Journal of Soft Computing and Computer Applications

Volume 1 | Issue 1 Article 1005

2024

Strangeness Detection from Crowded Video Scenes by Hand-Crafted and Deep Learning Features

Ali A. Hussan
Department of Computer Science, University of Technology - Iraq, Baghdad, Iraq,
cs.20.02@grad.uotechnology.edu.iq

Shaimaa H. Shaker
Department of Computer Science, University of Technology - Iraq, Baghdad, Iraq,
120011@uotechnology.edu.iq

Akbas Ezaldeen Ali
Department of Computer Science, University of Technology - Iraq, Baghdad, Iraq,
110034@uotechnology.edu.iq

Recommended Citation

Hussan, Ali A.; Shaker, Shaimaa H.; and Ali, Akbas Ezaldeen (2024) "Strangeness Detection from Crowded Video Scenes by Hand-Crafted and Deep Learning Features," *Journal of Soft Computing and Computer Applications*: Vol. 1: Iss. 1, Article 1005.
DOI: https://doi.org/10.70403/3008-1084.1005

JOURNAL OF SOFT COMPUTING AND COMPUTER APPLICATIONS 2024;1:1005

ORIGINAL STUDY

Strangeness Detection from Crowded Video Scenes by Hand-Crafted and Deep Learning Features

Ali A. Hussan *, Shaimaa H. Shaker, Akbas Ezaldeen Ali

Department of Computer Science, University of Technology – Iraq, Baghdad, Iraq

ABSTRACT

Video anomaly detection is one of the trickiest issues in intelligent video surveillance because of the complexity of real data and the hazy definition of anomalies. Since abnormal occurrences typically seem different from normal events and move differently. The global optical flow was determined with the maximum accuracy and speed using the Farneback approach for calculating the magnitudes. Two approaches have been used in this study to detect strangeness in the video. These approaches are Deep Learning (DL) and manuality. The first method uses the activity map's development of entropy to detect the oddity in the video using a particular threshold. The second method uses a Convolutional Recurrent Auto Encoder (CRAE). CRAE is a network that combines a convolutional autoencoder and an attention-based Convolutional Long-Short-Term Memory (ConvLSTM) network. The irregularity regarding the temporal pattern and the spatial irregularity, respectively, might be captured by the convolutional autoencoder and ConvLSTM network. The current output properties of each CovnLSTM layer were extracted from their hidden states using the attention method. Comparing the error with an experimentally established threshold, anomalies were specified to exist and a convolutional decoder was used to recreate the input video clip and the testing video clip. The best detection of whether in-frame variation was abnormal or normal, a trial-and-error threshold was 0.04 for handcrafted features through the University of Minnesota (UMN) dataset and 0.00035 for DL features through the avanue dataset.

Keywords: Strangeness detection, Hand-crafted, Optical flow, Deep learning, Autoencoder

1. Introduction

The value of camera video streams is equivalent to that of other significant sources, including sensor data, social media data, medical data, security data, and cutting-edge data from space research [1]. With the exponential growth of video data, there is an increasing need to learn and identify rare, unusual, and interesting events.

Received 15 March 2024; accepted 10 June 2024.
Available online 27 June 2024

* Corresponding author.
E-mail addresses: cs.20.02@grad.uotechnology.edu.iq (A. A. Hussan), 120011@uotechnology.edu.iq (S. H. Shaker), 110034@uotechnology.edu.iq (A. Ezaldeen Ali).

https://doi.org/10.70403/3008-1084.1005

Fig. 1. Examples of video distortions. The red part of each image represents abnormal objects [6].

Applications that rely on visual observation must be able to detect these anomalies in videos. They frequently have a very slim likelihood of occurrence [2]. These anomalies must be manually detected, and it is a very delicate task that calls for 10 more people than are typically available. As a result, quick, automated, and precise detection is now required. So, state-of-the-art technology necessitates a considerable amount of configuration work on each video stream before deploying the video analysis process. Additionally, it is challenging to generalize the detection model of other monitoring scenes [3]. Due to its great noise, dimension, and variety of interactions and events, video data presents a paradigm and a challenge. Running in a park could be normal, yet running in a restaurant might not be [4]. Due to such difficulties, Machine Learning (ML) techniques must identify video patterns that produce anomalies in practical applications. When security is a top priority, installing a camera must be installed anywhere [5]. This is illustrated in Fig. 1.

It appears tedious and time-consuming to manually monitor. In many situations, like the detection of violence, the identification of theft, the likelihood of an explosion, etc., security could be characterized in several ways. Security refers to practically all abnormal situations in crowded public areas [7]. Because it includes a group activity, violence is tough to handle; due to the limits imposed by the real world, it can be exceedingly challenging to analyze unusual or abnormal activity in crowd video scenes [8]. Computers will eventually be able to think like humans thanks to Artificial Intelligence (AI). By including learning and training components, ML makes it far more even [9]. The idea of DL, which automatically extracts features or the variables of difference that separate things from one another, is made possible by the availability of large datasets and high-performance computers. Video surveillance data is one of the many data sources that contribute to terabytes of big data [10]. Widespread surveillance data is available via cameras in industrial and residential regions, commercial enterprises, and educational institutions. Cameras are placed in centers, public transportation, and places of worship, contributing to public data cities and private data, respectively [11]. This is illustrated in Fig. 2.

Fig. 2. The scooter is going in reverse [12].

The main contribution of this study is to propose an algorithm that can detect strangeness from crowded video scenes by using hand-crafted and DL techniques to extract features. The outline of this study is as follows: Section 2 presents the hand-crafted features. Section 3 explains the DL techniques. The proposed oddity detection algorithm is present in Section 4. Section 5 illustrates the experimental results. Finally, the conclusion and future work is presented in Section 6.

2. Hand-crafted features

Learning techniques were utilized in the study of video anomaly detection to learn the anomaly detection model while using man-made features. Man-made features were used to represent the motion and appearance characteristics of people. These techniques fall into two categories: anomaly detection approaches depending on paths and anomaly detection approaches depending on cubes, and whether the object detection and tracking approach is used. Every motion trajectory serves as an image coordinate sequence for the target [13].

