works covering video anomaly detection in crowded scenes have been published, [2], [3], [4], [5], [6], [7]. However, abnormal behavior detection between passengers in the interior of shared autonomous vehicles has yet to be covered in detail making this one of the first studies to do so.

II. RELATED WORK

Anomaly detection refers to the problem of finding patterns in data that do not conform to the expected behavior [8]. These non-conforming patterns are often referred to as anomalies, outliers, exceptions or even contaminants in different application domains. The importance of detecting such anomalies lies in the fact that abnormal data often translates to significant or even critical actionable information [8]. Efforts have been made to equate video anomaly detection with binary classifiers, but as abnormal video sequences are rare, training these conventional classifiers in a fully supervised way can be impractical. For this reason, the majority of video anomaly detection algorithms have semi-supervised or unsupervised natures. The semi-supervised methods have an explicit training phase, in which models are constructed from normal samples, whereas the unsupervised techniques do not require any offline training. With Deep Learning, neural networks are now complex enough to learn video representations and extract features from both spatial and temporal dimensions. The majority of the State-of-the-Art Video Anomaly Detection algorithms today apply deep learning techniques [5], from which many fusion schemes, predictive and generative models have shown the best overall results in recent years. Among these categories, Variational Autoencoders (VAEs) and Convolutional Long-Short Term Memory Networks (ConvLSTMs) tend to perform better, as demonstrated in [9] on the UCSD ped 1, UCSD ped 2 and CUHK Avenue datasets. The S2-VAE and Spatiotemporal AE frameworks, presented in [5] and [10], respectively, prove the superiority of these types of deep autoencoder architectures.

On the other hand, action recognition algorithms have been largely driven by the advances in image recognition
methods, which were often adapted and extended to deal with video data. Many deep architectures for video recognition have been developed where, as for video anomaly detection, the input of the networks is a stack of consecutive video frames. The models are normally deep convolutional networks and are expected to implicitly learn spatio-temporal motion-dependent features. In the seminal work of [11], state-of-the-art action recognition architectures were evaluated in light of the Kinetics Human Action Video dataset. The authors provided an interesting analysis on how current architectures improve their performance on the benchmark datasets of UCF-101 and HMDB-51 after pre-training on the Kinetics dataset. Additionally, the proposed Two-Stream Inflated 3D ConvNet (I3D) architecture allows to learn seamless spatio-temporal feature extractors from video while leveraging successful ImageNet architecture designs. There have also been other efforts to split spatial and temporal components, as the one presented in [12]. In this particular two-stream architecture, the spatial stream was trained on individual video frames and the temporal stream was trained on multi-frame optical flow fields. In [13], the strategy is inherently different as each stream is strengthened with the knowledge from the other stream, allowing for cross-communication. This work presented many multiplicative fusion approaches for the combination of CNNs which were demonstrated to lead to better performance when compared with traditional additive approaches.

The majority of studies performed on in-vehicle sensing applications focus on the driver’s behavior or prediction of the driver’s activities, [14], [15], [16]. From our knowledge, no studies have been published regarding passenger interaction recognition or abnormal passenger activity detection. These topics are challenging in the way that a strong definition of non-violent and violent activities is needed as well as the employment of architectures capable of discerning subtleties and variations in the passengers’ interactions. In the following sections, the video anomaly detection framework adapted to in-vehicle video data is presented as well as its main results. Additionally, a new framework combining video anomaly detection and action recognition frameworks is proposed for future work.

III. SYSTEM ARCHITECTURE

To detect abnormal human interactions inside vehicles in a semi-supervised manner, the Spatiotemporal autoencoder framework from [10] was employed. The framework has been featured in other works, [17] and [18], displaying robustness as well as the necessary flexibility and adaptability properties needed for its use in different scenarios. The architecture features an autoencoder which is used to learn regularity in video sequences so that the reconstruction of regular video sequences results in a low reconstruction error and the reconstruction of irregular (unknown) video sequences results in a high reconstruction error. The framework has a sequence of three main steps - Preprocessing, Feature Learning and Anomaly Detection. In the Preprocessing stage, the global mean frame from all the training frames is calculated and then all the training, validation and testing frames are converted to grayscale, normalized (on a pixel level), resized to 224x224 and subtracted by the global mean frame. Then, a sliding window with stride $S = 1$ and window length $T = 10$, where $S$ is the amount of shift by frames in subsequent windows, is applied on each training, validation and testing video, resulting in 10 frame sequences, or volumes, see figure 1. As an example, if a video has $V = 200$ frames, the number of volumes $N$ and their reconstruction errors resulting from applying the aforementioned sliding window technique can be calculated with equation 1, leading to $N = 191$ [18].

$$N = \left\lfloor \frac{V - T}{S} \right\rfloor + 1$$

Figure 1. Preprocessing performed on the training, validation and testing frames according to the Spatiotemporal autoencoder framework.

