IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

ENHANCING VIDEO ANOMALY DETECTION THROUGH UNSUPERVISED LEARNING

Dr. Puttegowda D, Abubaker Siddiq, Bindushree R D, Chandan D V, Deepika H R

Professor, 4AD20CS003, 4AD20CS018, 4AD20CS019, 4AD20CS022

Department of Computer Science and Engineering,

ATME College of Engineering, Mysore, India

Abstract: We use an algorithm for unsupervised detection of anomalous regions in video sequences, termed "KMEANS," which identifies coherent spatial regions and time intervals indicative of anomalies. In contrast to existing techniques focusing on isolated anomalous data points, our approach enables ranking regions of varying sizes and ensures scalability for large-scale datasets through an interval proposal technique. Experimental validation on synthetic and real data spanning domains such as climate analysis, video surveillance, and text forensics underscores the broad applicability and utility of our method in uncovering noteworthy events across diverse data types.

***Keywords* -** Anomaly detection, video sequences, KMeans, unsupervised learning, spatial coherence, time intervals, scalability, synthetic data, real-world applications, event detection.

I. INTRODUCTION

Numerous pattern recognition methodologies aim to construct models from intricate and noisy data, capturing the typical behavior of observed systems which may be challenging to manually delineate and often remain indirectly observable through data reflections. These models facilitate inference about system properties, forecasting unseen data, and evaluating the "normality" of new observations. However, deviations from expected behavior within the data can disrupt model accuracy, prompting the development of techniques to either exclude abnormal observations prior to model learning or to ensure robust learning unaffected by outliers. While anomalies are commonly perceived as undesirable, attributed to random noise or measurement errors, they may also stem from rare events or intricate processes. Embracing anomalies within the data and exploring the insights they harbor can lead to a deeper comprehension of the analyzed system, revealing inadequacies or obsolescence in existing models, as exemplified by Gilbert Walker's discovery of the correlation between the El Niño weather phenomenon and extreme surface pressures over the equator through analysis of climate data extremes. Consequently, anomaly detection serves not only as a means of outlier removal in preprocessing but also as a vital task in its own right, given that deviations from normal behavior often constitute the primary focus in many applications. Beyond knowledge discovery, instances such as fraud detection (e.g., credit card fraud, identity theft), cyber-security intrusion detection, industrial process fault detection, healthcare anomaly detection (e.g., patient monitoring, disease outbreak detection), and early environmental disaster detection underscore the significance of automated anomaly detection methods, particularly in the face of vast data volumes exceeding human analytical capabilities.

II. LITERATURE SURVEY

Video anomaly detection stands as a cornerstone in computer vision, playing an indispensable role in surveillance, security, and anomaly monitoring applications. The field has witnessed substantial progress in methodologies and techniques aimed at bolstering the accuracy and resilience of anomaly detection systems. Early approaches, as outlined in [1], predominantly relied on basic background subtraction, which proved effective for identifying static anomalies in controlled settings. However, these methods faltered when confronted with dynamic backgrounds, fluctuating lighting conditions, and complex scenarios. A significant breakthrough in video anomaly detection was ushered in by Cheng et al.[3], who introduced a novel hierarchical feature representation approach. This innovative methodology underscored the significance of capturing both local and global anomalies in video data. By leveraging Gaussian Process Regression to model object interactions within scenes, their method offered a more precise portrayal of normal behavior. Notably, it excelled in identifying temporal anomalies and spatial abnormalities, rendering it well-suited for densely populated scenes.[2] Subsequent advancements were spearheaded by Cheng, Chen, and Fang [3], who refined the state-of-the-art by concurrently addressing temporal and spatial anomalies. Their pioneering approach involved extracting normal interactions from training videos efficiently, followed by constructing interaction templates. These templates were then modeled using Gaussian Process Regression, facilitating the detection of temporal anomalies and spatial abnormalities with discriminant saliency. The efficacy of this method surpassed that of existing techniques, particularly in crowded environments.