2.1. Optical flow

The relationship between temporal changes and spatial properties in images, or motion, is an essential feature of frame sequences. It demonstrates dynamics tire. Optical flow estimation means displaying motion data from an image sequence. Optical flow is a 2D animated map that displays the 3D motion of a scene on the image plane. Remember that if the camera position parameters are given, the optical flow module can also be used to extract parallax in fixed stereovision [14]. The optical flow will work when the pixel intensities do not fluctuate with time, the nearby pixels move similarly, the motion is locally smooth, and the visible gradients are static. This is illustrated in Equation (1) and Fig. 3.

$$U.I_x + V.I_y = I_t \tag{1}$$

where V and U represent the optic flow components in vertical and horizontal directions. I_x, I_y, and I_t are brightness function derivatives from the x and y image coordinates at time (t) [15].

Since the image pixels and intensity are the same from one frame to the next, Equation (2) states that the pixel displacement is (dx, dy):

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t) \tag{2}$$

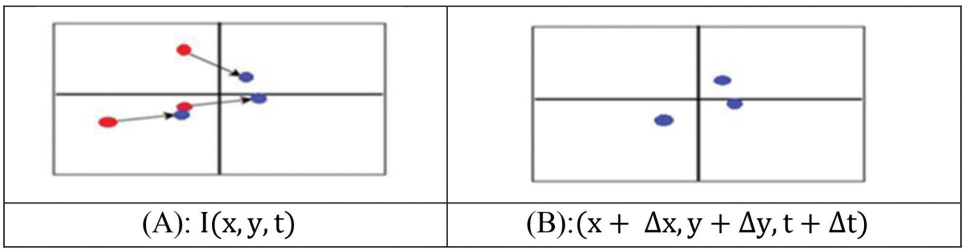

Fig. 3. Pixel offset in two consecutive images. (A) and (B): Blue pixels and red pixels correspond to the image at time dt and t, respectively [15], where I (x, y, t) is a pixel in the first frame, which advances to the following frame, taking dt time.

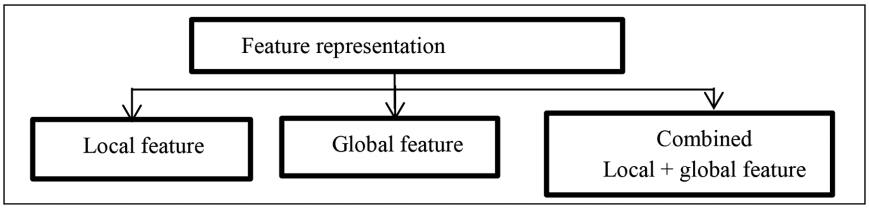

Fig. 4. Categorization of feature representation.

where I (x, y, t) is the assumed pixel at locations (x, y, t), Δx, Δy, Δt stands for movement between the two frames, and c represents a real-valued constant number [16].

2.2. Global and local features

Local and global optical flow approaches are distinguished from one another. While global/dense approaches process every pixel in the image, local approaches only require processing a small portion of the total number of pixels on the image. For additional flow data to be available in sparse/global flow extraction techniques. The global features are those where each image is captured by a single feature vector that contains data from the entire image. The components of the image, such as specific objects or regions, receive no consideration. Once the features of each image have been calculated, a distance metric might be used to determine how comparable any two images [17]. This is illustrated in Fig. 4.

2.3. Farneback method

The Farneback technique is a two-frame motion estimation algorithm that utilizes polynomial expansion to approximate the neighborhood of each image pixel. This is illustrated in Fig. 5 [15].

Each level of the image pyramid created by the Farneback algorithm has a lesser resolution than the level before. The algorithm may follow points at multiple resolution levels, starting at the lowest level when a pyramid level is higher than 1. The algorithm can manage bigger point displacements between frames by expanding several pyramid levels and this will be done by more calculations [18]. This is illustrated in Fig. 6.

With each one of the levels, the algorithm improves the tracking in this way. The algorithm might manage large pixel motions that could span distances bigger than the size of the neighborhood [19].

3. Deep Learning

Researchers have started to investigate the detection of abnormal crowd behavior depending on DL due to the active development of AI, which has produced several results [20]. DL algorithms, in contrast to craft-based approaches, concentrate on extracting high-level aspects of the movement and appearance of pedestrians in video. DL can differentiate between abnormal and normal behavior [21], offering a 3D convolution network-based spatiotemporal autoencoder. The decoder reconstructs the frames after the encoder takes the temporal and spatial information [20]. By calculating, the reconstruction will be lost using the Euclidean distance between the reconstructed and original batch, and the abnormal events are found [22, 23].

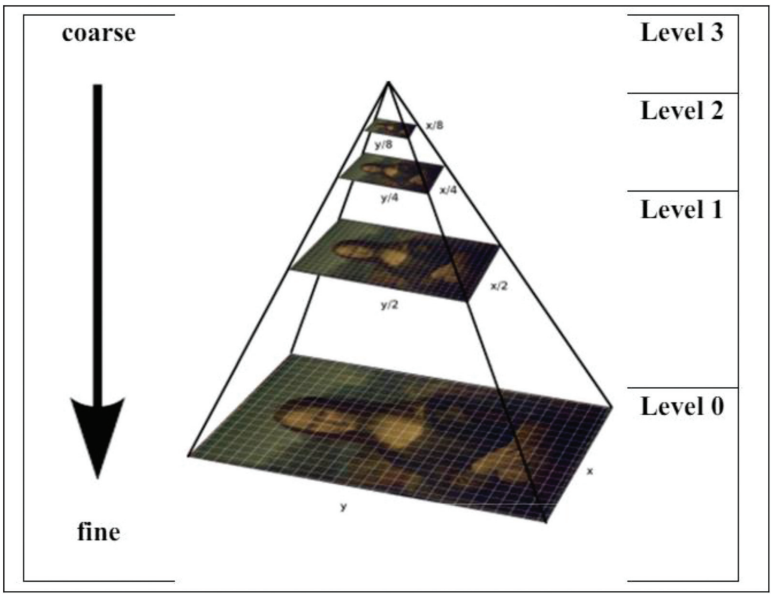

Fig. 5. Image pyramid with four levels. At each level, the image is downsized. Optical flow computation starts at the top of the pyramid (level 4) and ends at the bottom (level 0) [15].

