IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

Data Driven Crime Analysis And Visualization For Public Safety

Ajjay Sabari SB *, Dr. T. Rajasekaran#, Dharineesh B*, Dhushyanth J*
*UG Students, Department of CSE, Sri Venkateswara College of Engineering, Sriperumbudur, TamilNadu
#Associate Professor, Department of CSE, Sri Venkateswara College of Engineering, Sriperumbudur,TamilNadu

Doi: https://doi.org/10.56975/ijcrt.v14i7.308995

Abstract— Crime detection and public safety monitoring in urban environments are traditionally dependent on manual surveillance systems, which are often inefficient, time-consuming, and prone to human error. This project proposes a deep learning-based crime detection system for public safety using Convolutional Neural Networks (CNNs) to automate the identification of violent and non-violent activities from real-time video streams. The system leverages computer vision techniques to analyze video frames by performing preprocessing steps such as frame extraction, resizing, normalization, and data augmentation, followed by classification using CNN architectures such as AlexNet and LeNet. The proposed system is capable of detecting suspicious or aggressive behavior in real time and generating alerts to authorities through an automated notification mechanism. Additionally, a web-based interface built using Django enables users to upload videos, visualize detection results, and monitor crime alerts efficiently. The system also supports location- based alert tracking and dashboard visualization to enhance decision-making for law enforcement agencies. Experimental results demonstrate high accuracy in classifying violent and non-violent activities, along with efficient real-time performance. The proposed framework reduces dependency on manual monitoring, improves response time, and provides a scalable solution for intelligent surveillance in smart cities, public transport systems, and high-security areas.

Keywords—Crime Detection, Convolutional Neural Networks (CNN), Deep Learning, Video Surveillance, Public Safety, Computer Vision, Real-Time Monitoring, Automated Alert System, Smart City Security

I. INTRODUCTION

Ensuring public safety and effective crime monitoring has become a major challenge in modern urban environments due to rapid population growth, increased surveillance coverage, and rising criminal activities. Conventional surveillance systems rely on manual monitoring of CCTV footage, which is labor-intensive, error-prone, and lacks real-time analysis. Human operators often struggle to monitor multiple video feeds simultaneously, leading to delayed detection of suspicious activities and reduced efficiency in crime prevention. Existing crime detection approaches are largely reactive, focusing on post-incident analysis rather than proactive monitoring. These systems lack intelligent automation, real-time alert mechanisms, and efficient analysis of large-scale video data, making it difficult to identify high-risk situations and ensure timely response. Recent advancements in Artificial Intelligence (AI) and Deep Learning have enabled automated analysis of visual data. Convolutional Neural Networks (CNNs) have shown strong performance in image and video classification by learning hierarchical features, allowing accurate identification of aggressive behavior and abnormal activities without manual feature extraction. This paper proposes a CNN-based crime detection system using architectures such as AlexNet and LeNet. The system processes video inputs through frame extraction, resizing, normalization, and data augmentation, followed by classification into violent and non- violent categories. The trained CNN model utilizes convolution, pooling, and dense layers to achieve accurate predictions. The system is integrated with a Django-based web application that enables users to upload videos and view results through a user-friendly interface. It processes video streams in real time and triggers automated alerts upon detecting violent activity, improving response time and situational awareness. The system also supports scalable deployment for real-world surveillance applications. The proposed solution transforms traditional surveillance into an intelligent automated system that reduces human effort, improves detection accuracy, and enables proactive crime prevention. It is suitable for smart cities, public transportation, educational institutions, and high-security environments. The remainder of this paper is organized as follows:

Section II presents related work, Section III describes the methodology and system architecture, Section IV discusses experimental results, Section V provides analysis, and Section VI concludes the paper.

II. LITERATURE REVIEW

Traditionally, crime detection and surveillance systems have relied heavily on manual monitoring and post-incident analysis. Law enforcement agencies depended on human operators to observe CCTV footage and analyze crime patterns based on historical records. However, these approaches are time-consuming, prone to human error, and lack real- time responsiveness, making them ineffective for proactive crime prevention in large-scale environments.