In recent years, deep learning has revolutionized video anomaly detection, exhibiting promising results in capturing intricate patterns and temporal dependencies. Kiran et al.[7] provided an extensive overview of deep learning-based methods, showcasing their potential in anomaly recognition across diverse domains. Similarly, Wang et al.[9] introduced a method centered on exploiting spatio-temporal relationships among objects, markedly enhancing feature extraction and detection accuracy. Additionally, Yang et al.[9] underscored the effectiveness of modeling spatio-temporal relationships, achieving notable recognition rates in real-world scenarios While these advancements represent substantial strides forward. , video anomaly detection still faces several challenges. Adapting to the plethora of anomalies encountered in real-world settings and reducing reliance on pre-trained models that may lack generalizability remains imperative. Chong et al.[4] emphasize the necessity of developing models capable of handling topological deformations and addressing data imbalances in training datasets, highlighting the importance of comprehensive datasets for robust evaluation and benchmarking. From rudimentary background subtraction to sophisticated deep learning-based methodologies, video anomaly detection has evolved significantly, bolstering its efficacy. These advancements have translated into improved accuracy and robustness, positioning video anomaly detection as a pivotal element within contemporary computer vision systems. However, ongoing research endeavors are poised to address challenges pertaining to real-world diversity, data imbalances, and evaluation methodologies, further propelling this pivotal domain forward.

III. PROPOSED WORK

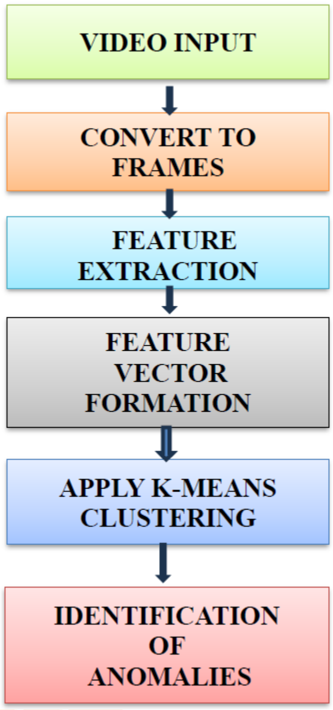

Fig 3.1: video anomaly detection flowchart.

Proposing a system for anomaly detection in videos using the K-Means algorithm involves several steps. K-Means is traditionally used for clustering, but it can be adapted for detecting anomalies based on the distribution of feature vectors in the data. Here's an outline of our system using K-Means for anomaly detection in videos:

1. Video Preprocessing:

- **Frame Extraction:** Break the video into individual frames.
- **Feature Extraction:** - Extract relevant features from each frame. These could include color histograms, texture features, or more complex features derived from deep neural networks.

2. Feature Vector Formation: For each frame, create a feature vector based on the extracted features. Each feature vector represents a point in the feature space.

3. K-Means Clustering: Apply the K-Means algorithm to cluster the feature vectors into k clusters. Choose an appropriate value for k based on the characteristics of the data and the expected number of normal patterns.

4. Centroid Assignment: Assign each feature vector to the nearest cluster centroid.

5. Anomaly Detection: Calculate the distance (e.g., Euclidean distance) between each feature vector and its assigned cluster centroid. Define a threshold for the distance beyond which a feature vector is considered an anomaly. This threshold could be determined empirically or through statistical analysis.

6. Identification of Anomalies: Frames with feature vectors having distances exceeding the defined threshold are considered anomalies. Optionally, we can apply temporal smoothing techniques to reduce false positives by considering the context of neighboring frames.

IV. RESULTS AND DISCUSSION

UCSD Perl Dataset Benchmark

The UCSD Ped1 dataset provides 34 short clips for training, and another 36 clips for testing. All testing clips have frame-level ground truth labels, and clips have pixel-level ground truth labels. There are 200 frames in each clip.

Results:

1. **Detection Time**: The time taken by our proposed algorithm to detect anomalies in each frame is very less. Real-time or near real-time detection is crucial for practical applications, especially in scenarios where quick responses are required.

2. **Voice Announcement Accuracy:** Evaluate the accuracy and clarity of voice announcements generated by pyttsx3. Ensure that the announcements are clear and easily understandable to the user.