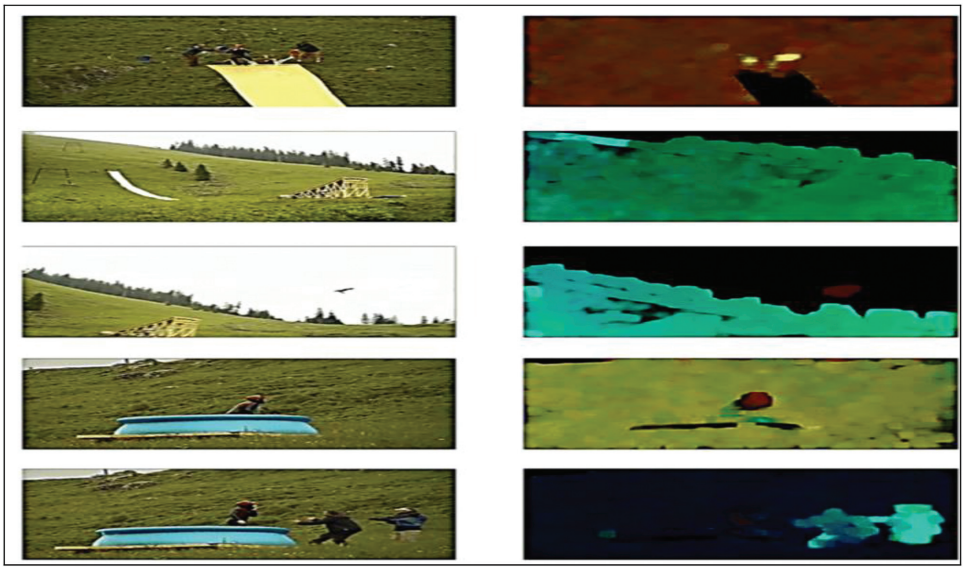

Fig. 6. Demonstration of the behavior of dense optical flow (utilizing the Farneback approach) [18].

Deep Neural Networks (DNNs) are used for automatic video learning, video representing, and feature extracting from both temporal and spatial dimensions. This is done through performing 3D convolutions. As the name implies, autoencoders have two stages: encoding and decoding. To reduce the dimensionality, there is a need to set the number of encoder output units lower than the input [24]. The Back Propagation (BP) method is typically used to train models in an unsupervised manner, reducing the reconstruction error of

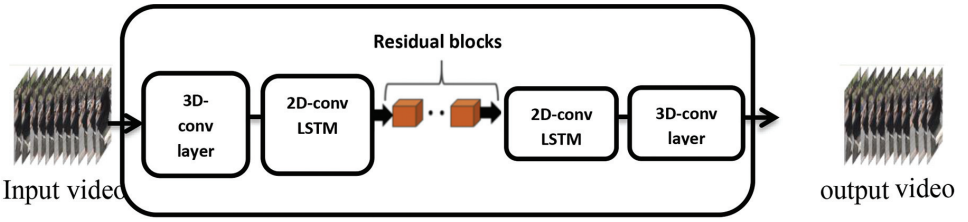

Fig. 7. Training autoencoder for video Strangeness detection [28].

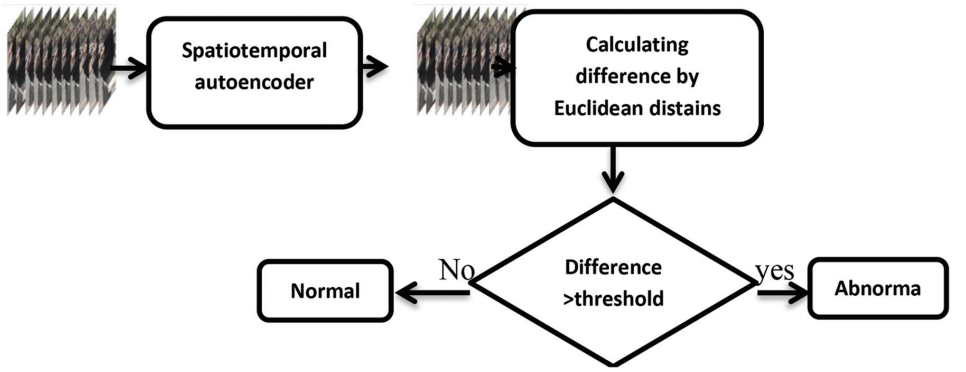

Fig. 8. Testing phase for video Strangeness detection [28].

the decoding outputs from the original inputs. A nonlinear activation function allows an autoencoder to extract more beneficial features [25].

The encoder extracts temporal and spatial information by extracting frames from the given input video. While the decoder reconstructs the frames again into the normal form and reconstructs the video [26].

On normal videos, the autoencoder was trained. Discovering abnormal events will depend on the custom video that feeds Euclidean distance and the frames predicted by the autoencoder. The main function of the convolutional network is to extract features from the input image [27]. Figs. 7 and 8 illustrate this.

Through learning image attributes from small squares of input data, convolution preserves the spatial relation between pixels. Mathematically, the convolution process produces dot products between the network design, as suggested in Fig. 2. It accepts an input sequence of length T and produces an output sequence. This output sequence is a reconstruction of the input sequence. The number on the rightmost side indicates the size of the output layer. After processing $T = 10$ frames, the spatial encoder takes one frame at a time as input. The encoded features from $T = 10$ frames are concatenated and sent into the temporal encoder for motion encoding. The decoders mirror the encoders to reassemble the video volume [29].

4. Proposed oddity detection algorithms

This section presents the hand-crafted feature and DL-based anomaly detection method.

Algorithm 1. Strangeness detection through hand-crafted feature.

```
Input: RGB video
Output: give ALARM on video when Strangeness detection
Begin
1 Repeat
    2 While (video frames not terminated) do
        3 Convert video into sequential frames
        4 Convert RGB frames into gray
        5 Apply Gaussian filter on each frame.
        6 Estimate activity map (Optical Flow)
        7 Apply to post-preparing (Median Filter)
        8 Calculate the difference between two frames
        9 Estimate entropy from the difference
        10 If the entropy level > threshold
            give ALARM on video (abnormal)
        11 else
            give normal video
        12 End while
13 Repeat this until frames are end
14 End
```

4.1. Hand-crafted feature

Optical flow and entropy level approaches have been used to extract global features. These approaches have been used because a video global descriptor can be defined as a set of features describing a video as a whole and, thus, is best capable of describing the normal patches of the video. Algorithm 1 shows how the hand-craft features method can be used to detect strangeness.

