Cross-Layer Optimization for Automated Video Surveillance

Mohammad Alsmirat
Department of Computer Science
Jordan University for Science and Technology
Irbid 48202, Jordan
Email: masmirat@just.edu.jo

Nabil J. Sarhan
Department of Electrical and Computer Engineering
Wayne State University
Detroit, Michigan 48202, USA
Email: nabil.sarhan@wayne.edu

***Abstract*—This paper develops an accuracy-based cross-layer optimization solution for wireless automated video surveillance systems, in which multiple sources stream videos to a central proxy station. The proposed solution manages the application rates and transmission opportunities of various video sources based on the dynamic network conditions in such a way that maximizes the overall detection accuracy of the computer vision algorithm(s). We demonstrate the effectiveness of the proposed solution through extensive experiments.**

***Index Terms*—Automated Video Surveillance, Bandwidth Allocation, Computer Vision Algorithm, Cross-Layer Optimization.**

I. Introduction

The interest in video surveillance systems has grown dramatically. Automated Video Surveillance (AVS) serves as an efficient approach for the realtime detection of threats and for monitoring their progress. Most research on AVS focused on developing robust computer vision algorithms for the detection, tracking, and classification of objects and the detection and classification of unusual events [1], [2] (and sources within). Much less work, however, considered the scalability and cost of video surveillance systems. The scalability-cost problem arises because increasing the coverage through employing additional video sources leads to increasing the required bandwidth and the computational capability to process all these video streams. Power consumption is another major problem, especially in battery-powered (wireless) video sources.

This paper considers an AVS system in which multiple video sources capture and send video streams to a central station over a single-hop IEEE 802.11e wireless LAN (WLAN). As shown in Figure 1, the wireless video sources (video cameras or sensors) share the same medium and can be either battery-powered or outlet-powered. The central proxy station is connected with a high-bandwidth link to the access point, and thus this link is not deemed as a bottleneck. The proxy station runs computer vision algorithms to generate automated alerts whenever suspicious threats, events, or subjects are detected in the monitored site. Large systems can be composed of multiple such systems or cells.

The main objective in this study is to provide a cross-layer optimization solution that dynamically distributes and allocates the available network bandwidth among various video sources in such a way that optimizes the overall weighted detection accuracy. This solution utilizes and controls parameters in the application, link, and physical layers. A few studies [3], [4] considered distortion-based optimization in video streaming systems where videos are delivered from many sources to one destination. Our approach differs in the sense that we seek to optimize the overall threat, event, or subject detection or recognition accuracy, which is the single most important metric in AVS. Up to our knowledge, this is the first solution that optimizes the detection/recognition accuracy. Recent studies suggest that computer vision algorithms are not as sensitive as humans are to video quality [5], [6], and thus our proposed accuracy-based optimization approach is expected to be highly effective in AVS systems.

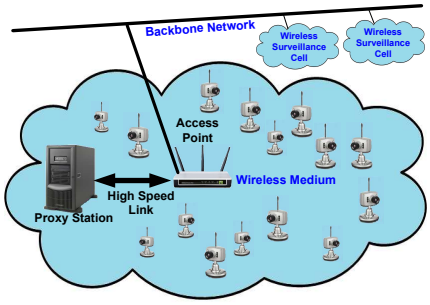


Fig. 1: A Wireless Video Surveillance System

We formulate the bandwidth allocation problem as a cross-layer optimization problem of the sum of the weighted detection accuracy (or alternatively the sum of the weighted detection error), subject to the constraint in the total available bandwidth. The weights represent the current importance levels of various video sources. The total available bandwidth can be modeled by the effective medium airtime, which is the fraction of the network time that is used for delivering useful data. Study [4] developed an online and dynamic effective airtime estimation algorithm, which addresses the limitations of the analytical models in [3], [7]. We use that estimation algorithm in this work.

The rest of this paper is organized as follows. Section II presents the proposed cross-layer optimization framework. Subsequently, Section III discusses the performance evaluation methodology. Finally, Section IV presents and analyzes the

This work was supported in part by NSF grant CNS-0834537.

main results.