In the Feature Learning process, the model is trained to minimize the reconstruction error between input video volumes and the respective output video volumes reconstructed by the model. The framework features a Spatiotemporal autoencoder for learning spatial structures of each frame and a temporal autoencoder for learning temporal patterns of the encoded spatial structures. The spatial encoder and decoder have two convolutional and deconvolutional layers, respectively, while the temporal encoder/decoder is implemented using a convolutional LSTM with three layers. The LSTM network is used for its property of leveraging the history of inputs to learn temporal dependencies between sequential inputs and time series data. The convolutional LSTM variant is utilized instead of a regular LSTM, for the purpose of feature extraction, due to its ability to preserve the spatial relationship between the frames’ pixels. A detailed representation of the model is presented on figure 2.

In the Anomaly Detection stage, as in [10], the reconstruction error of all the pixel values in frame $t$ of a test video sequence is taken as the Euclidean distance between the input frame $x(t)$ and the reconstructed frame $f_W(x(t))$, see equation 2.

$$e(t) = ||x(t) - f_W(x(t))||_2$$
Having a lot of different actors shall lead to a better generalization ability post training but could distract the network from focusing on the actors movements. Using only the human body pose information of the two subjects could help in reducing the variability of human physiognomy and body composition but it would imply the removal of the important hand and finger information necessary to differentiate between similar activities. Having this in mind, this alternative preprocessing procedure was not implemented, which is why the one mentioned in section III was adopted instead.

Making use of these two major data splits, two variants for Training, Validation and Testing were constructed:

- **IVS Talk** - This in-vehicle sensing (IVS) split used a simple data distribution, gathering only the Talking, Talk_Left and Talk_Right clips from the TRAIN_VAL_SET as the normal class. The split has a varied number of actors for training and validation and testing was performed with videos from the TEST_SET which allowed to evaluate the framework on its ability of generalization of activities between different actors. Videos containing Unlabeled interactions were removed from testing since the model was not trained or validated with any Unlabeled action clips.

- **IVS Violence** - This additional IVS split was conceived with the goal of evaluating the framework in the difficult use case of detecting violent interactions. This implied that the non-violent activities from the TRAIN_VAL_SET selected for training and validation (normal class) included static activities as well as dynamic non-violent interactions such as the complex Hugging activity and many Unlabeled behaviors. Unlike in IVS Talk, the system was also trained, validated and tested with Unlabeled behaviors.
V. RUN-TIME ANALYSIS

As previously mentioned, all the frames from the Training, Validation and Testing sets are preprocessed before initiating any of the different stages. It involves calculating the global mean frame from the training data and subtracting it from the training, validation and testing data. Then, for all the sets (training, validation and testing), spatiotemporal volumes (10 frame sequences) are computed and stored for future use. In terms of testing, the main output of the framework are the regularity scores, calculated on a per video basis, and stored in a data file. With the scores of each frame from the data file and the video’s ground truth data, stating which frames are abnormal, the complete video regularity score graph is generated. Having this in mind, the run-time analysis of the anomaly detection framework running on a Xeon Gold 6150 CPU and a Tesla V100-SXM2 32GB GPU is in table II. The analysis can be split into three parts: mean frame subtraction, volume building and prediction (which corresponds to the computation of the regularity score data file). The stage that takes on average the least amount of time to complete is the mean frame subtraction and the one that takes a longer period of time to complete is the prediction stage. Nonetheless, the total time of computation per frame is less than 50 ms. As the system stands, it cannot work in a real-time scenario because the preprocessing stages are conducted for all the testing videos before the testing stage may begin, but with real-time preprocessing, proper network compression and optimization, it is a realistic scenario in future iterations of the algorithm.

