Real-Time Anomaly Detection Using Deep Learning and Computer Vision

Chauhan Hitansh
Karnavati University

Dhairya Shah
Karnavati University

Vishva Patel
Karnavati University

Abstract

This paper presents a real-time anomaly detection system using deep learning and computer vision, specifically targeting the detection of smoking and spitting activities. Utilizing a model trained through Google's Teachable Machine platform and deployed using TensorFlow and OpenCV, the system is capable of live webcam monitoring and anomaly alert generation. Our system shows that machine learning can effectively enhance traditional surveillance systems with automated behavior recognition.

1. Introduction

Public health and safety are increasingly important in crowded urban spaces. Manual monitoring is time-consuming, prone to human error, and inefficient when scaling to large environments. Automated systems for recognizing hazardous or socially undesirable behavior, such as smoking in prohibited areas or spitting in public places, are critical innovations. In this study, we explore the application of deep learning for real-time anomaly detection.

2. Literature Review

Previous works in anomaly detection have focused largely on general activity recognition or crowd behavior analysis. Convolutional Neural Networks (CNNs) have been particularly effective in visual recognition tasks. Techniques like YOLO (You Only Look Once), MobileNet, and custom CNN architectures have achieved high performance in object and action recognition. However, specific applications for socially undesirable behaviors such as smoking and spitting detection remain relatively underexplored.

3. Methodology

The methodology for developing the anomaly detection system is divided into several key stages: data preparation, model training, deployment, and real-time inference.

3.1 Data Preparation

Images and videos representing smoking, spitting, and normal activities were collected. The dataset was manually labeled, and data augmentation techniques (rotations, brightness changes, flips) were applied to improve model generalization.

3.2 Model Training

The model was trained using Google's Teachable Machine platform. A MobileNet-based architecture was selected for transfer learning. The model was trained for 100 epochs using the Adam optimizer, minimizing categorical cross-entropy loss.

3.3 Deployment

The trained model was exported and integrated into a Python application using TensorFlow. OpenCV was used to capture and process frames from the webcam. Each frame was resized, normalized, and passed through the model to predict behavior classes.

---

4. Experiments

Experiments were conducted to evaluate the model's performance in different lighting conditions, varying camera angles, and different subject appearances.

4.1 Evaluation Metrics

- **Accuracy**: Overall correct predictions.
- **Precision**: Correct positive predictions over total positive predictions.
- **Recall**: Correct positive predictions over actual positives.
- **F1-Score**: Harmonic mean of precision and recall.
- **Latency**: Time taken to process each frame.

---

5. Results

The system achieved an overall accuracy of 87.5% across all classes.

Class	Precision	Recall	F1-Score
Smoking	90%	88%	89%
Spitting	88%	85%	86.5%
Normal	93%	95%	94%

Real-time frame processing maintained an average latency of approximately 50 milliseconds, ensuring a smooth user experience.

---

6. Discussion

The proposed system demonstrates that lightweight deep learning models can be effectively used for anomaly detection in real-time settings. However, detection accuracy declined under poor lighting conditions or when subjects were partially occluded. Although Teachable Machine simplified model training, it may not scale effectively for detecting multiple or more complex behaviors without custom modifications.

---

7. Conclusion

This study proves that integrating deep learning with computer vision can significantly enhance surveillance systems by enabling real-time anomaly detection. The simplicity, low cost, and high accuracy make this approach suitable for urban monitoring systems.

---

8. Future Work

Future improvements may include:

- Expanding the anomaly categories (e.g., vandalism, theft).
- Enhancing detection robustness in different environments.
- Developing a lightweight mobile or edge-device deployment version.
- Integrating multi-camera inputs for broader area monitoring.

References

1. TensorFlow Documentation: https://www.tensorflow.org/
2. OpenCV Documentation: https://docs.opencv.org/
3. Teachable Machine by Google: https://teachablemachine.withgoogle.com/
4. Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement.
5. Howard, A. G., et al. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications.