Violence Recognition in Surveillance Videos with Dense Networks

Askarbek Assubayev [1], Aizhan Altaibek[1,2], Marat Nurtas[1,2] and Aigerim Altayeva[3]

[1] International Information Technology University, Manas St. 34/1, Almaty, 050000, Kazakhstan
[2] Institute of Ionosphere, Gardening community IONOSPHERE 117, Almaty, 050020, Kazakhstan
[3]Al-Farabi Kazakh National University, al-Farabi Avenue 71, Almaty, 050040, Kazakhstan

Abstract

Security camera-based surveillance systems are becoming the main tool in public settings, utilizing computer vision and machine learning techniques for diverse applications in safety monitoring. The proposed model is a spatial feature-extracting DenseNet-121 followed by convolutional LSTM for temporal feature extraction and classification. In this paper, we propose a novel architecture for violence detection from various video data of surveillance cameras. The proposed model is called DenseNet-LSTM it demonstrates efficient computational performance while achieving good results. The model is evaluated on a diverse set of violent action datasets such as Hockey, Movies, Movie Datasets, Violent Flow, and Real-Life Violence Situations (RLVS). DenseNet-LSTM promises near state-of-the-art performance and accuracy on most of them while requiring low computational time.

Keywords

Violence recognition, surveillance, deep learning, computer vision, DenseNet

1. Introduction

Video surveillance, commonly referred to as CCTV (Closed Circuit Television) [3], has become an integral tool for enhancing security and monitoring in various environments. In urban settings such as cities and town centres, video surveillance is deployed to deter criminal activities, including theft, vandalism, and public disturbances. The presence of visible cameras acts as a powerful deterrent, discouraging potential wrongdoers from engaging in illicit activities [2]. In commercial establishments like malls, banks, and retail stores, video surveillance serves both as a proactive security measure and a monitoring tool. It helps in safeguarding assets, ensuring the safety of employees and customers, and investigating any incidents that occur within the premises. Video surveillance is a versatile and indispensable technology, playing a vital role in maintaining security and offering valuable insights for effective monitoring across a wide range of environments. The utilization of data captured by video surveillance-based systems can be used as a powerful tool to detect and prevent violent events. Nonetheless, the majority of individuals perceive video surveillance as an intrusion into their privacy [1]. There's apprehension regarding the potential uses or abuses of video recordings, particularly with advancements in contemporary image-processing technologies.

This research work proposes a low-weigh, efficient deep-learning method to classify violent and non-violent actions in security camera footage. The primary objective is to conduct an in-depth exploration and analysis of the DenseNet [4] network for spatial feature extraction. This entails a comprehensive examination of the network's capabilities and characteristics in the context of spatial feature extraction. Additionally, the research aims to identify and propose optimal solutions to seamlessly integrate the spatial feature extraction model with a temporal

DTESI 2023: Proceedings of the 8th International Conference on Digital Technologies in Education, Science and Industry, December 06–07, 2023, Almaty, Kazakhstan

36005@iitu.edu.kz (A. Asubayev); aizhan.altaibek@yandex.ru (A. Altaibek); maratnurtas@gmail.com (M. Nurtas); aikosha1703@gmail.com (A. Altayeva)

0000-0001-8431-7950 (A. Altaibek); 0000-0003-4351-0185 (M. Nurtas); 0000-0002-9802-9076 (A. Altayeva)

feature extraction model, achieving an efficient and effective fusion of spatial and temporal information for further violent event recognition.

2. Proposed method

The main goal of the proposed method is to conduct an analysis of DenseNet architecture for video classification that is comparable with the efficiency of state-of-the-art models while maintaining low classification time per frame.

The algorithm under consideration primarily consists of three main stages:

1. Spatial feature extractor
2. Temporal feature extractor
3. Classifier

Fig. 1 shows the architecture of the proposed method. First pre-processing steps are applied to the input video frames. The next two consecutive stages of feature extraction are applied; a DenseNet-121 model stage which is responsible for spatial feature extraction for each frame, and a ConvLSTM [5] stage which works as a temporal features extractor. Finally, the extracted features are fed to Dense layers for classification.