At first, the video must be pre-processed by converting it to gray. This is done because the important thing in the proposed algorithm is the movement of the object, not it is color density. After that, the video is converted into several sequential frames through resolution. Then, a Gaussian filter is applied for noise reduction. The video stream is broken up into frames in the pre-processing module, and the dynamic and static features were extracted from each frame. For each old and new frame, the changing percentage was calculated by calculating the size of the optical flow. Then, the entropy of the frames is measured, and through the experiment, a certain threshold was determined for each video to show the abnormal conditions in the video. If the abnormal movement has occurred in the video, a printing sign or warning informs is displayed and prints a graph of the entropy concerning the time of occurring abnormal movement as shown in Fig. 9.

4.2. Deep-learning-based Strangeness detection method

Two mechanisms have been used for training and testing by using the 3D convolution autoencoder and threshold, as shown by Algorithm 2 and 3.

Set up a function to process and save video frames, then describe the directory path variable. Call the store method and extract the frames from the video. It converts the input image into arrays of numeric data for computer processing. Those frames are retrieved and processed individually, and the image list should be saved in a numpy file. To produce a saved model, the spatial autoencoder was designed to analyze the encoder model using the stored frames. This is then utilized to analyze the abnormal events, and all instructional films solely include instances of typical occurrences. Both abnormal and normal events can be shown in testing recordings. Table 1 shows the architecture of the proposed model.

Algorithm 2. Strangeness detection through deep learning (training phase).

Input: RGB video
Output: training model
Begin
1 Repeat
    2 While (video frames not terminated) do
        3 Convert video into Sequential frames
        4 Resize RGB frames into 277 × 277 × 10
        5 Convert RGB frames into gray
        6 Normalization: Divide the mean values on the standard deviation for each image
        7 End while
        8 Create spatial autoencoder architecture
        9 Model training (using training video)
        10 Save training model
11 Until training video frames terminated
12 End

Algorithm 3. Strangeness detection through deep learning (testing phase).

Input: RGB test video, The trained model, threshold (trial and error)
Output: give ALARM on video when Strangeness detection
Begin
1 Repeat
    2 While (video frames not terminated) do
        3 Convert video into Sequential frames
        4 Resize RGB frames into 277 × 277 × 10
        5 Convert RGB frames into gray
        6 Normalization: Divide the mean values on the standard deviation for each image
    7 End while
    8 Production test frame using the trained model
    9 Calculate the value of the loss before and after testing the model in the same frames by using the Euclidean distance:
    $d = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}$ where x1, y1, x2, y2 are coordinates of two points
    10 If the loss > threshold
    give ALARM on frames (abnormal) and save it
11 Until testing video frames terminated
12 End

Table 1. Architecture of the model proposed.

Layer (type)	Kernel size	Strides	Activation	Output shape
Input	–	–	–	10 × 277 × 277
(Conv3D) 1	(11,11,1)	(4,4,1)	tanh	(None, 55, 55, 10, 128)
(Conv3D) 2	(5,5,1)	(2,2,1)	tanh	(None, 26, 26, 10, 64)
(ConvLSTM2D) 1	(3,3)	1	return_sequences = True	(None, 26, 26, 10, 64)
(ConvLSTM2D) 2	(3,3)	1	return_sequences = True	(None, 26, 26, 10, 32)
(ConvLSTM2D) 1	(3,3)	1	return_sequences = True	(None, 26, 26, 10, 64)
(Conv3DTranspose) 1	(5,5,1)	(2,2,1)	tanh	(None, 55, 55, 10, 128)
(Conv3DTranspose) 2	(11,11,1)	(4,4,1)	tanh	(None, 227, 227, 10, 1)
Output	–	–	–	10 × 277 × 277

where Conv is a convolutional layer, LSTM is a ConvLSTM layer, Transpose is a deconvolutional layer, Output is the output layer. Conv1, Conv2, Transpose1, and Transpose2 make up the encoder and decoder, respectively.

Make a test for a different video and then check the outcomes of abnormal event detection on any custom video. The abnormal events are detected depending on the Euclidean distance of the custom video feed and the frames predicted by the autoencoder. The threshold controls how responsive the detection system should be. For instance, setting

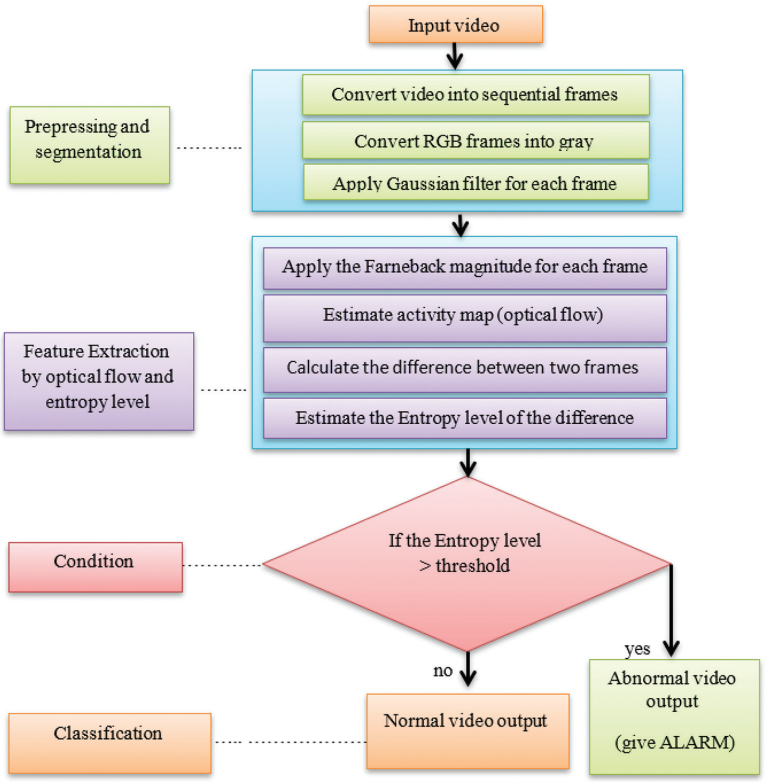

Fig. 9. Strangeness behavior detection algorithm based on global optical flow and entropy level.

a low threshold causes the system to become responsive to events in the scene, resulting in more alarms being triggered, as shown in Fig. 10.

5. Experimental results

The hand-crafted feature utilized a University of Minnesota (UMN) dataset to find unusual crowd detection. UMN dataset includes three scenes—the first is outdoor, the second is indoors, and the third is outdoors—each video at a frame rate of 30. (320 × 240 × 3) is the RGB frame size (see Fig. 6). The dataset contains the raw data needed to classify abnormalities, as shown in Fig. 11.