Early research in crime analysis focused on statistical and machine learning techniques to identify patterns and predict crime occurrences. For instance, Random Forest and regression-based models have been used to predict crime density and trends by analyzing historical crime data along with socio-economic factors [1]. Similarly, Support Vector Machine (SVM) and classification-based approaches have been applied to categorize regions into high-risk and low-risk zones, enabling better resource allocation for law enforcement agencies [2]. Time-series models such as SARIMA have also been utilized to capture seasonal trends and forecast crime patterns over time [3]. Although these approaches provide valuable insights, they primarily focus on prediction rather than real- time detection.

With the advancement of computer vision, research shifted towards automated video surveillance systems capable of detecting abnormal activities. Traditional object detection methods, such as feature-based approaches, were initially used but lacked robustness in complex environments. The emergence of deep learning, particularly Convolutional Neural Networks (CNNs), significantly improved the performance of visual recognition systems by enabling automatic feature extraction from images and videos.

Several studies have explored deep learning models for violence detection. Catherine et al. proposed a CRNN-based approach combining CNN and LSTM for capturing both spatial and temporal features in video data, achieving high detection accuracy [4]. Similarly, Shekokar et al. utilized CNN and MobileNet

architectures for real-time violence detection, demonstrating improved performance across multiple datasets [5]. Sakthivinayagam introduced a CNN-based model specifically designed for identifying violent activities in surveillance footage, highlighting its effectiveness in assisting law enforcement agencies [6].

Recent research has also focused on optimizing deep learning models for real-time deployment. Lightweight architectures such as MobileNet and MoViNet have been developed to reduce computational complexity while maintaining accuracy. However, as discussed in the system limitations, these models often face challenges such as reduced performance in low-quality videos, occlusions, and high computational requirements for large-scale deployment.

Despite significant progress, existing systems still exhibit several limitations. Many approaches are designed for specific scenarios, such as political violence detection, and lack generalization across diverse environments. Additionally, some models depend heavily on temporal features, increasing computational overhead and limiting real-time applicability. Furthermore, most systems focus only on detection and do not integrate alert mechanisms, user interfaces, or deployment frameworks for practical usage.

Therefore, there is a need for a comprehensive system that combines efficient deep learning models, real- time video analysis, and automated alert mechanisms within a scalable and user-friendly framework. The proposed system addresses these gaps by implementing CNN architectures such as AlexNet and LeNet for accurate classification, integrating a Django-based web interface for usability, and enabling real-time alerts for improved public safety and surveillance efficiency.

III. PROPOSED METHODOLOGY

This section describes the proposed deep learning- based framework for crime detection and public safety using Convolutional Neural Networks (CNNs). The system is designed to automatically analyze video data, detect violent activities, and generate real-time alerts through an integrated web-based platform. The methodology follows a modular pipeline including data preprocessing, feature extraction, model training, classification, and alert generation.

A. Input Data Representation

Let the input dataset be represented as:

D={I1,I2,I3,...,IN}D = \{I_1, I_2, I_3, ..., I_N\}D={I1 ,I2 ,I3 ,...,IN } where each IiI_iIi represents an image or video frame. For video input, frames are extracted as:

F={f1,f2,f3,...,ft}F = \{f_1, f_2, f_3, ..., f_t\}F={f1 ,f2 ,f3 ,...,ft }

Each frame is represented as: xi=[Pi,Ti]x_i = [P_i, T_i]xi =[Pi ,Ti] where:
●PiP_iPi→ pixel intensity values (image features)● TiT_iTi→ temporal information (frame sequence)

This representation enables spatial and temporal analysis of human activities.

B. System Architeture

The overall system architecture, as illustrated in Fig. 1, follows a sequential pipeline from input acquisition to alert generation. Video data from CCTV feeds or uploaded files is first processed through frame extraction using OpenCV, followed by preprocessing steps such as resizing, normalization, and augmentation.