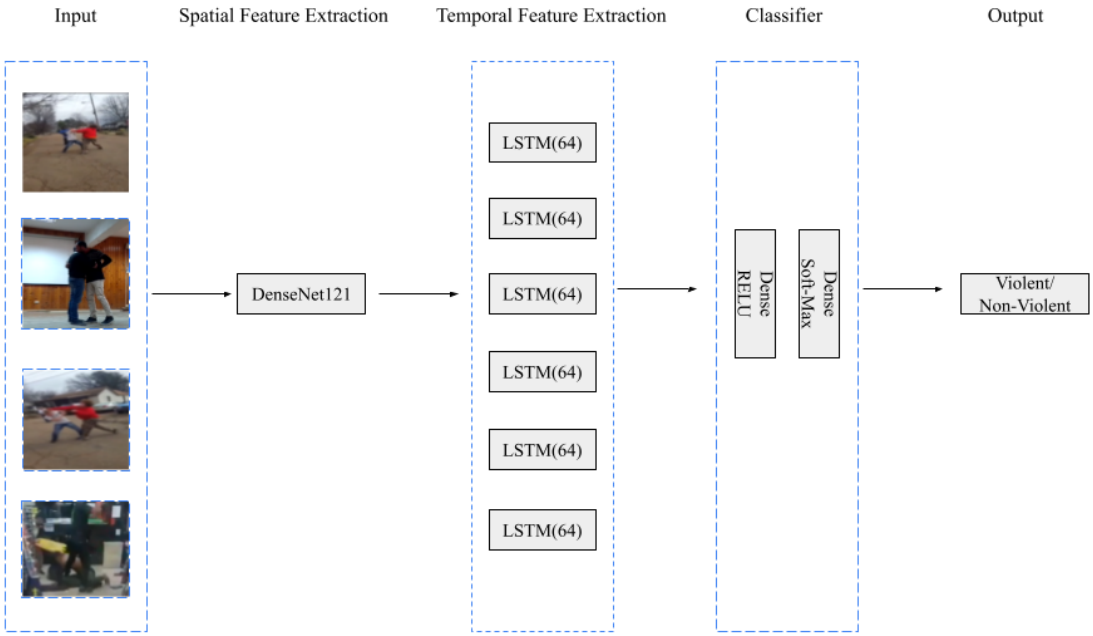

Figure 1: DenseNet-LSTM network model

DenseNet is a feed-forward Convolutional Neural Network (CNN) [6] architecture that connects each layer to every other layer. DenseNet design is founded on a straightforward and basic principle: by concatenating the feature maps of all previous layers, a dense block [5] allows each layer to access the features of all preceding levels. In classic CNNs, each layer only has access to the characteristics of the layer immediately before it. The architecture is arranged of transition layers and dense blocks. Each block is connected with a convolutional layer inside a dense block

that is connected to every other layer within the block. The transition layers minimize the size of the feature maps across dense blocks letting the network grow effectively.