II. Proposed Accuracy-Based Cross-Layer Optimization Solution

Our main objective is to provide a cross-layer optimization solution that dynamically distributes and allocates the available network bandwidth among various video sources in such a way that optimizes the overall weighted detection accuracy. This solution utilizes and controls parameters in the application, link, and physical layers. As shown in Figure 1, the system has $S \geq 1$ video sources and each source s streams a different encoded video at rate r_s. Each video source s may have a different physical rate (y_s) and importance level or weight (w_s). The weight represents the importance level of a video source at the current time. To keep the paper focused, we assume that the weights are predetermined.

A. Optimization Problem Formulation

The main idea of this study is to formulate the bandwidth-management problem as a cross-layer optimization problem of the overall weighted error in the detection accuracy of the computer vision algorithm running at the central proxy station. Since all video sources share the same medium, the bandwidth allocation solution should determine the fraction of airtime (f) that each video source receives, with the total not exceeding the effective medium airtime (A_{eff}). Specifically, the problem can be formulated as follows:

$$Find \quad F^* = \arg\min_{F} \sum_{s=1}^{S} w_s \times AccuracyError_s(r_s) \quad (1a)$$

$$s.t. \quad \sum_{s=1}^{S} f_s = A_{eff} \quad (1b)$$

$$r_s = f_s \times y_s \quad (1c)$$

$$0 \leq f_s \leq 1 \quad (1d)$$

$$s = 1, 2, 3, ..., S, \quad (1e)$$

where F^* is the set of optimal fractions (f_s^*) of the airtime of all sources. The optimal application-layer rate of video source s is referred to as r_s^*. A_{eff} is estimated using the online estimation algorithm proposed in [4]. Since the solution requires characterizing the accuracy error, let us first discuss this characterization before proceeding to the optimization solution.

B. Rate-Accuracy Characterization

We seek to develop a model characterizing the relationship between the video bitrate and the accuracy error of computer vision algorithms. To keep the paper focused, we analyze only face detection. Experimenting with other computer vision algorithms is left for another study. We use the Viola-Jones algorithm [8] as implemented in OpenCV. We experiment with MJPEG surveillance video streams. Due to power consumption and other considerations, MJPEG is still used in many surveillance applications, especially in battery-operated cameras.

We determine the rate and accuracy error relationship based on a variety of image datasets as described in Section III. For each image set, we use the IJG JPEG library to compress each image with quality factors ranging from 1 to 100, with 1 being the lowest, and then apply the computer vision algorithm on each image to calculate the accuracy using predefined ground truth about the location of the faces in the image. We use two metrics for the detection accuracy: *positive index* and *negative index*. The positive index is the number of correctly detected faces divided by the total number of faces, whereas the negative index is determined as the number of incorrectly detected faces divided by the total number of faces. Subsequently, we can find the average size, the average positive index, and the average negative index of all images with the same quality factor. The accuracy error can then be calculated as the sum of the total error: $AccuracyError = (1 - PositiveIndex) + NegativeIndex$.

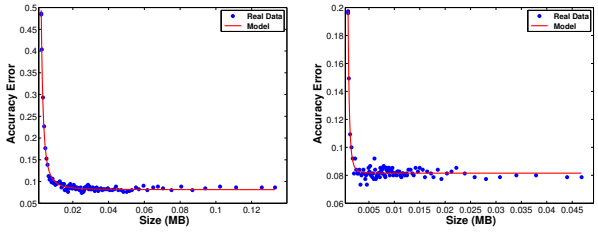
For comparative purposes, we also perform rate-distortion characterization in a similar manner. We assess the distortion of each video frame against the uncompressed video frame using the Root Mean Square Error (RMSE) metric. The video distortion can then be calculated as the average frame distortion of all the frames in the video.

Figure 2 shows the results of the rate-accuracy and rate-distortion characterizations for MJPEG The results with the other considered datasets that are not shown in the figure exhibit the same behavior. We determine that the accuracy error ($AccuracyError$) can be characterized as follows:

$$AccuracyError = a \times Z^b + c, \quad (2)$$

where Z is the frame size and a, b, and c are constants. Interestingly, the *same model* but with different constant values can be used to represent the distortion. This model is referred to as "Model" in Figure 2. The frame size Z can be calculated as $Z = R/\tau$, where R is the video playback rate and τ is the video frame rate.