<table>
<thead>
<tr>
<th>Statistics</th>
<th>Mean frame subtraction</th>
<th>Volume building</th>
<th>Prediction</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>9.621</td>
<td>16.75</td>
<td>20.94</td>
<td>47.3</td>
</tr>
<tr>
<td>Standard deviation</td>
<td>1.01</td>
<td>4.15</td>
<td>1.24</td>
<td>-</td>
</tr>
</tbody>
</table>

VI. EXPERIMENTS AND RESULTS

According to the definitions of the normal and abnormal classes, groundtruth files were generated for the 34 and 35 testing videos, for IVS Talk and IVS Violence, respectively. As the majority of other video anomaly detection studies, to measure the performance of the framework on the two data splits, the overall frame-level AUC scores were calculated. It can be seen on table III that the maximum AUC scores on these data splits were above 74%.

<table>
<thead>
<tr>
<th>Data split</th>
<th>Frame-level AUC score</th>
</tr>
</thead>
<tbody>
<tr>
<td>IVS Talk</td>
<td>74.88%</td>
</tr>
<tr>
<td>IVS Violence</td>
<td>78.89%</td>
</tr>
</tbody>
</table>

Looking at the regularity scores on figure 3, it can be seen that the anomaly detection framework presented good discrimination between non-violent interactions (in white) and violent interactions (in red). On video 22, the Punching activity is the violent action performed between the two actors. In this particular clip, the framework showed the best discrimination from all the testing set decisively detecting the Punching action performed repeatedly for the first 465 frames of the video for either a 0.5 or 0.7 threshold. On video 25, besides some stationary (non-violent) moments, only Fighting sequences between the two actors took place. The anomaly detection framework showed good discrimination on the first two fighting sequences but the last sequence showed a slightly higher regularity score, compromising the classification of some frames. As with the test video 22, a threshold of 0.7 would lead to a greater number of frames being correctly classified as abnormal, with a residual increase of the false positive rate.

The Punching action from test video 22 and the Talking action from test video 25 can be seen on figure 4. To visualize the regularity score in real-time alongside the video footage, a visualization tool was developed. It shows a binary ground truth indicator displaying the green color if the current frame being presented is non-violent or the red color if it is violent and a gauge with the framework’s prediction of how regular (non-violent) the frame is. This tool was essential in validating the performance of the algorithm and perceive how it responded according to the actors’ movements.

![Figure 3. Regularity scores (in blue) obtained by the anomaly detection framework for testing videos 22 and 25 of IVS Violence. To ease analysis, two thresholds were added to the graphs - the 0.5 and 0.7 thresholds, in red and green respectively.](image)

Additionally, a test was conducted to measure the model’s performance based on how well it classified each action from the test set as violent (abnormal) or not. To do so, the best performing anomaly detection model on IVS Violence’s test set was used (AUC score of 78.89%). Using this model, the error rates (ERR = 1 - ACC) were calculated for each specific action, (ACC stands for Accuracy). To measure the error rate of the framework on all classes, the (continuous) regularity scores of all frames were converted to 0 or 1 values depending on the imposed threshold. Having an extremely low threshold results in high error rates for the violent actions and low error
rates for the non-violent actions. This happens because any regularity score above the threshold is converted to 1, which means that the majority of abnormal actions are classified as being non-violent, and almost all non-violent actions are correctly classified. Similarly, for high thresholds, most of the non-violent actions with regularity scores below the threshold are misclassified as abnormal actions and violent actions end up being correctly classified as such. For this reason, it is important to set a threshold that ensures the least error rate across all the activities. The lower range of thresholds provide the best performance for the detection of normal actions but as this system is primarily being used as a violence detector, it is arguably more important to encounter less false negatives than false positives. Taking this into account, the 0.7 threshold was selected as it showed the least amount of error rate for the violent actions while presenting fairly low error rates for the detection of non-violent actions. The resulting true positive, false negative, true negative and false positive rates for the IVS Violence testing set are displayed on table IV.