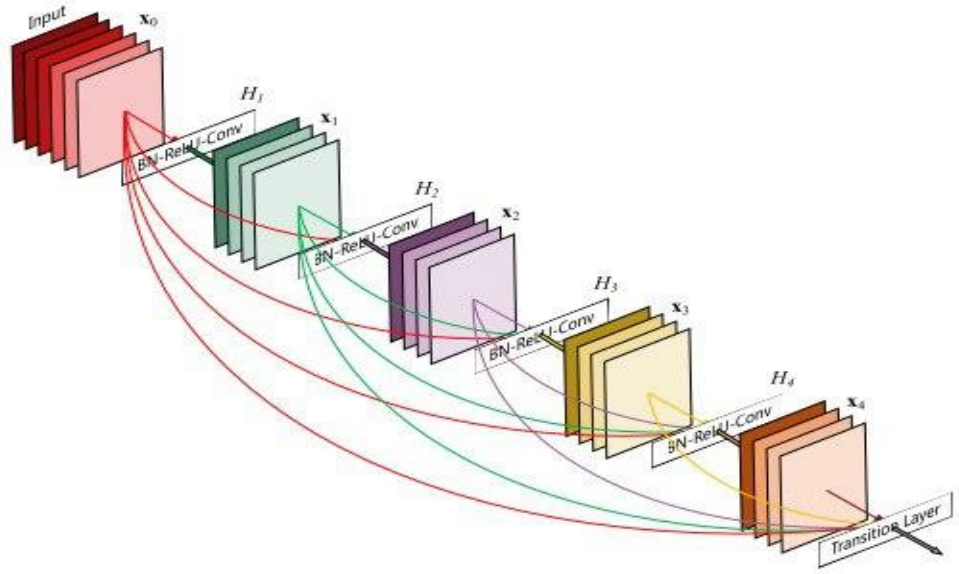

Figure 2: A 5-layer dense block with a growth rate of k = 4. Each layer takes all preceding feature maps as input. (2016). Densely connected convolutional networks. (Cornell University)

Figure 2 in the paper Densely Connected Convolutional Networks [4] shows a comparison between the traditional convolutional network and the DenseNet architecture. In the DenseNet architecture, each layer is connected to all subsequent layers. This dense connectivity pattern allows for a better flow of information and gradients throughout the network, making it easier to train. Each layer has direct access to the gradients from the loss function and the original input signal, leading to implicit deep supervision, which helps in the training of deeper network architectures. Further, dense connections have a regularizing effect, which reduces overfitting tasks with smaller training set sizes.

The benefits of using DenseNet as a feature extractor for computer vision tasks are that DenseNets naturally integrate the properties of identity mappings, deep supervision, and diversified depth. They allow feature reuse throughout the networks and can consequently learn more compact and, according to the experiments conducted by the authors, more accurate models. Because of their compact internal representations and reduced feature redundancy, DenseNets are a good feature extractor for violence recognition.

ConvLSTM [5] is a type of neural network architecture that extends the idea of LSTM (Long Short-Term Memory) to have convolutional structures in both the input-to-state and state-to-state transitions. This allows the network to capture spatiotemporal correlations in the input data, making it particularly useful for spatiotemporal sequence forecasting problems such as precipitation nowcasting. The ConvLSTM layer preserves the advantages of LSTM but is also suitable for spatiotemporal data due to its inherent convolutional structure. The network captures spatiotemporal correlations by extending the idea of FC-LSTM to have convolutional structures in both the input-to-state and state-to-state transitions. By stacking multiple ConvLSTM layers and forming an encoding-forecasting structure, the network can learn complex spatiotemporal patterns in the dataset through its nonlinear and convolutional structure. This allows the network to handle the strong spatial correlation in the radar maps and discover sudden changes in the encoding network, which is difficult to achieve with other methods like optical flow and semi-Lagrangian advection-based methods.

According to sequence-to-sequence learning [7] framework that can be used for a wide range of spatiotemporal sequence forecasting problems, including video classification and action

recognition in computer vision. The ConvLSTM network can be used as a building block in this framework to extract temporal features from the input sequence, which can then be fed into other layers of the network for further processing. By stacking multiple ConvLSTM layers and forming an encoding-forecasting structure, the network can learn complex spatiotemporal patterns in the dataset and generate accurate predictions.

2.1. Data Preprocessing

The videos from the dataset are input as shapes of (V, F,128,128,3): V is the number of videos, F number of video frames, and (128,128) height. These input videos are processed in the next pipeline; Every 3rd frame is skipped to a lower amount of duplicate frames; Each frame is resized to (128,128,3) shape; Data augmentation consists of converting frames to grayscale and vertical flipping. Thereafter Dataset is split into 80% for training sets and 20% for validation sets.

2.1.1. Feature extraction

In this stage two different feature types are extracted consequently; the first spatial features are extracted by the pre-trained Densenet-121 on the ImageNet dataset. The end part of it is a global average pooling (GAP) layer applied. Global average pooling reduces the spatial dimensions of the feature maps to a 1x1xN tensor, where N is the number of channels. This operation summarizes the spatial features and creates a compact representation. The total trainable parameters from this step are 7037504 parameters with shapes (V, 4, 4, 1024) where 1024 is the number of features extracted from each frame. Due to violent actions performed by humans occurring throughout the time in the sequence of video frames, a second type of feature set is required.