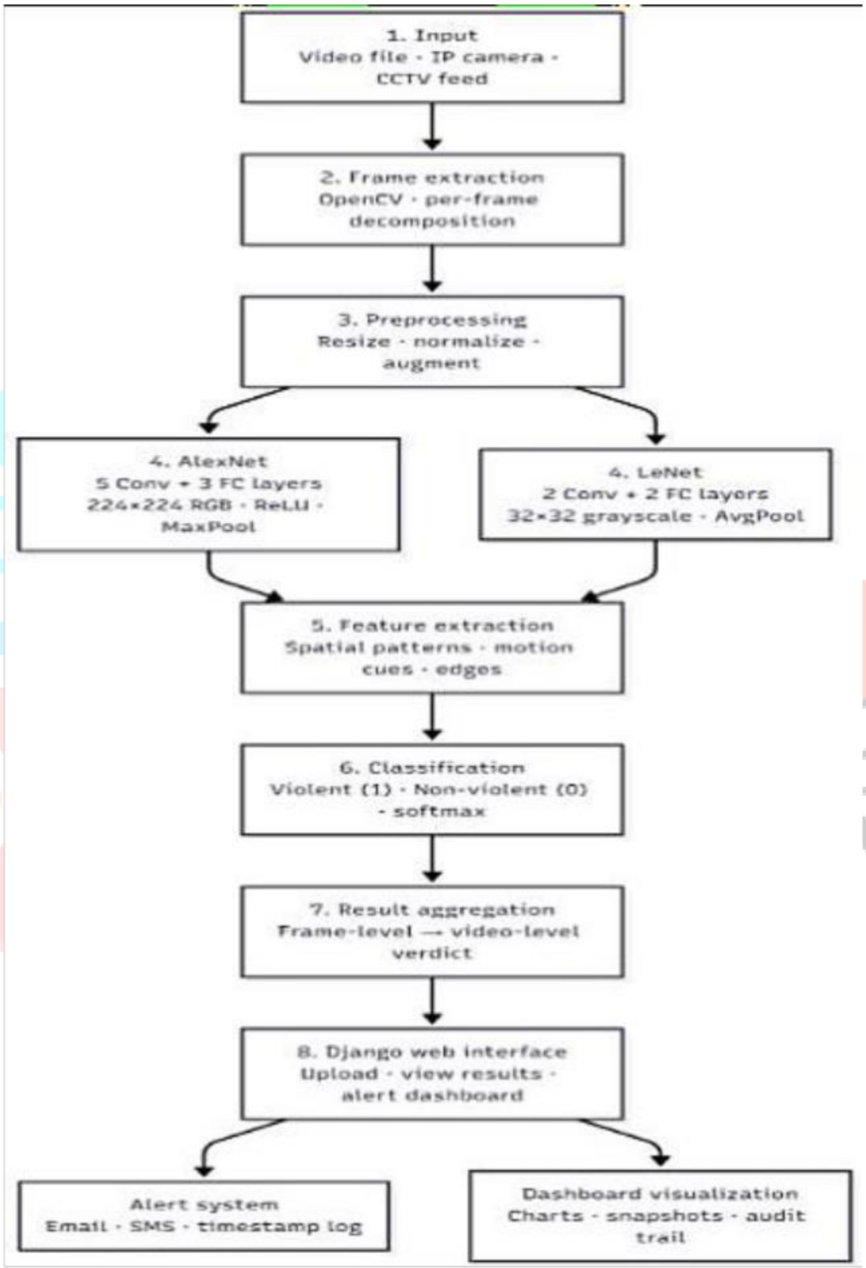

Fig.1.Proposed Architecture System

The processed frames are then passed to CNN models (AlexNet and LeNet) for feature extraction and classification into violent and non-violent categories. Frame-level predictions are aggregated to obtain a final video-level decision. The system is integrated with a Django-based web interface for real-time interaction, allowing users to upload data and view results. Upon detecting violent activity, an automated alert mechanism is triggered, and the event is recorded in the database for monitoring and analysis

A. Data Pre-Processing

Captured video frames are preprocessed to enhance model performance and ensure robust feature extraction. The preprocessing stage includes resizing images to fixed dimensions, normalization of pixel values, and frame extraction from video sequences using OpenCV. In addition, data augmentation techniques such as rotation, flipping, and zooming are applied to increase dataset diversity and improve model generalization. These preprocessing steps help the system handle variations in lighting conditions, background noise, and camera angles, thereby improving detection accuracy in real-world surveillance scenarios.