The accuracy-based formulation in Equation (1a) uses the $AccuracyError$ shown in Equation (2). The distortion model will be used to compare distortion-based optimization (the old approach proposed in [9]) with the accuracy-based optimization proposed in this paper.



(a) CMU/MIT (b) Georgia Tech at 50% Res.

Fig. 2: Rate-Accuracy Characterization for MJPEG

C. Cross-Layer Optimization Solution

After determining the accuracy error function, we can now solve the problem formulated in Equation (1). Since the problem meets the conditions of budget constrained convex programming problems, it can be solved using Lagrangian

relaxation. Assuming that all video sources have the same b_s, which is empirically valid, we get the following solution:

$$f_s^* = \left(\frac{-\lambda^* \tau_s}{w_s a_s b_s y_s (y_s/\tau_s)^{(b_s-1)}}\right)^{(1/(b_s-1))}, \quad (3)$$

where

$$\lambda^* = \left(\frac{A_{eff}}{\sum_{s=1}^{S}\left(\frac{-\tau_s}{w_S a_s b_s y_s (y_s/\tau_s)^{(b_s-1)}}\right)^{(1/(b_s-1))}}\right)^{(b_s-1)}. \quad (4)$$

Therefore, the AP determines λ^* using Equation (4) and sends it to all video sources in the network using the beacon packet. When a video source s receives λ^*, it determines its fraction of the airtime f_s^* using Equation (4) and changes the application data rate (i.e., video encoding rate) according to the equation: $r_s^* = f_s^* \times y_s$.

The optimization results can be enforced by the link layer through changing the TXOP limit. As in [4], each video source determines its TXOP limit as the time required to transmit the packets that belong to a single video frame along with all associated overhead.

III. Performance Evaluation Methodology

We use OPNET to simulate the network using a realistic MJPEG video traffic. We implement a traffic source in OPNET that streams MJPEG videos as Real-time Transport Protocol (RTP) packets. Moreover, we implement a realistic video streaming client at the application layer of the proxy station. This client receives and reassembles the RTP packets from various video streams, and then carries out error concealment [10] to mitigate the impact of packet loss. The use of MJPEG enables the use of standard image datasets that are suitable for surveillance applications. We use the following standard image sets to assemble the video streams: CMU/MIT, Georgia Tech., and SCFace. The frames are chosen randomly from the dataset used. The Georgia Tech. image set produces a highly limited range of bitrates for the studied network. To produce a better variety, we generate two image sets from the original Georgia Tech. image set by changing the resolution of each image in the set to 30% and 50% of the original resolution, respectively. At each video source, the streamer takes a bitrate, a frame rate, and an image set as inputs and produces a corresponding MJPEG video stream. We assume a frame rate of 20 fps.

We conduct experiments with networks of one AP, one proxy station co-located with the AP, and a variable number of video sources. The video sources start sending the video streams to the proxy station randomly within 1 second of the beginning of the simulation. After every 1 second, each video source sends its status update as a small packet, called *state report*, to the AP. The state report includes the physical rate, dropping rate (buffer + retransmission), and possibly local accuracy error model parameters. These data will be utilized by the AP to execute the effective airtime estimation algorithm and to determine λ in the optimization solution. Table I summarizes the main simulation parameters.

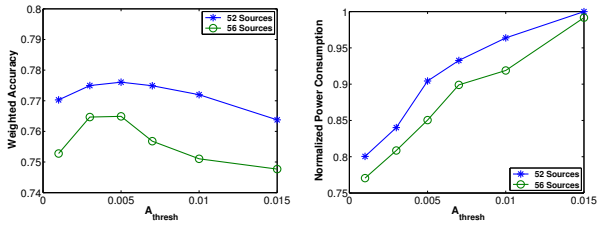
We compare the proposed accuracy-based optimization solution, referred to in the results as *Weighted Accuracy Optimization* (WAO), with the following two solutions. (1)