The Avenue dataset comprises 21 testing videos (15,324 frames) and 16 training videos (15,328 frames) for the DL phase. The training set of the dataset contains a small number of anomalies. Additionally, normal situations are scarcely depicted in the training videos. Each video frame has a resolution of 360 by 640 pixels. Ground-truth locations of anomalies are reported for each test frame using of pixel-level masks. Both anomalous and normal events can be found in the test set. Anomalous events discovered within the test frames are shown in Fig. 12.

5.1. Hand-crafted feature

When the proposed algorithm was applied to the three videos of the dataset, and they were crowded and external, it was noticed after taking pictures of the results before and

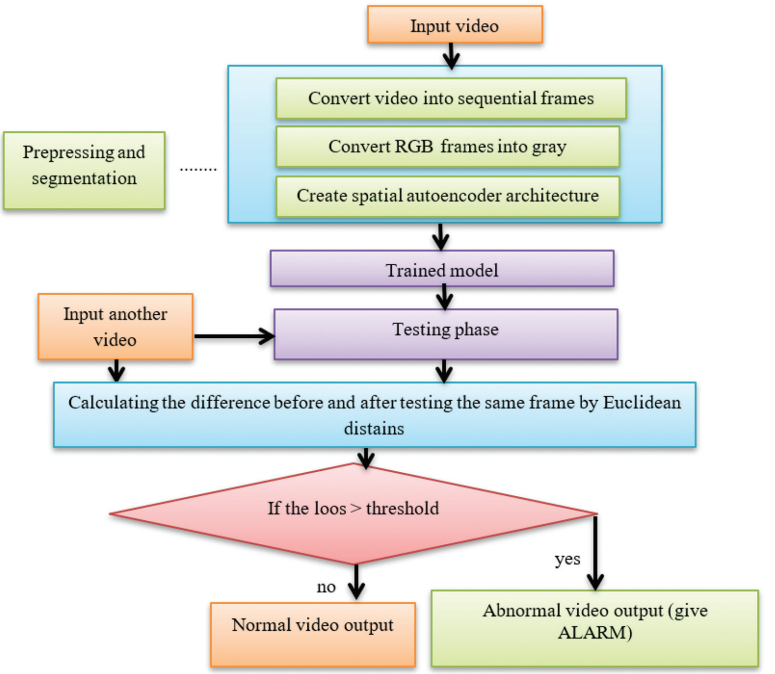

Fig. 10. Strangeness behavior detection algorithm based on deep learning.

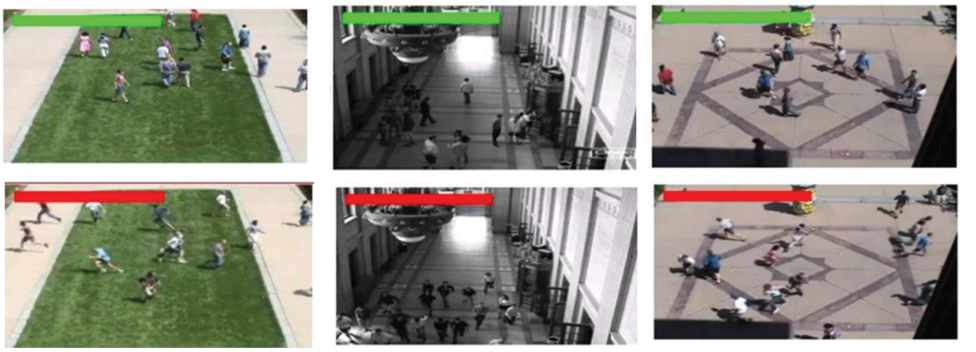

Fig. 11. UMN data-set samples for each of 3 scenes: Normal (green) and Abnormal (red).

after the occurrence of strangeness in the video and the sudden movement of the crowd, the big difference in the visual flow when the activity map of the people in those videos changes, and calculating the difference between the previous and subsequent activity map in the frame sequence, after producing the optical flow, it utilizing Farneback technique, an activity map has been created with the use of multiple frames for showing the continuity regarding the flow over time. Then, the activity map was utilized to generate the entropy and show the max entropy level, and anomaly detection in the video was identified. ALARM was given, as shown in Figs. 13 and 14.

The result shows that the maximum entropy level for outdoor video is more than 0.45. However, the maximum entropy level for indoor video is more than 0.16. with a threshold of 0.04) which was obtained experimentally (for classifying each frame as abnormal or normal in a given sequence, as shown in Table 2.

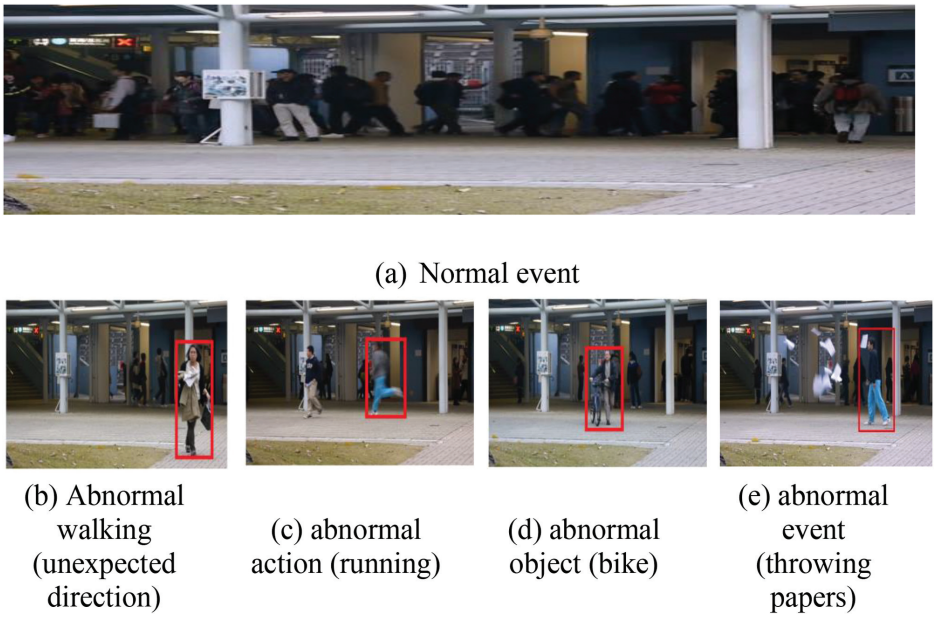

Fig. 12. Sample images from the Avenue dataset showing normal and anomalous events.