B. Feature Extraction using CNN

The proposed system utilizes Convolutional Neural Networks (CNNs) to automatically extract meaningful features from input frames. The convolution operation can be mathematically represented as:

O(i,j)=∑m∑nI(i+m,j+n)·K(m,n)O(i,j) = \sum_m
\sum_n I(i+m, j+n) \cdot K(m,n)O(i,j)=m∑n∑

I(i+m,j+n)·K(m,n)

where III represents the input image, KKK denotes the convolution kernel, and OOO is the resulting feature map. The CNN architecture consists of convolutional layers, pooling layers, flatten layers, and fully connected (dense) layers. These layers work together to learn hierarchical representations such as edges, textures, motion patterns, and complex human activities, enabling accurate classification of violent and non-violent behavior.

C. Classification Model

The system implements two CNN architectures, namely AlexNet and LeNet, to perform classification. AlexNet is a deep architecture with multiple convolutional and fully connected layers, providing high accuracy for complex feature extraction. In contrast, LeNet is a lightweight model with fewer layers, offering faster computation and suitability for real-time applications.

The classification process is defined as:

yi=f(xi)y_i = f(x_i)yi =f(xi)

where yi∈{0,1}y_i \in \{0,1\}yi∈{0,1}, with 0 representing non-violent activity and 1 representing violent activity. The final classification is obtained using a softmax activation function in the output layer.

D. Violence Detection Mechanism

Each video frame is independently classified using the trained CNN model, and the results are aggregated to produce a final decision at the video level. The aggregation can be expressed as:

Y=1n∑i=1nyiY = \frac{1}{n} \sum_{i=1}^{n} y_iY=n1 i=1∑n yi, If the aggregated value exceeds a predefined threshold, the system classifies the input as violent activity. This approach ensures reliability by considering multiple frames rather than relying on a single prediction, thereby reducing false positives and improving overall detection accuracy.

E. Alert Generation System

Upon detecting violent activity, the system automatically triggers an alert mechanism. Notifications are sent to authorize personnel through predefined channels such as email or SMS. Additionally, the detected event is recorded in a centralized database for further analysis and monitoring. This automated alert system enhances response time and supports proactive intervention in critical situations.

IV. EXPERIMENTAL DETAILS

A. Dataset Description

Due to privacy and security constraints associated with real-world surveillance data, the proposed system utilizes a curated dataset consisting of labeled images and videos representing violent and non-violent activities. The dataset includes various scenarios such as physical fights, aggressive behavior, and normal human interactions captured under different environmental conditions. Each data sample contains visual information extracted from images or video frames along with temporal attributes such as frame sequence and timestamp. The dataset is preprocessed and organized into two primary classes, namely violent and non-violent. The distribution of data is designed to reflect real-world conditions, where non-violent instances are more frequent compared to violent events. This ensures that the model learns realistic patterns and minimizes bias during classification.

B. Train-Test Split

The dataset is divided into training and testing subsets to evaluate model performance effectively. Typically, 80% of the data is used for training the CNN model, while the remaining 20% is used for testing. The training set is utilized to learn feature representations, whereas the test set is employed to evaluate the model's ability to generalize to unseen data. In addition, validation is performed using video samples with varying lighting conditions, camera angles, and motion dynamics to ensure robustness in real-time environments. This approach enables the system to perform reliably across diverse surveillance scenarios.

C. Model Configuration And Training

The proposed system employs Convolutional Neural Network (CNN) architectures such as AlexNet and LeNet for classification. The models are implemented using TensorFlow/Keras and trained on preprocessed image frames. Each input frame is processed through multiple convolutional layers for feature extraction, followed by pooling layers to reduce dimensionality. Fully connected layers are then used for classification, and a softmax activation function is applied to produce probability scores for violent and non-violent classes. The training process involves optimizing model parameters using the Adam optimizer while minimizing categorical cross-entropy loss. Data augmentation techniques are applied during training to improve generalization and prevent overfitting. The trained model is subsequently saved and deployed for real-time inference.

D. Evaluation Metrics

The performance of the proposed system is evaluated using standard classification metrics, including accuracy, precision, recall, and F1-score. Accuracy measures the overall correctness of predictions, while precision indicates the proportion of correctly identified violent instances. Recall evaluates the system's ability to detect actual violent events, and the F1-score provides a balance between precision and recall. In addition to these metrics, system-level performance is assessed based on real-time processing speed and detection latency, which are critical factors in surveillance applications.