Thus, a fresh set of frame features is forwarded to the second stage to extract temporal features. In this stage, ConvLSTM is utilized to capture sequential information along the video frames. The LSTM setup uses a 64-unit ConvLSTM2D layer having dimensionality of the output. A Flatten layer is used after ConvLSTM2D.

2.1.2. Classification

In this step Dense layers are used for classification where the first layer has **N** neurons, the second layer has **N** neurons and the last layer contains only 2 for violence and nonviolence classes.

The first and second layers used the Rectified Linear Unit (RELU) activation function as in (1), while the last layer used the soft-max activation function as in (2).

$$f(x) = Max(0, x) \tag{1}$$

$$f(x_j) = \frac{e^i}{j\Sigma e^{xj}} \tag{2}$$

3. Experiments and results

The proposed model is evaluated against three of the state-of-the-art benchmark datasets including hockey fight [n], violent flow [n], and movie [n] datasets. Benchmark of Real-Life Violence Situations (RLVS) is most diverse in terms of environment, and actions, thus it is used for both testing and fine-tuning of the proposed model.

3.1. Dataset preparation

3.1.1. Hockey dataset

The Hockey Dataset [8] was assembled from 1000 National Hockey League games. While 500 of them are non-violent 500 of them contain hockey players fighting. All frames from videos were extracted without pre-processing due to all of them having consistent backgrounds, during experiments 25 frames were extracted skipping every 3rd frame to reduce duplicate frames.

3.1.2. Movie dataset

The Movie Dataset [8] consists of 200 videos divided into 100 violence and 100 non-violence videos. The violent videos were collected from movie scenes, while the non-violence videos were collected from other actions. Unlike the hockey dataset, the movie dataset exhibits diverse backgrounds. During the experiments conducted, 15 frames were selected from each video.

3.1.3. Violent flow dataset

Violent flow [9] comprises 246 videos depicting crowd scenes featuring fights that occurred between individuals. These videos were sourced from violent incidents during football matches. In the performed experiments, 20 frames were utilized as input for our proposed model.

3.1.4. Real life violent situations dataset

Real Life Violence Situations [10] benchmark consists of 2000 videos divided into 1000 violence clips and 1000 nonviolence clips. The violent clips involve fights in many different environments such as streets, prisons, and schools. The non-violence videos contain other human actions such as playing tennis, football, basketball, swimming, and eating. In creating the RLVS dataset, some videos were captured manually, while others were sourced from YouTube to prevent redundancy in persons and environment in the captured videos. Long videos are cut into short-length videos with a maximum duration of 7 seconds, minimum duration of 3 seconds, and average duration of 5 seconds. These segments are considered to have high resolution (480p – 720p) and to include a variety of people in race, age, and gender with different environments.

3.2. Evaluation parameters

The Main metric that was used to determine the best type of LSTM network that works in combination with DenseNet-121 is accuracy and loss. We also will be using recall precision and f-1 score metrics to evaluate the quality of the model's predictions in future work. The proposed method was tested on Hockey, Movie, Violent Flow, Real Life Violent Situations datasets. All models were trained on each dataset using Adam as an optimizer. The learning rate was set to 0.0001. The Networks were trained for 20 epochs.

3.3. Results

Currently, the performed experiments were aimed at determining the type of LSTM network that works best with DenseNet-121.Table 1 shows the accuracy and loss with datasets.

Table 1
LSTM network accuracy

Dataset	Loss	Accuracy
Hockey	0.0917	97.67%
Movie	0.0998	98.23%
Violent Flow	0.1595	91.56%
RLVS	0.3079	90.44%

