ADAPTIVE SERVER-DRIVEN VIDEO STREAMING FOR REAL-TIME SUSPICIOUS ACTIVITY DETECTION

Sampath Bandara Thennakoon

(209385G)

Master of Science in Computer Science and Engineering
(Specializing in Data Science Engineering and Analytics)

Department of Computer Science and Engineering
University of Moratuwa

Sri Lanka

July 2023

ADAPTIVE SERVER-DRIVEN VIDEO STREAMING FOR REAL-TIME SUSPICIOUS ACTIVITY DETECTION

Sampath Bandara Thennakoon

(209385G)

This dissertation submitted in partial fulfillment of the requirements for the Degree of MSc in Computer Science Specializing in Data Science Engineering and Analytics

Department of Computer Science and Engineering

University of Moratuwa

Sri Lanka

July 2023

DECLARATION

I declare that this is my own work, and this thesis does not incorporate without acknowledgment any material previously submitted for degree or diploma in any other University or institute of higher learning, and to the best of my knowledge and belief, it does not contain any material previously published or written by another person except where the acknowledgment is made in the text.

Also, I hereby grant to the University of Moratuwa the non-exclusive right to reproduce and distribute my thesis, in whole or in part, in print, electronic or other media. I retain the right to use this content in whole or part in future works (such as articles or books).

Signature: …… Date: …15.07.2023…

Name: Mr. Sampath Thennakoon

The supervisor/s should certify the thesis with the following declaration.

The above candidate has carried out research for the Masters thesis under my supervision. I confirm that the declaration made above by the student is true and correct.

Signature of the supervisor: ……. Date: ……

Name: Dr. Charith Chitraranjan

ACKNOWLEDGEMENTS

I want to express my deep gratitude and profound admiration to all those who helped me and encouraged me in any aspect to accomplish this project. My special thanks to the University of Moratuwa for allowing me to complete this project.

Especially, I express profound gratitude to my advisor, Dr. Charith Chitraranjan, for his invaluable support in providing relevant knowledge, materials, advice, supervision, and valuable suggestions throughout this research work. His expertise and continuous guidance enabled me to complete my research successfully. Also, I would like to thank the external supervisors of my project, Dr. Adeesha Wijayasiri and Dr. T. Sivakumar, for their guidance, suggestions, and support throughout the project.

Further, I would like to thank all my colleagues for their help in finding relevant research material, for sharing knowledge and experience, and for their encouragement. I am as ever, especially indebted to my parents and my sister for their love and support throughout my life. Finally, I express my gratitude to all my colleagues for the support given to me in managing my MSc research studies.

Thank you.

ABSTRACT

Real-time video streaming and analytics are the solutions to many safety and management systems. Even though it is widely used for video analytics-oriented applications everywhere, it must first consider the importance bounded by inference accuracy and the usage of resources such as hardware or network. Especially internet-based video streaming applications (IOT) must be adjusted to make the best use of application-level quality and adjusted to the limited network infrastructures. With deep neural networks (DNNs), instead of outmoded video streaming techniques, new techniques for optimizing the visual quality, such as rapid compression or pruning of pixels in outside areas, can achieve high inference accuracy. Currently, most of the device-based surveillance video streaming hinges mainly on the camera, the video generation device, which manages which frames and pixels to transfer through the network. However, the video camera-driven technique has aided well for typical video streaming applications. Also, real-time analytics-oriented applications still suffer from the source-driven application-level approach, where the video generation devices can only estimate the quality, and it takes work to get the measurement of the user experience directly.

Similarly, most streaming protocols are guided by video sensor equipment (sensors or cameras), where the device computing power is deficient compared with the computers / mobile phones we use in our day-to-day life. A novel approach of DNN-driven streaming (DDS), which can be used to transfer a limited quality camera clips stream to the distance server while saving a lot of network bandwidth, and the server process the advanced deep neural network model in parallelly to regulate with the higher quality of stream of video clips to increase the inference accuracy moreover the Quality of Experience (QoE) over the existing commercial vision analytics-based surveillance monitoring systems. In the vision-based surveillance monitoring industry, understanding the scene's context is one of the most important aspects of a new generation of video surveillance markets. Recognizing the scene context from just one glance at the image is relatively easy for a human. However, it is still hard for a machine to accomplish. With the help of deep-learning algorithms and scene-type labels as a guide, we can identify the scenes to create a balanced dataset for training with advanced instance identification models.

In this research, I implemented an adaptive DNN-driven streaming application with a novel approach for real-time identifying suspicious activities and unique scene detection. The latter part of this research will propose a generalized form of video analytics pipeline (VAP) approach, which can be suited for any AI-driven domain scenario.