TABLE I: Summary of Simulation Parameters

Parameter	**Model/Value(s)**
# video sources	16-76
Simulation Time	10 min
Packet Size	1024 bytes
Application Rate	Optimized, Default = max physical rate / #sources
Video Frame Rate	20 frames/s
Physical characteristics	Extended Rate (802.11g)
Physical Data Rate	Random from {12,18,24,36,48,54} Mbps
Weight	Random from five levels between [0 1]
Buffer size	256 Kb
Video TXOP limit	Optimized, Default = 3008μs
Video CW	min = 15, max = 31
Retry Limit	short = 7, long = 4
Others	Beacon Interval = 0.02s State Report Interval = 1s

The cross-layer solution in [4], called *Enhanced Distortion Optimization* (EDO), which is the best performer among exiting solutions. (2) A hypothesized version of EDO that uses weights for various video sources, which is referred to here as *Weighted Distortion Optimization* (WDO). We analyze the following **performance metrics**: *Weighted Accuracy*, *Overall Network Load*, and *Power Consumption*. The weighted accuracy is the weighted detection accuracy of the video sources. The accuracy for each source is determined as the average accuracy of the received frames sent by that source. The accuracy of a dropped video frame is assumed to be 0. The overall network load is defined as the total load sent by the application layers of all video sources. Finally, power consumption is the average power consumption of the wireless interfaces of the video sources and is determined by using the power consumption model in [11].

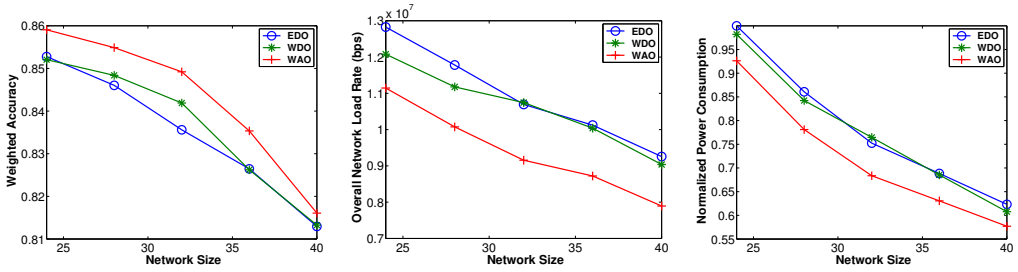
IV. Result Presentation and Analysis



(a) Weighted Accuracy (b) Power Consumption

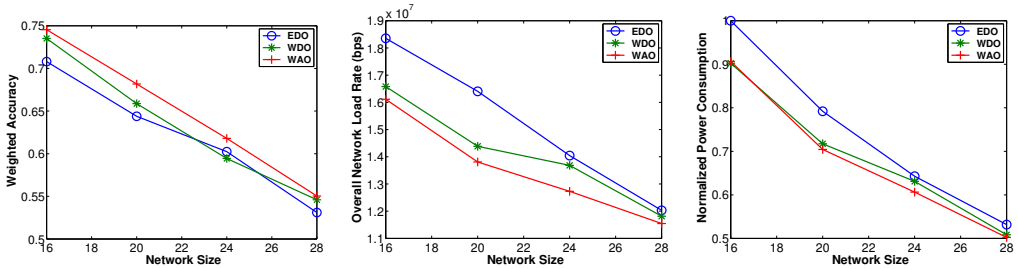
Fig. 3: Impact of A_{thresh} with Proposed Weighted Accuracy Optimization [MJPEG, Georgia Tech. at 30% Res.]

Let us start by analyzing the impact of A_{thresh} on the weighted accuracy and power consumption. As discussed earlier, this threshold controls the allowable packet dropping in the network. Figure 3 shows the results of running the proposed WAO solution for two different network sizes. The results demonstrate that the weighted accuracy improves with A_{thresh} up to a certain point and then starts to worsen. The peak happens when A_{thresh} is smaller than 0.01, suggesting that that the optimal accuracy is achieved when the dropping is very small. As expected, the power consumption increases with A_{thresh} because the sending rate increases with A_{thresh}. Therefore, A_{thresh} should be selected based on the proper tradeoff between accuracy and power consumption.