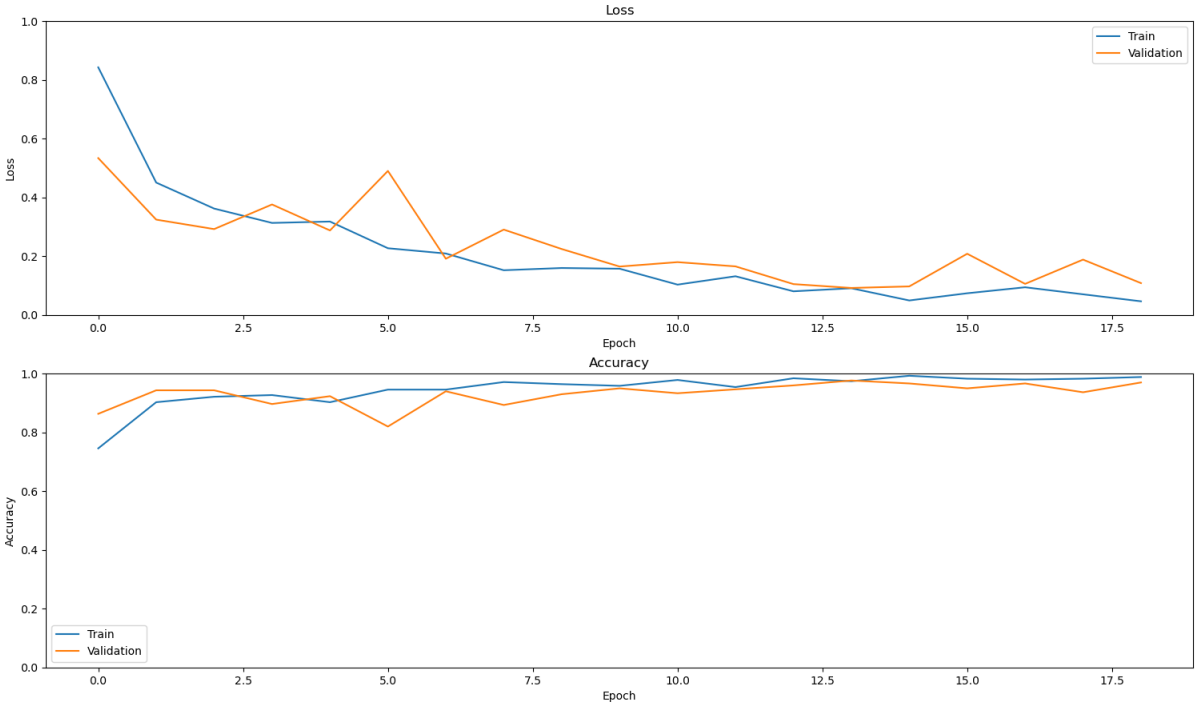

Figure3: Accuracy and Loss during training

Accuracy

Accuracy serves as a metric for assessing the performance of a classification model, and it is usually presented as a percentage. Accuracy quantifies the number of predictions where the model's output matches the actual value, essentially a binary (true/false) outcome for individual samples. Depicted in Figure 3 and tracked throughout the training phase, the value is often associated with the overall or final model accuracy.

$$accuracy = \frac{(TruePositive + TrueNegative)}{(TruePositive + FalsePositive + TrueNegative + FalseNegative)}$$

where TruePositive - is a correctly predicted label, and True Negative - is incorrectly labeled True when it's originally False. FalsePositive - incorrectly labelled False when it was originally True. And FalseNegative - fully incorrectly predicted label

Loss

A loss function sometimes called a cost function, considers the probabilities or the level of uncertainty in a prediction by measuring how much the prediction deviates from the actual value. This provides a more detailed assessment of the model's performance. Unlike accuracy, which is expressed as a percentage, loss is the cumulative measure of errors made for individual samples in training or validation datasets. Loss is typically employed in the training phase to determine the optimal parameter values for the model, such as the weights in a neural network. In the training process, the objective is to minimize this value.

$$loss = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot log_2(y_{pred_i}) + (1 - y_i) \cdot (1 - y_{pred_i})$$

where N is its output size, y_i - sample, y_pred_i - predicted value from the model, minus means for us to minimize function.