The newer version of the server-driven video streaming project for suspicious activity detection with demonstrations can be set up on any machine with GPU. Also, this research provides a custom suspicious activity detection dataset (based on the CAVIAR dataset[1], 2004), which is publicly available at https://github.com/sampaththennakoon/caviar_data_set, and the upgraded version of the server driven video streaming for suspicious activity detection project is available for download at https://github.com/sampaththennakoon/dds-v2.

Keywords: Suspicious Activity Detection, Server Driven Video Streaming, Deep Learning

TABLE OF CONTENTS

DECLARATION ..... **i**

ACKNOWLEDGEMENTS ..... **ii**

ABSTRACT ..... **iii**

TABLE OF CONTENTS ..... **v**

LIST OF FIGURES ..... **ix**

LIST OF TABLES ..... **x**

LIST OF ABBREVIATIONS ..... **xi**

LIST OF APPENDICES ..... **xii**

Chapter 1 - INTRODUCTION **1**

1.1: Background 1

1.2 Introduction to Real-Time Video Analytics Applications 2

1.3 Server-Driven Video Analytics Applications 3

1.4 Research gaps and Problems 4

1.5 Objectives 6

1.6 Motivation 7

1.7 Thesis Organization 8

Chapter 2 - LITERATURE REVIEW **9**

2.1: Video Analytics Systems 9

2.1.1 Multi-App Concurrent Execution 11

2.1.2 Video Analytics Pipeline Specific Camera Selection 11

2.1.3 Cross-Camera Collaboration 11

2.2: Optimizations Related with Video Analytics Techniques 11

2.2.1 Cross Camera Workload Sharing and Adaptation of Resource Scheduling ... 13

2.3: Video Analytics Pipelines 14
2.3.1: Recent Efforts of VAP Implementations 15
2.3.1.1: DDS 15
2.3.1.2: AWStream 15
2.3.1.3: Glimpse 16
2.3.1.4: Reducto 17
2.3.1.5: Vigil 17
2.3.1.6: DeepDecision 17
2.3.1.7: Chameleon 18
2.3.1.8: NoScope 19
2.3.2: Different types of VAPs 19
2.3.2.1: Reduction of the internet usage when cameras have a reduced processing capability 20
2.3.2.2: Reduction of the internet usage with splitting the responsibility between the camera and the server 20
2.3.2.3: Reduction of the computes the cost of a resource-constrained devices 21
2.3.3: Limitations of existing video analytics pipelines 21
2.4: Techniques for optimized VAP Implementation 22
2.4.1: Feedback regions 22
2.4.2: Object detection (based on bounding boxes) 23
2.4.3: The Approach of Semantic Segmentation 23
2.4.4: Dynamic ROI Implementation with Encoding 24
2.4.5: Adaptive Offloading 24
2.4.6: Parallelization on Video Streaming and Inference 25
2.4.7: Dependency-Aware Inference 25

2.4.8: Mobile Vision Offloading 25
2.4.9: Adaptive Bitrate Streaming 26
2.5: Deep learning architectures for image classification in video analytics 27
2.6: Deep learning architectures for real-time video analytics 29
2.6.1: 2D object detection via two stages approach 30
2.6.2: 2D object detection via single-stage approach 31
Chapter 3 – METHODOLOGY 34
3.1 Introduction 34
3.2 Data Sets 34
3.2.1 Introduction 34
3.2.2 Extraction of Data 37
3.2.3 Annotation of Data 38
3.2.3.1 Image Data Annotation Tools 38
3.2.3.2 Image Data Annotation Formats 40
3.2.4 Augmentation of Data 43
3.3 Development of the Model 43
3.3.1 Model Preparation 43
3.3.2 Selection of the Best Object and Activity Detection Model 44
3.3.3 Introduction to YOLO 49
3.3.3.1 Specialty About YOLO 49
3.3.3.2 Architecture Summery 50
3.3.3.3 YOLO v1 Network Design 55
3.3.3.4 History of YOLO 56
3.3.4 Training Dataset 63
3.3.5 Hyperparameter Tuning 64

3.3.6 Model Training and Monitoring....65
3.3.7 Training Results and Validation Curves....67
3.3.8 Training Process....69
3.4 Object Tracking Methods....72
3.4.1 Centroid Tracking....72
3.4.2 Bounding Box Tracking....73
3.5 DDS Object Detection Overall Iterative Workflow....74
3.6 Activity Detection System Architecture....77
3.6.1 Single Image Activity Detection....78
Chapter 4 EVALUATION....79
4.1 Introduction....79
4.2 Different Evaluation Metrics....79
4.2.1 Binary Class Classification....79
4.2.2 Multiclass classification metrics....83
4.2.3 Object Detection Metrics....85
4.3 Model Evaluation....88
4.3.1 Evaluating Object Detection Models....88
Chapter 5 - DISCUSSION & CONCLUSION....91
5.1 Discussion....91
5.2 Future Improvements....92
5.3 Conclusion....93
Chapter 6 - REFERENCES....94
APPENDIX A....105
Training and Validation Results....105