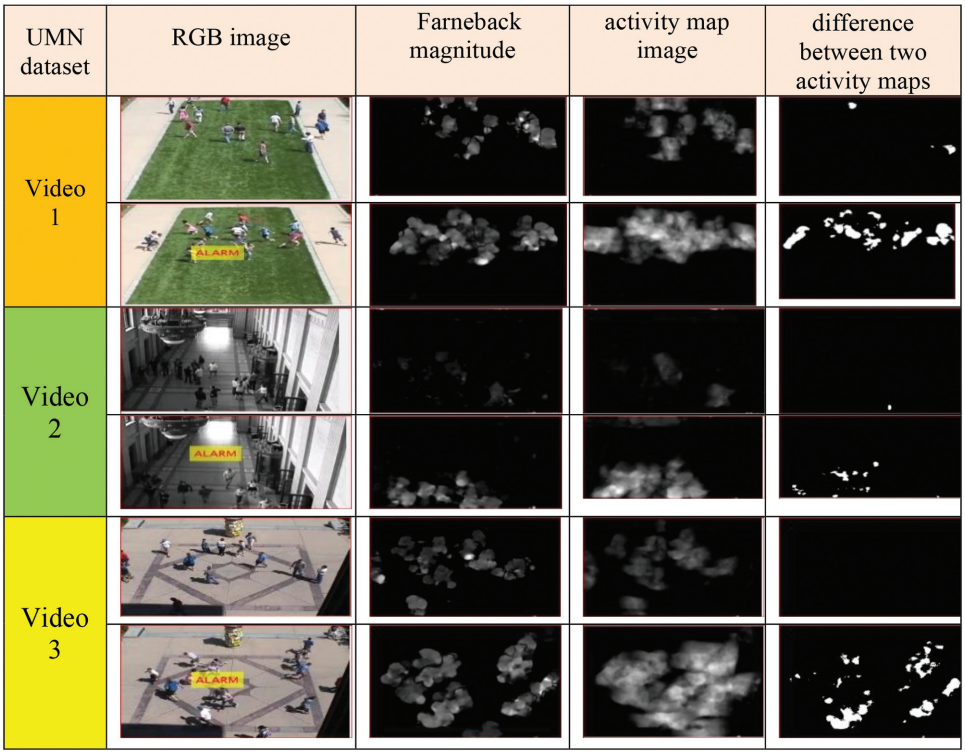

Fig. 13. Indoor and outdoor Localization of strangeness from Crowded Video Scenes based on optical flow.

Table 2. Shows the entropy level and threshold for indoor and outdoor videos.

UMN dataset video	Number of frames	Entropy threshold	Max entropy level
Outdoor video1	614	0.04	0.58
Indoor video2	575	0.04	0.165
Outdoor video3	638	0.04	0.48

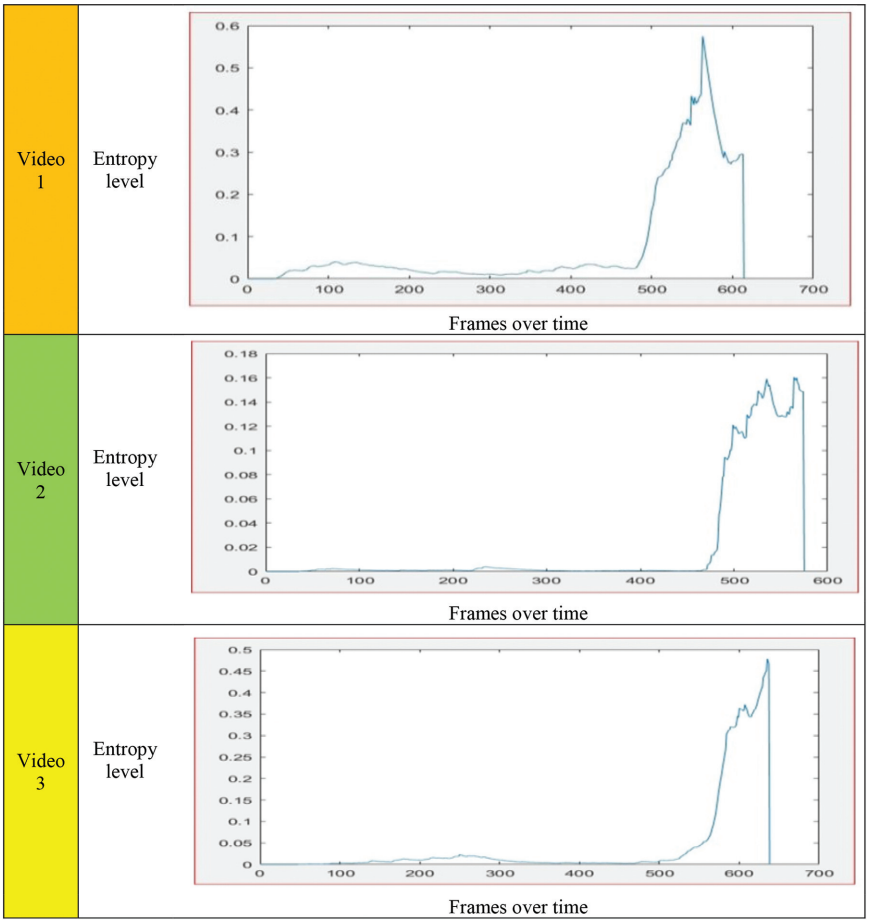

Fig. 14. Entropy level of the indoor and outdoor video.

5.2. Deep Learning

The visual results demonstrate the effectiveness of the suggested method in detecting anomalies present at the frame level (a global anomaly in the UMN dataset) and pixel level (a local anomaly in the Avenue dataset).

5.2.1. Training phase

After preprocessing the training videos where the videos, is converted into frames and each frame into an array and resized is converted into a gray type, image normalization is made for the pixel values to be between 0 and 1. After that, the videos were trained using the proposed model, where five training videos were used from Avenue dataset with

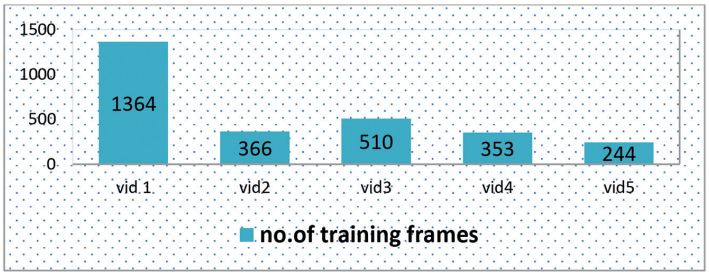

Fig. 15. The number of frames for five training videos used.