Discussion:

1. **Effectiveness of K Means:** Using the K-Means algorithm for anomaly detection in video frames offers both strengths and limitations. It provides a straightforward and scalable approach for clustering feature vectors extracted from video frames. The feature extraction process efficiently prepares the data for clustering, making it suitable for large datasets. Moreover, KMeans operates as an unsupervised learning algorithm, eliminating the need for labeled data during training, which is advantageous for anomaly detection tasks where anomalies may not be explicitly labeled. However, there are notable limitations to consider. KMeans is sensitive to the number of clusters (K), as seen in the code where manual tuning is required. This process can be time-consuming and requires domain expertise. Additionally, KMeans assumes that clusters are spherical and of similar size, which may not accurately reflect the complex shapes and sizes of anomalies in video frames. Furthermore, the algorithm's sensitivity to outliers can affect its accuracy, as outliers may distort cluster centroids and impact anomaly detection results. In conclusion, while KMeans provides a straightforward method for anomaly detection in video frames, its effectiveness depends on various factors, including parameter tuning and the nature of the data. It serves as a useful baseline but may require additional techniques to address its limitations and achieve more accurate and robust anomaly detection results.

2. **Voice Announcement Impact:** Voice announcements play a crucial role in enhancing the usability of anomaly detection systems by providing auditory feedback in addition to visual alerts. In the program code, the integration of `pyttsx3` library enables the system to deliver spoken alerts, which can be particularly beneficial in scenarios where users may not be actively monitoring visual displays, such as during real-time anomaly detection in video streams. The `pyttsx3` library is utilized to synthesize text into speech, allowing the system to verbally communicate important information to the user. For example, in the `import_and_test_abnormal` function, the script prompts the user with spoken messages such as "Press enter to show the abnormal frames" and "Press Enter to continue or 1 to replay." This auditory feedback prompts the user to take action or provides additional context about the anomaly detection process. Moreover, the integration of voice announcements provides accessibility benefits, making the system more inclusive for users with visual impairments. By combining auditory feedback with visual alerts, the system ensures that anomalies are promptly communicated to users through multiple modalities, improving overall situational awareness. Furthermore, auditory alerts can be particularly effective in environments with high levels of noise or when users are multitasking. In such scenarios, spoken alerts can cut through distractions and ensure that users are promptly notified of anomalies without relying solely on visual cues. Overall, the integration of `pyttsx3` enhances the usability of the anomaly detection system by providing auditory feedback in addition to visual alerts. This improves user engagement, accessibility, and situational awareness, ultimately leading to a more effective anomaly detection solution.

```
>>>
= RESTART: D:\python projects\abnormal event detection\Abnormal-Event-Detection-master\Code\project.py
Enter Process method : train
Number of Frames imported :  120

--------------Done Feature Extraction------------
Number of feature generated :  4992
Length of each feature :  1000
------------------------------------------------

files done: 1
Number of Frames imported :  150

--------------Done Feature Extraction------------
Number of feature generated :  6240
Length of each feature :  1000
------------------------------------------------

files done: 2
Number of Frames imported :  150
```