(a) Weighted Accuracy (b) Overall Network Load (c) Power Consumption

Fig. 4: Comparing Various Bandwidth Allocation Solutions [MJPEG, Georgia Tech. at 50% Resolution $A_{thresh} = 0.001$]



(a) Weighted Accuracy (b) Overall Network Load (c) Power Consumption

Fig. 5: Comparing the Effectiveness of Various Bandwidth Allocation Solutions [MJPEG, CMU/MIT, $A_{thresh} = 0.005$]

Figures 4 and 5 compare the effectiveness of the proposed cross-layer optimization solution (WAO) with EDO and WDO for the Georgia Tech. at 50% resolution and $A_{thresh} = 0.001$, and the CMU/MIT dataset and $A_{thresh} = 0.005$, respectively. The results with other combinations of datasets and A_{thresh} values exhibit the same behavior and thus are not shown. These results demonstrate that the proposed accuracy-based optimization solution outperforms other solutions in all metrics. In particular, it improves both the weighted accuracy and power consumption by up to 10%. The improvement in power consumption is due to reducing the application rates (network loads) of various video sources. In addition, the results indicate that incorporating weights for various video sources with the distortion-based optimization has no noticeable improvement. On the contrary, it may worsen the weighted accuracy in some cases. By detailed analysis of the results, we observe that applying weights in WDO forces the video sources with low importance factors to send the video streams at extremely low bitrates, and in some cases, it prevents these sources from sending any video, leading to unacceptable levels of accuracy for these sources.

V. Conclusions

The results show that the proposed cross-layer optimization solution significantly enhances both the detection accuracy and power consumption, compared to the distortion-based optimization. The reduction in power consumption is due to sending and dropping much less data.

References

[1] Qiang Chen, Zheng Song, Jian Dong, Zhongyang Huang, Yang Hua, and Shuicheng Yan, "Contextualizing object detection and classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 1, pp. 13–27, Jan 2015.

[2] Kai-Wen Cheng, Yie-Tarng Chen, and Wen-Hsien Fang, "Gaussian process regression-based video anomaly detection and localization with hierarchical feature representation," *IEEE Transactions on Image Processing,*, vol. 24, no. 12, pp. 5288–5301, Dec 2015.

[3] Cheng-Hsin Hsu and Mohamed Hefeeda, "A framework for cross-layer optimization of video streaming in wireless networks," *TOMCCAP*, vol. 7, no. 1, pp. 5, 2011.

[4] Mohammad A. Alsmirat and Nabil J. Sarhan, "Cross-layer optimization and effective airtime estimation for wireless video streaming," in *Computer Communications and Networks (ICCCN), 2012 21st Int'l Conf. on*, 30 2012-aug. 2 2012, pp. 1 –7.

[5] Pavel Korshunov and Wei Tsang Ooi, "Video quality for face detection, recognition, and tracking," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 7, no. 3, pp. 14:1–14:21, Sept. 2011.

[6] Yousef Sharrab and Nabil J. Sarhan, "Accuracy and power consumption tradeoffs in video rate adaptation for computer vision applications," in *Proc. of the IEEE Int'l Conf. on Multimedia & Expo (ICME 2012)*, Melbourne, Australia, July 2012, pp. 410 – 415.

[7] Ye Ge, Jennifer C. Hou, and Sunghyun Choi, "An analytic study of tuning systems parameters in IEEE 802.11e enhanced distributed channel access," *Computer Networks*, vol. 51, no. 8, pp. 1955–1980, 2007.

[8] Paul Viola and Michael Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2001, vol. 1, pp. 511–518.

[9] Mohammad A. Alsmirat and Nabil J. Sarhan, "Cross-Layer optimization and effective airtime estimation for wireless video streaming," in *21st Int'l Conf. on Computer Communications and Networks (ICCCN 2012)*, Munich, Germany, July 2012.

[10] Shahram Shirani, Faouzi Kossentini, Samir Kallel, and Rabab Ward, "Reconstruction of jpeg coded images in lossy packet networks," Technical Report, 1997.

[11] Yousef Sharrab and Nabil J. Sarhan, "Aggregate power consumption modeling of live video streaming systems," in *ACM Multimedia Systems (MMSys 2013)*, Oslo, Norway, February 2013.