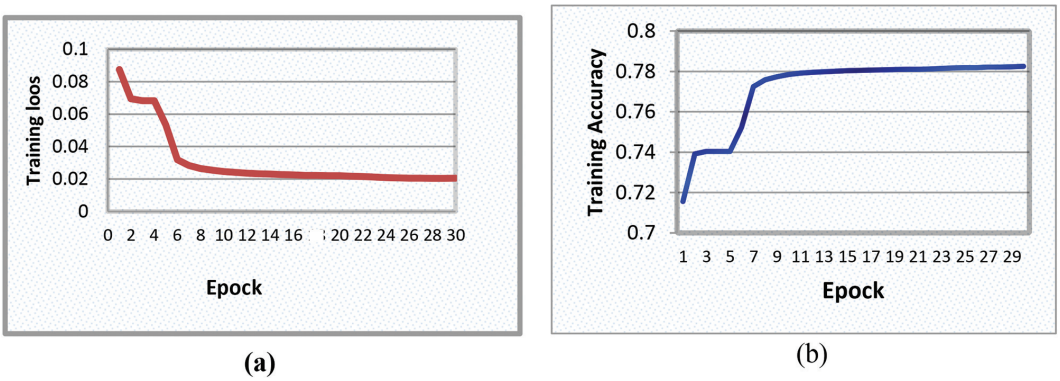

Fig. 16. (a) and (b) loos and accuracy values after training the Avenue dataset videos for each epoch.

the size of 2837 frames shown in Fig. 15 and through the use of a threshold equal to 0.00035, which was obtained experimentally by trial and error, loss and accuracy results were obtained for each iteration, as shown in Fig. 16.

5.2.2. Testing phase

After completing model training, it will be stored and then predicted and tested using 12 test videos (6010 frames) from Avenue dataset, as shown in Fig. 17. The same preprocessing operations are performed on the test videos. After that, the testing phase will be made by using the trained model. The results were obtained for abnormal movements in the test videos marked by red text (abnormal event) of each detected frame, as shown in Fig. 18.

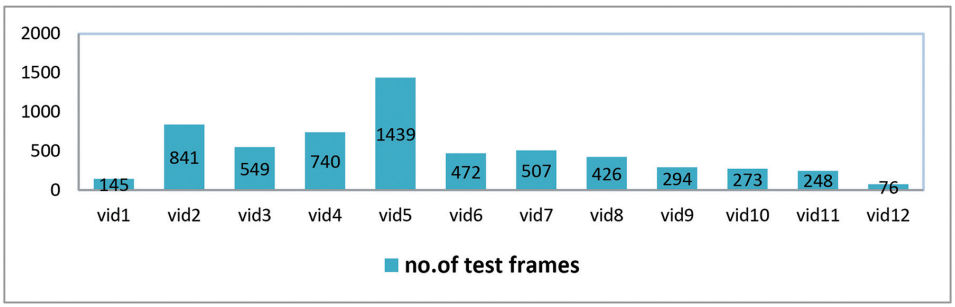

Fig. 17. The number of frames for 12 test videos used.

Fig. 18. The sample of anomalous detection results of the proposed approach marked by red text (abnormal event).

6. Conclusions

This study provided an overview of the most recent technologies for recognizing human actions. Modeling the crowded video scene and identifying abnormal movements should be done quickly and accurately. This is accomplished by supplying global and local features. Global and local features are powerful and effective. Anything that deviates from these features of natural movements is regarded as abnormal movement. To improve the recognition of human actions from 3D sensor data, anomalies in the object should be identified through tracking, recording movements, extracting features, and ensuring that these features are strong and separate the object from other objects. This is done by using one of the hand-crafted and DL approaches. The result shows that the maximum entropy level for indoor video was less than 0.16, and for outdoor video, it was greater than 0.45 in the hand-crafted feature method. In the DL method, the largest loss (difference) obtained during the training phase was 0.0875, and the largest accuracy (similarity) was 0.7825. This represents the difference and similarity between the frames before and after training. Experiments on two datasets proved that the features learned in the proposed approach were very effective for the detection task.

References

1. S. A. Jebur, K. A. Hussein, H. K. Hoomod, L. Alzubaidi, and J. Santamaría, "Review on deep learning approaches for anomaly event detection in video surveillance," *Electron.*, vol. 12, no. 1, Art. no. 29, 2022, doi: 10.3da0/electronics12010029.