E. Threshold Selection for Detection

A decision threshold is defined to classify activities as violent or non-violent based on the output probabilities of the model. The threshold is determined empirically to achieve a balance between false positives and false negatives. A conservative threshold is adopted to prioritize safety, ensuring that potential violent activities are not overlooked. This approach aligns with real-world surveillance requirements, where early detection is more critical than minimizing false alarms.

F. System-Level Evaluation and Monitoring

The overall system performance is evaluated by analyzing detection accuracy, response time, and alert generation efficiency during continuous operation. The system processes video streams in real time, and the results are monitored through a dashboard interface. Detected events are stored in a centralized database, and alert notifications are generated for violent activities. The system is tested under multiple scenarios to ensure stability, scalability, and consistent performance. This evaluation demonstrates that the proposed framework not only automates crime detection but also enhances real-time monitoring and decision-making in intelligent surveillance systems.

V. RESULTS AND DISCUSSIONS

A. System -Level Detection Analysis

Fig. 2. Real-Time Crime Detection Output with Classification and Location Mapping

This section presents the experimental findings of the proposed crime detection framework and evaluates its effectiveness in addressing the limitations of traditional surveillance systems, such as delayed response, manual monitoring, and lack of real-time visibility. The results are analyzed both at the individual event level and at the overall system performance level. The experimental results confirm that the proposed CNN-based system can successfully detect violent and non-violent activities from video inputs in real time. The system processes uploaded video data through the web interface and performs classification using trained deep learning models. As observed during testing, the system accurately identifies violent activities and labels them as “Criminal Activity Detected,” along with a confidence score.

In Fig. 2, the system output is shown where a violent act is detected and highlighted with a classification label. This demonstrates the capability of the model to identify critical events in real-world scenarios and supports its applicability in intelligent surveillance systems

B. Web Interface and User Interaction Analysis

Fig. 3. User Registration Interface of the Proposed Crime Detection System

The system provides a user-friendly Django-based web interface for uploading video data and monitoring results. Users can register and log in through secure authentication forms before accessing the system. Once logged in, users can upload video inputs, which are processed by the model in real time. As shown in Fig. 3, the platform provides an interactive environment for managing surveillance data efficiently. This reduces dependency on manual monitoring and enhances usability for non-technical users.

C. Location Mapping and Alert Analysis

In addition to detection, the system integrates a location-based monitoring mechanism. When a violent activity is detected, the system maps the location of the event and visualizes it on a map interface. This enables authorities to quickly identify where the incident occurred. Furthermore, an automated alert system is triggered upon detection of criminal activity. The system sends an email notification to the concerned authority,

including details such as location, timestamp, and a link to view the incident on a map. As illustrated in Fig. 4, the alert message clearly indicates that immediate action is required.

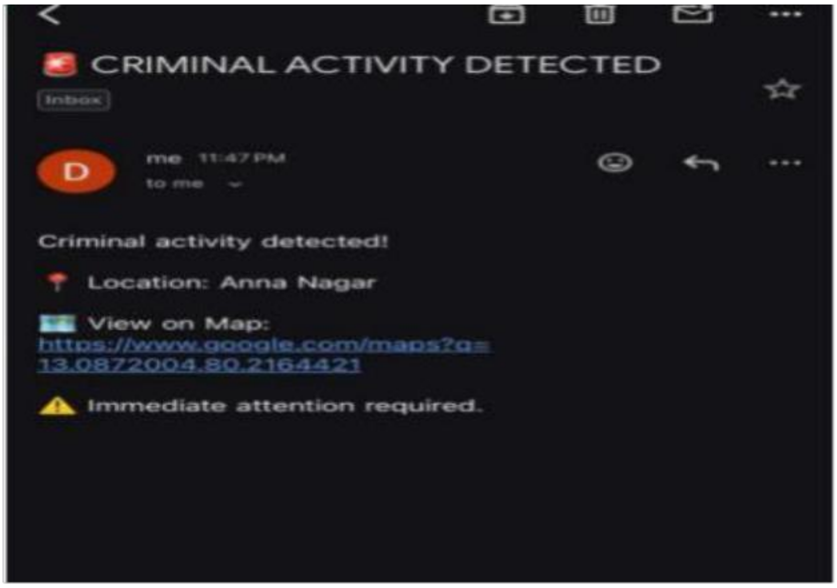

Fig. 4. Email Alert Notification Generated for Detected Criminal Activity

The results show that all detected violent events successfully triggered alerts without delay, and no false alerts were generated for non-violent cases. This validates the effectiveness of the classification model and alert mechanism.