<table>
<thead>
<tr>
<th>Data split</th>
<th>TPR</th>
<th>FNR</th>
<th>TNR</th>
<th>FPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>IVS Violence</td>
<td>0.79</td>
<td>0.21</td>
<td>0.67</td>
<td>0.33</td>
</tr>
</tbody>
</table>

The anomaly detection algorithm presents an overall good performance in correctly detecting the violent classes but a slightly high false positive rate due to higher error rates when classifying the Arguing and Talking actions. Despite sharing similarities with the Talking activity, Arguing can be sometimes considered violent, depending on the context. Adding audio data captured from the scenes would help to discern between Arguing and all the other activities, reducing the number of misclassified occurrences.

Bearing this in mind, it can be concluded that the anomaly detection approach is a viable alternative to action recognition algorithms in applications where a higher false positive rate is acceptable. If for a particular application, the false positive rate must be very low, then action recognition algorithms as the ones mentioned in section II may be better approaches for detecting violent interactions inside vehicles. On the other hand, as there is an overwhelming variety of human behaviors, some cannot be represented in a reasonable number of classes to train action recognition architectures. Moreover, some interactions are so rare that there is not enough data available for effective training. Anomaly detection algorithms could support action recognition systems to detect these sparse actions and movements that could be later featured in new classes. A possible system architecture featuring a cascade of the two alternatives can be seen on figure 5. The action recognition model would discriminate the video sequence first as being non-violent or violent. Then, the anomaly detector, with its knowledge of only non-violent activities, would classify the video sequences marked as non-violent by the action recognition model as non-violent or violent. In theory, this synergy of the two approaches would give more robust results than each system on its own.

![Figure 5. Action recognition and anomaly detection cascade architecture.](image)

VII. CONCLUSION

The emergence of SAVs have led to major interest in the scientific community in developing the necessary systems that compensate the absence of the driver. Besides engine control and steering, interior perception is of utmost importance to ensure the safety of the passengers. This work focused on human interactions inside the vehicles, more specifically the detection of abnormal behaviors.

Semi-supervised deep learning models have shown great performance and practicality by performing one-class classification, i.e., learning how to reconstruct normal frame sequences so that during testing, poor reconstruction leads to
the detection of unknown (abnormal) frame sequences. Having this in mind, the Spatiotemporal autoencoder from [10] was employed on two subsets with increasing complexity from the in-vehicle datasets from Bosch Car Multimedia. The first dataset, IVS Talk, had the abnormal class defined as any action different from conversational activities. The second dataset, IVS Violence, was constructed to test the system on a violence detection scenario. The AUC scores obtained, particularly the one on IVS Violence, reflect the strong discrimination power of the anomaly detection framework even on the more complicated use case of violence detection.

It was concluded that the anomaly detection approach can support state of the art action recognition algorithms to detect unknown actions and movements. Still, probably the greatest advantage of adopting an anomaly detection system, particularly a deep learning one, is its semi-supervised nature which allows the definition of only the expected normal data, the use of datasets with reduced labeling, and faster deployment since less time is spent defining hand-crafted features.

VIII. FUTURE WORK

In terms of future work, an interesting solution would be the use of optical flow volumes, which according to the work of [17] improved the AUC score significantly when compared with the work of [10] on UCSD ped 1, ped 2 and CUHK Avenue. This solution could improve the results for parked vehicles by emphasizing motion patterns but would certainly struggle with moving vehicles due to variations on light conditions and constant change of outdoor elements. Besides the inputs, changes could also be made to the framework. The threshold applied to the regularity scores can be changed according to the dataset being used but it could be interesting to have it optimally predicted by an auxiliary system such as a Kalman filter or a Recurrent Neural Network. Moreover, convolutional layers could be added to the spatial encoder and decoder structures and/or adopt a fusion scheme where the raw frames could be combined with audio data to improve the overall discrimination ability.

The cascade architecture presented on figure 5 could also be implemented to infer if the anomaly detection framework and an additional action recognition framework can be used together to take advantage of the two approaches, while minimizing the problems associated with each one when used individually.

ACKNOWLEDGEMENTS

We would like to thank the Faculty of Engineering of the University of Porto and Bosch Car Multimedia for allowing the use of its computing infrastructure, in-vehicle datasets and providing continuous support during the development of this project. This work was partially supported by European Structural and Investment Funds in the FEDER component, through the Operational Competitiveness and Internationalization Programme (COMPETE 2020) [Project no. 039334; Funding Reference: POCI-01-0247-FEDER-039334].

REFERENCES