Fig 4.1: training dataset from given train folder.

```
*Python 3.7.9 Shell*
File Edit Shell Debug Options Window Help
frames/Test/Test1/Test018_gt []
frames/Test/Test1/Test019 []
frames/Test/Test1/Test019_gt []
frames/Test/Test1/Test020 []
frames/Test/Test1/Test021 []
frames/Test/Test1/Test021_gt []
frames/Test/Test1/Test022 []
frames/Test/Test1/Test022_gt []
frames/Test/Test1/Test023 []
frames/Test/Test1/Test023_gt []
frames/Test/Test1/Test024 []
frames/Test/Test1/Test024_gt []
frames/Test/Test1/Test025 []
frames/Test/Test1/Test026 []
frames/Test/Test1/Test027 []
frames/Test/Test1/Test028 []
frames/Test/Test1/Test029 []
frames/Test/Test1/Test030 []
frames/Test/Test1/Test031 []
frames/Test/Test1/Test032 []
frames/Test/Test1/Test032_gt []
frames/Test/Test1/Test033 []
frames/Test/Test1/Test034 []
frames/Test/Test1/Test035 []
frames/Test/Test1/Test036 []
Number of Frames imported :  200

--------------Done Feature Extraction------------
Number of feature generated :  8320
Length of each feature :  1000
------------------------------------------------

#######################################
frames/Test/Test1/Test001  video is ::
Normal
#######################################
Number of Frames imported :  200
```

```
*Python 3.7.9 Shell*
File Edit Shell Debug Options Window Help
#######################################
frames/Test/Test1/Test003_gt  video is ::
Normal
#######################################
Number of Frames imported :  200

--------------Done Feature Extraction------------
Number of feature generated :  8320
Length of each feature :  1000
------------------------------------------------

#######################################
frames/Test/Test1/Test004  video is ::
Normal
#######################################
Number of Frames imported :  200

--------------Done Feature Extraction------------
Number of feature generated :  8320
Length of each feature :  1000
------------------------------------------------

#######################################
frames/Test/Test1/Test004_gt  video is ::
Normal
#######################################
Number of Frames imported :  200

--------------Done Feature Extraction------------
Number of feature generated :  8320
Length of each feature :  1000
------------------------------------------------

#######################################
frames/Test/Test1/Test005  video is ::
Abnormal at time50 seconds.
Abnormal at time55 seconds.
```

Fig 4.2: testing dataset from given test folder.

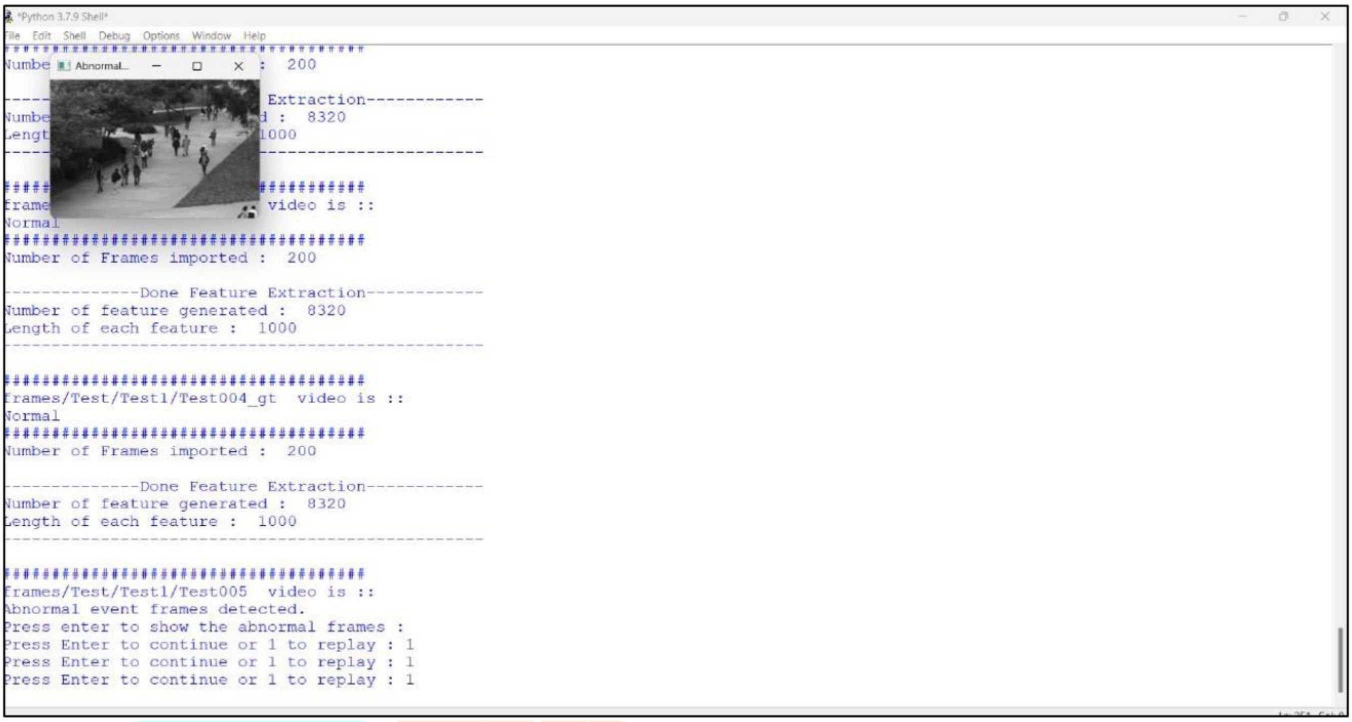

Fig 4.3: detection of anomalies from test folder.