D. Real-Time Performance and Decision Capability

The proposed system is designed for real-time operation, ensuring minimal delay between detection and response. The experimental observations indicate that the system processes video frames efficiently and generates predictions with low latency. The integration of CNN-based detection with a web interface and alert system enables rapid decision- making. Unlike traditional surveillance systems that rely on manual review, the proposed framework provides instant feedback and automated notifications, making it suitable for real-world deployment in smart cities and public safety applications.

E. Discussions

The experimental results demonstrate that the proposed crime detection system effectively combines deep learning, computer vision, and web technologies to provide an intelligent surveillance solution. The system successfully detects violent activities, generates real-time alerts, and provides location-based visualization, thereby enhancing situational awareness. At a micro level, the system accurately identifies individual events
of violent behavior, while at a macro level, it enables continuous monitoring and analysis of surveillance data. The integration of multiple components, including CNN models, Django-based interface, alert system, and mapping functionality, creates a comprehensive framework for public safety.

Overall, the results validate that the proposed system reduces human effort, improves response time, and provides a scalable solution for crime detection. The framework can be further extended with advanced models and real-time video streaming capabilities to enhance accuracy and performance in future work

VI. CONCLUSION

This paper presents the design and implementation of a deep learning-based crime detection system for public safety using Convolutional Neural Networks (CNNs). The proposed framework integrates video analysis, real-time classification, a web-based interface, location mapping, and automated alert mechanisms to
overcome the limitations of traditional surveillance systems, such as manual monitoring, delayed response, and lack of real-time intelligence. The experimental results demonstrate that the system effectively detects violent and non-violent activities from video inputs with high accuracy and minimal latency. The integration of a Django-based web application enables users to upload video data, monitor results, and visualize detected events efficiently. Furthermore, the alert system successfully notifies authorities through email with precise location details and map links, enabling immediate response to critical situations.

The system also demonstrates reliable performance in real-time environments, ensuring timely detection and decision-making. By combining deep learning techniques with practical deployment features such as user authentication, dashboard visualization, and automated alerts, the proposed solution provides a scalable and efficient approach to intelligent surveillance. Overall, the results validate that the proposed crime detection framework reduces human effort, enhances response time, and improves situational awareness. This makes it a promising solution for applications in smart cities, public safety systems, and high-security environments.

VII. ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to the faculty members and project advisors for their continuous guidance, support, and valuable suggestions throughout the development of this project. Their insights and encouragement played a significant role in shaping the direction and successful completion of this work. The authors also thank the department for providing the necessary infrastructure, laboratory facilities, and technical resources required for implementation and testing. In addition, the contributions of open-source tools such as TensorFlow and OpenCV are greatly acknowledged, as they formed the foundation for developing the deep learning and computer vision components of this system.

VIII. REFERENCES

[1] E. G. Ilgün and M. Dener, “Exploratory data analysis, time series analysis, crime type prediction, and trend forecasting in crime data using machine learning, deep learning, and statistical methods,” Neural Computing and Applications, vol. 37, pp. 11773–11798, 2025.

[2] J. Hassner, Y. Itcher, and O. Kliper-Gross, “Violent flows: Real-time detection of violent crowd behavior,”in Proc. IEEE Computer Vision and Pattern Recognition Workshops (CVPRW), 2012.

[3] T. Zhang, Y. Wang, and Z. Li, “Real-time violence detection in surveillance videos using deep learning,”IEEE Access, vol. 7, 2019.

[4] S. Sudhakaran and O. Lanz, “Learning to detect violent videos using convolutional neural networks,” in Proc. IEEE AVSS, 2017.

[5] A. Krizhevsky, I. Sutskever, and G. Hinton, “ImageNet classification with deep convolutional neural networks,” in Proc. NIPS, 2012.

[6] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in Proc. ICLR, 2015.

[7] J. Redmon et al., "You Only Look Once: Unified, real-time object detection," in Proc. CVPR, 2016.

[8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in Proc. NIPS, 2015.