2. Y. Hao, Y. Liu, J. Fan, and Z. Xu, "Group abnormal behaviour detection algorithm based on global optical flow," *Complexity*, vol. 2021, Art. no. 5543204, 2021, doi: https://doi.org/10.1155/2021/5543204
3. R. Vrskova, R. Hudec, P. Kamencay, and P. Sykora, "Human activity classification using the 3DCNN architecture," *Appl. Sci.*, vol. 12, no. 2, Art. no. 931, 2022, doi: 10.3390/app12020931.
4. S. Mei, J. Ji, Y. Geng, Z. Zhang, X. Li, and Q. Du, "Unsupervised spatial-spectral feature learning by 3D convolutional autoencoder for hyperspectral classification," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 57, no. 9, pp. 6808–6820, 2019, doi: 10.1109/TGRS.2019.2908756.
5. T. Gangopadhyay *et al.*, "3D convolutional selective autoencoder for instability detection in combustion systems," *Energy and AI*, vol. 4, Art. no. 100067, 2021, doi: 10.1016/j.egyai.2021.100067.
6. W. Shin, S. J. Bu, and S. B. Cho, "3D-convolutional neural network with generative adversarial network and autoencoder for robust anomaly detection in video surveillance," *Int. J. Neural Syst.*, vol. 30, no. 6, Art. no. 2050034, 2020, doi: 10.1142/S0129065720500343.
7. Y. Li, Y. Cai, J. Liu, S. Lang, and X. Zhang, "Spatio-temporal unity networking for video anomaly detection," *IEEE Access*, vol. 7, pp. 172425–172432, 2019, doi: 10.1109/ACCESS.2019.2954540.
8. V. Kumar, V. Tripathi, and B. Pant, "Learning unsupervised visual representations using 3D convolutional autoencoder with temporal contrastive modeling for video retrieval," *Int. J. Math. Eng. Manag. Sci.*, vol. 7, no. 2, pp. 272–287, 2022, doi: 10.33889/ijmems.2022.7.2.018.
9. A. A. Karim and R. A. Sameer, "Static and dynamic video summarization," *Iraqi J. Sci.*, vol. 60, no. 7, pp. 1627–1638, 2019, doi: 10.24996/ijs.2019.60.7.23.
10. A. A. Karim, "Construction of a robust background model for moving object detection in video sequence," *Iraqi J. Sci.*, vol. 59, no. 2B, pp. 969–979, 2018, doi: 10.24996/IJS.2018.59.2B.19.
11. H. I. Abdulrazzaq and N. F. Hassan, "Modified siamese convolutional neural network for fusion multimodal biometrics at feature level," *2019 2nd Scientific Conference of Computer Sciences (SCCS)*, Baghdad, Iraq, pp. 12–17, 2019, doi: 10.1109/SCCS.2019.8852593.
12. X. Zhang, S. Yang, X. Zhang, W. Zhang, and J. Zhang, "Anomaly detection and localization in crowded scenes by motion-field shape description and similarity-based statistical learning," 2018, arXiv1805: 10620.
13. L. Gionfrida, W. M. R. Rusli, A. E. Kedgley, and A. A. Bharath, "A 3DCNN-LSTM multi-class temporal segmentation for hand gesture recognition," *Electron.*, vol. 11, no. 15, Art. no. 2427, 2022, doi: 10.3390/electronics11152427.
14. P. A. S. Mendes, M. Mendes, A. P. Coimbra, and M. M. Crisóstomo, "Movement detection and moving object distinction based on optical flow," in *Transactions on Engineering Technologies*, S. I. Ao, L. Gelman, H. K. Kim, Eds (London, UK), pp. 143–158, 2019.
15. S. Husseini, "A survey of optical flow techniques for object tracking." Accessed Sep. 15, 2023. [Online]. Available: http://efaidnbmnnnibpcajpcglclefindmkaj/https://core.ac.uk/download/pdf/196557054.pdf.
16. H. L. Masoner and A. Hajnal, "Does optic flow provide information about actions?," *Attention, Perception, Psychophys.*, vol. 85, no. 4, pp. 1287–1303, 2023, doi: 10.3758/s13414-023-02674-9.
17. Y. He, Y. Zhong, L. Wang, and J. Dang, "GLFormer: global and local context aggregation network for temporal action detection," *Appl. Sci.*, vol. 12, no. 17, Art. no. 8557, 2022, doi: 10.3390/app12178557.
18. P. Gunawardena *et al.*, "Real-time automated video highlight generation with dual-stream hierarchical growing self-organizing maps," *J. Real-Time Image Process.*, vol. 18, pp. 1457–1475, 2021, doi: 10.1007/s11554-020-00957-0.
19. S. S. Mahmood and L. J. Saud, "An efficient approach for detecting and classifying moving vehicles in a video-based monitoring system," *Eng. Technol. J.*, vol. 38, no. 6, pp. 832–845, 2020. https://www.iasj.net/iasj/download/c168f4362b7cfd4b
20. B. R. Kiran, D. M. Thomas, and R. Parakkal, "An overview of deep learning-based methods for unsupervised and semi-supervised anomaly detection in videos," *J. Imaging*, vol. 4, no. 2, Art. no. 36, 2018, doi: 10.3390/jimaging4020036.
21. B. Wang and C. Yang, "Video anomaly detection based on convolutional recurrent autoencoder," *Sensors*, vol. 22, no. 12, Art. no. 4647, 2022, doi: 10.3390/s22124647.
22. R. Vyškovský, D. Schwarz, V. Churová, and T. Kašpárek, "Structural MRI-based schizophrenia classification using autoencoders and 3D convolutional neural networks in combination with various pre-processing techniques," *Brain Sci.*, vol. 12, no. 5, Art. no. 615, 2022, doi: 10.3390/brainsci12050615.
23. S. Kay *et al.*, "Integrating autoencoder and heteroscedastic noise neural networks for the batch process soft-sensor design," *Industrial & Engineering Chemistry Research*, vol. 61, no. 36, pp. 13559–13569, 2022. doi: 10.1021/acs.iecr.2c01789.
24. G. Dong, G. Liao, H. Liu, and G. Kuang, "A review of the autoencoder and its variants: a comparative perspective from target recognition in synthetic-aperture radar images," *IEEE Geoscience and Remote Sensing Magazine*, vol. 6, no. 3, pp. 44–68, 2018. doi: 10.1109/MGRS.2018.2853555.

25. G. Zhu, L. Zhang, P. Shen, J. Song, S. A. A. Shah, and M. Bennamoun, "Continuous gesture segmentation and recognition using 3DCNN and convolutional LSTM," *IEEE Trans. Multimed.*, vol. 21, no. 4, pp. 1011–1021, 2018, doi: 10.1109/TMM.2018.2869278.
26. S. Abbasi, M. Famouri, M. J. Shafiee, and A. Wong, "Outliernets: highly compact deep autoencoder network architectures for on-device acoustic anomaly detection," *Sensors*, vol. 21, no. 14, Art. no. 4805, 2021, doi: 10.3390/s21144805.
27. M. Yan, J. Meng, C. Zhou, Z. Tu, Y.-P. Tan, and J. Yuan, "Detecting spatiotemporal irregularities in videos via a 3D convolutional autoencoder," *Journal of Visual Communication and Image Representation*, vol. 67, Art. no. 102747, 2020, doi: 10.1016/j.jvcir.2019.102747.
28. K. Deepak, S. Chandrakala, and C. K. Mohan, "Residual spatiotemporal autoencoder for unsupervised video anomaly detection," *Signal, Image and Video Processing*, vol. 15, no. 1, pp. 215–222, 2021, doi: 10.1007/s11760-020-01740-1.
29. E. Kalinicheva, J. Sublime, and M. Trocan, "Unsupervised satellite image time series clustering using object-based approaches and 3D convolutional autoencoder," *Remote Sens.*, vol. 12, no. 11, Art. no. 1816, 2020, doi: 10.3390/rs12111816.