V. CONCLUSION

The project successfully implements abnormal event detection in video frames using the UCSD dataset and the KMeans algorithm. By analyzing the frames in the provided folder, the system identifies abnormal instances and prompts notifications via voice announcements using the Pyttsx3 library.

Key Highlights:

1. Algorithm Choice: The utilization of the KMeans algorithm for abnormal event detection showcases a thoughtful choice. KMeans clustering efficiently segregates normal and abnormal patterns within the video frames, enabling effective anomaly detection.

2. Dataset Selection: The UCSD dataset is a suitable choice for this project due to its variety of real-world scenarios and labeled abnormal events. Leveraging this dataset ensures that the model is trained on diverse instances, enhancing its generalization capability.

Future Improvements:

As we review the system, there are some areas where we could make improvements to enhance its effectiveness. Firstly, we could work on improving the voice announcements to provide more informative feedback to users. By using techniques from natural language processing, we could make the announcements more context-aware and adjust the speech rate and tone based on the severity of the detected anomalies. Adding sentiment analysis could also help convey the urgency of the situation. In terms of the user interface, we could develop a graphical interface that allows users to interact more easily with the system. This could include features like real-time feedback, parameter adjustment, and interactive anomaly exploration. For example, users could zoom in on specific frames or regions of interest to get a better understanding of the detected anomalies. It would be beneficial to enable real-time anomaly detection capabilities. This would involve optimizing our algorithms and processing pipelines to handle streaming video data efficiently. Implementing techniques for real-time feature extraction and adaptive parameter tuning, as well as exploring parallel processing, could help ensure that we can detect and respond to anomalies as they occur.

V. REFERENCES

[1] G. Walker, “World weather,” Quarterly Journal of the Royal Meteorological Society, vol. 54, no. 226, pp. 79–87, 1928.

[2] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, “Lof: identifying density-based local outliers,” in ACM sigmod record, vol. 29, no. 2. ACM, 2000, pp. 93–104.

[3] J. Kim and C. D. Scott, “Robust kernel density estimation,” Journal of Machine Learning Research, vol. 13, no. Sep, pp. 2529–2565, 2012.

[4] J. F. MacGregor and T. Kourti, “Statistical process control of multivariate processes,” Control Engineering Practice, vol. 3, no. 3, pp. 403–414, 1995.

[5] B. Sch ¨olkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, “Estimating the support of a high-dimensional distribution,” Neural computation, vol. 13, no. 7, pp. 1443–1471, 2001.

[6] E. Keogh, J. Lin, and A. Fu, “Hot sax: Efficiently finding the most unusual time series subsequence,” in IEEE International Conference on Data Mining (ICDM). IEEE, 2005, pp. 8–pp.

[7] H. Ren, M. Liu, X. Liao, L. Liang, Z. Ye, and Z. Li, “Anomaly detection in time series based on interval sets,” IEEJ Transactions on Electrical and Electronic Engineering, 2018.

[8] S. Liu, M. Yamada, N. Collier, and M. Sugiyama, “Change-point detection in time-series data by relative density-ratio estimation,” Neural Networks, vol. 43, pp. 72–83, 2013.

[9] P. Senin, J. Lin, X. Wang, T. Oates, S. Gandhi, A. P. Boedihardjo, C. Chen, and S. Frankenstein, “Grammarviz 3.0: Interactive discovery of variable-length time series patterns,” ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 12, no. 1, p. 10, 2018.

[10] M. Jiang, A. Beutel, P. Cui, B. Hooi, S. Yang, and C. Faloutsos, “A general suspiciousness metric for dense blocks in multimodal data,” in IEEE International Conference on Data Mining (ICDM). IEEE, 2015, pp. 781–786.