LIST OF FIGURES

Figure 2.1: General Architecture of a Video Analytics Platform. 10
Figure 2.2: General Overview of a Video Analytics Pipeline..... 14
Figure 3.1: Annotation formats in CAVIAR [1] dataset. 42
Figure 3.2: CAVIAR Custom Data Set Label Distribution. 43
Figure 3.3: Latest Object Identification State of the Art Architectures. 47
Figure 3.4: Comparison of Frames Processed Per Second in Various Models [84]. 48
Figure 3.5: YOLO Bounding Box Vector Generation for Each Object in an Image. 52
Figure 3.6: Calculation of the Confidence Score in Predicted Bounding Box 53
Figure 3.7: Sample IOU Confidence Scores and identification of TP, TN and FP 53
Figure 3.8: Applying Non-Max Suppression for An Image in YOLO v1. 54
Figure 3.9: YOLO v1 Architecture. 55
Figure 3.10: YOLO v5 (The EfficientDet) Architecture..... 60
Figure 3.11: YOLOv6 Framework Architecture..... 61
Figure 3.12: Weights & Biases Model learning Metrices. 67
Figure 3.13: Series of Entity Overlapping Frames..... 69
Figure 3.14: Weights & Biases Model Training Evaluation. 70
Figure 3.15: Different Matrices Provided in the YOLOv8 Implementation. 71
Figure 3.16: DSS Iterative Workflow. 75
Figure 3.17: Time Line Utilization of DSS Workflow. 77
Figure 3.18: Single Image Activity Detection Flow Chart 78
Figure 4.1: Binary Class Confusion Matrix. 80
Figure 4.2: Average Precision Calculation Formula. 87
Figure 4.3: The Formula of Mean Average Precision. 87
Figure 4.4: The Precision Graph. 89
Figure 4.5: The Recall Graph. 89
Figure 4.6: The AP@0.5 Graph. 89
Figure 4.7: The AP@0.5:0.95 Graph. 89

LIST OF TABLES

Table 2.1: Video Analytics Techniques and Optimizations. 12
Table 2.2: Research Matters Prior of Vision Based Analytics Platforms. 12
Table 2.3: Types of VAPs based on the techniques used in the source device. 19
Table 3.1: Comparison of publicly available human activity tracking datasets. 37
Table 3.2: Occurrences of Activity Incidences for Respective Labels in the Dataset. 42
Table 3.3: Google Collab Pro Server Specification. 66
Table 3.4: Google Collab Free Server Specification. 66
Table 4.1: YOLOv8 Model Training Metrics. 88
Table 4.2: Model Training Metrics. 90

LIST OF ABBREVIATIONS

WAN:	Wide Area Network	AI:	Artificial intelligence
DETR:	DEtection TRansformer	AR:	Augmented Reality
DDS:	DNN Driven Streaming	HD:	High Definition
AWstream:	Adaptive Wide-Area Streaming Analytics	DNN:	Deep Neural Network
FPN:	Feature Pyramid Networks	DRE:	Dynamic ROI Encoding
VAP:	Video Analytics Pipeline	DP:	Dual Prediction
GPU:	Graphics Processing Unit	DC:	Data Compression
PSI:	Parallel Streaming and Inference	TPU:	Tensor Processing Unit
NPU:	Neural Processing Unit	ABR:	Adaptive Bit Rate
RTMP:	Real Time Messaging Protocol	IoT:	Internet of Things
DASH:	Dynamic Adaptive Streaming over HTTP	NN:	Neural Network
ACM:	Association for Computing Machinery	CV:	Computer Vision
COCO:	Common Objects in Context	SLO:	Service Level Objectives
RPN:	Region Proposal Network	QoE:	Quality of Experience
YOLO:	You Only Look Once	IoU:	Intersection over Union
SSD:	Single Shot Multi Box Detector	ROI:	Region Of Interest
IEEE:	Institute of Electrical and Electronics Engineers	HLS:	HTTP Adaptive Streaming
FCOS:	Fully Convolutional One-Stage Object Detection	TP:	True Positive
FP:	False Positive	TN:	True Negative
FN:	False Negative	mAP:	Mean Average Precision
AP:	Average Precision	OvO:	One vs One
CCTV:	Closed-Circuit Television	OVR:	One vs Rest
LSTM	Long Short-Term Memory		

LIST OF APPENDICES

Appendix	Description	Page
Appendix - A	Training and Validation Results	105
Appendix - B	Additional Materials (DVD)	108