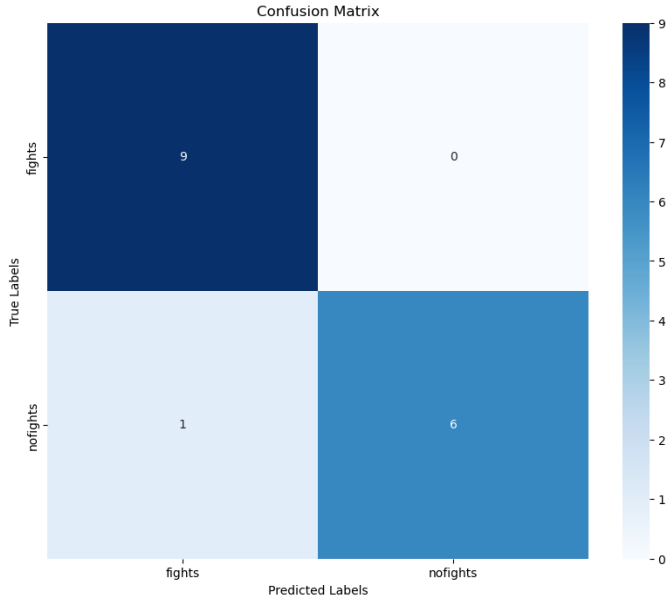

Figure4: Confusion matrix

A widely used evaluation metric for characterizing the performance of a classification model - confusion matrix Figure 4. is typically presented as a table. This table allows for a comparison between predicted and actual values to assess the model's performance.

4. Conclusion and future work

We proposed the DenseNet-LSTM network specifically for violence detection and classification. Because of their reduced feature redundancy and compact internal representations, DenseNets are viable feature extractors for convolutional features. Feature transfer of DenseNets allows them to be used in computer vision tasks. Due to this combining them together with ConvLSTM allowed us to achieve high accuracy on various datasets. For our future work, we will introduce other benchmarks in order to test and analyze more diverse settings and complex actions between humans and potentially dangerous objects.

5. Acknowledgements

This Word template was created by Aleksandr Ometov, TAU, Finland. The template is made available under a Creative Commons License Attribution-ShareAlike 4.0 International (CC BY-SA 4.0).

6. References

[1] Chiba, N., & Hino, K. (2017). CCTV installation in public areas balancing privacy and security. Reports of the City Planning Institute of Japan, 16(2), 124–128. https://doi.org/10.11361/reportscpij.16.2_124.

[2] Hino, K. (2022). Changes in public attitudes toward CCTV installations in residential areas between 2008 and 2019. Cities, 128, 103810. https://doi.org/10.1016/j.cities.2022.103810

[3] Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2016). Densely connected convolutional networks. arXiv (Cornell University). https://doi.org/10.48550/arxiv.1608.06993.

[4] Piza, E. L., Welsh, B. C., Farrington, D. P., & Thomas, A. L. (2019). CCTV surveillance for crime prevention. Criminology & Public Policy, 18(1), 135–159. https://doi.org/10.1111/1745-9133.12419.

[5] Shi, X., Chen, Z., Wang, H., Yeung, D., Wong, W., & Woo, W. (2015). Convolutional LSTM network: a machine learning approach for precipitation nowcasting. arXiv (Cornell University). https://doi.org/10.48550/arxiv.1506.04214.
[6] Teuwen, J., & Moriakov, N. (2020). Convolutional neural networks. In Elsevier eBooks (pp. 481–501). https://doi.org/10.1016/b978-0-12-816176-0.00025-9.
[7] Sutskever, I. (2014, September 10). Sequence to Sequence Learning with Neural Networks. Retrieved from https://arxiv.org/abs/1409.3215.
[8] Gracia, I. S., Deniz, O., García, G. B., & Kim, T. (2015). Fast fight detection. PLOS ONE, 10(4), e0120448. https://doi.org/10.1371/journal.pone.0120448.
[9] Hassner, T., Itcher, Y., & Kliper-Gross, O. (2012). Violent flows: Real-time detection of violent crowd behavior. 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops. https://doi.org/10.1109/cvprw.2012.6239348.
[10] Soliman, M., Kamal, M. H., Nashed, M., Mostafa, Y. M., Chawky, B. S., & Khattab, D. (2019). Violence Recognition from Videos using Deep Learning Techniques. 2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS). https://doi.org/10.1109/icicis46948.2019.9014714.