University of Cincinnati

Date: 6/27/2019

I, Lingchao Kong, hereby submit this original work as part of the requirements for the degree of Doctor of Philosophy in Computer Science & Engineering.

It is entitled:

Modeling of Video Quality for Automatic Video Analysis and Its Applications in Wireless Camera Networks

Student's name: **Lingchao Kong**

This work and its defense approved by:

Committee chair: Rui Dai, Ph.D.

Committee member: Dharma Agrawal, D.Sc.

Committee member: H. Howard Fan, Ph.D.

Committee member: Carla Purdy, Ph.D.

Committee member: Julian Wang, Ph.D.

34239

Modeling of Video Quality for Automatic Video Analysis and Its Applications in Wireless Camera Networks

A Dissertation submitted to the

Graduate School

of the University of Cincinnati

in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

in the Department of Electrical Engineering and Computer Science

of the College of Engineering and Applied Science

by

Lingchao Kong

M.S. Harbin Institute of Technology, China

B.S. Wuhan University of Technology, China

June 2019

Dissertation advisor and Committee chair: Rui Dai, Ph.D.

Abstract

Wireless camera networks are ubiquitously deployed in various distributed sensing applications. The basic functions of each sensor node include video capture, video encoding or local video processing, and data transmission. The process of video analysis is implemented either in the central server or in the sensor node. Automatic video analysis can efficiently extract useful information from a huge amount of videos without human intervention. Object detection is the first and the most essential step of automatic video analysis. Thanks to abundant information provided by cameras and the development of computer vision techniques, automatic video analysis in wireless distributed systems is applied further. However, traditional network quality measures, such as QoS and QoE, do not necessarily reflect the quality of automatic video analysis in wireless camera networks. The overall goal of this dissertation is to propose new quality measures that could reflect the quality of automatic video analysis in wireless camera networks and to design efficient video processing and encoding schemes for wireless cameras that could boost the quality of automatic video analysis.

The impact of lossy compression on object detection is systematically investigated. It has been found that current standardized video encoding schemes cause temporal domain fluctuation for encoded blocks in stable background areas and spatial texture degradation for encoded blocks in dynamic foreground areas of a raw video, both of which degrade the accuracy of object detection. Two measures, the sum-of-absolute frame difference (SFD) and the degradation of texture (TXD), are introduced to depict the temporal domain fluctuation and the spatial texture degradation in an encoded video, respectively. A model of object detection quality on compressed videos is established based on these two measures. Then we have proposed an efficient video encoding framework for boosting the accuracy of object detection for distributed sensing applications. The proposed encoding framework is designed to suppress unnecessary temporal fluctuation in stable background areas and preserve spatial texture in dynamic foreground areas based on the two measures, and it introduces new mode decision strategies for both intra and inter frames to improve the accuracy of object detection while maintaining an acceptable rate-distortion performance.

Video analysis at network edges in a distributed manner can alleviate bandwidth pressure, enable better real-time response and achieve higher system reliability. We investigate the impact of imaging quality, such as noise and blur, on the performance of distributed in-network video analysis. We propose a no-reference regression model based on a bagging ensemble of regression trees to predict the accuracy of object detection using observable features in an image. Based on the estimation of detection performance, we propose a quality adjustment framework to provide satisfactory object detection performance on embedded cameras. Key components of the framework include a blind regression model for predicting the performance of object detection and two classifiers for determining the type of distortion in an image. The proposed framework achieves accurate estimations of both image quality and image distortion types with low computational complexity and it can effectively enhance the performance of object detection on embedded cameras.

Acknowledgements

I am indebted to my advisor, Professor Rui Dai, for her invaluable support, inspiration, encouragement, and guidance throughout the course of this dissertation research. It has been an honor to be her Ph.D. student and to work with her from Fargo to Cincinnati. Her extensive knowledge, enlightened direction and continuous help encourage me for my entire Ph.D. study and future career. I believe I would benefit from this precious experience in rest of my life.

I would like to thank other members of my dissertation committee. Professor Dharma Agrawal, Professor H. Howard Fan, Professor Julian Wang, and Professor Carla Purdy have closely supervised my dissertation research. Their patient help and constructive advice make this dissertation better.

I am grateful to all members of the Multimedia Networking and Computing (MNC) Laboratory for their assistance and collaboration in research as well as friendship. It has been a pleasure to work with all the talented people in the MNC Lab. I want to thank my friends who play badminton together. I really enjoy that good time. I also want to thank all my friends that I cannot enumerate during my life time in Cincinnati. Their help and support are everywhere in my everyday life.

In particular, I want to thank my parents, my elder sister, and my grandparents for their continual support and encouragement. I am very grateful to my girlfriend, Yuan Tian, for her continuous understanding, caring and love.

Contents

List of Abbreviations

adjR^2	adjusted R^2
AMR	Average Miss Rate
ANOVA	Analysis of Variance
CD	Configuration Distance
cTwS	combined TFRE with STPE scheme
DCT	Discrete Cosine Transform
FDA	Frame Detection Accuracy
FN	False Negative
FOV	Fields Of View
FP	False Positive
FR	Full-Reference
Gdir	gradient direction
Gmag	gradient magnitude
GMM	Gaussian Mixture Model
HEVC	High Efficiency Video Coding
HOG	Histogram of Oriented Gradients
HVS	Human Visual Vystem

IQA	Image Quality Assessment
KRCC	Kendall Rank Correlation Coefficient
Ku	kurtosis
LCC	Linear Correlation Coefficient
MAE	Mean Absolute Error
MB	Macroblock
MSE	Mean Squared Error
NIIRS	National Imagery Interpretability Ratings Scale
NR	No-Reference
NSS	Natural Scene Statistic
PSNR	Peak Signal-to-Noise Ratio
QoE	Quality-of-Experience
QoS	Quality-of-Service
QP	Quantization Parameter
RDO	Rate-Distortion Optimization
RFC	Reducing Flicker video Coding approach
rFDA	revised Frame Detection Accuracy
RMSE	Root-Mean-Squared Error
RR	Reduced-Reference
SFD	Sum-of-absolute Frame Difference
SFDA	Sequence Frame Detection Accuracy
SIT	Separable Integer 4×4 Transform
Sk	skewness

SROCC	Spearman Rank Order Correlation Coefficient
SSAC	Sum-of-absolute 15 SIT AC Coefficients
SSD	Sum of Squared Differences
SSIM	Structural Similarity index
STPE	Spatial-Texture-Preserved video Encoding scheme
SVM	Support Vector Machine
TFRE	Temporal-Fluctuation-Reduced video Encoding scheme
TP	True Positive
TXD	Degradation of Texture
$\mathrm{TXD}^{\mathrm{SIT}}$	Degradation of Texture in 2-D transform domain
VQA	Video Quality Assessment

List of Figures

List of Tables

List of Publications

1. Lingchao Kong, Ademola Ikusan, Rui Dai, Jingyi Zhu, and Dara Ros. A no-reference image quality model for object detection on embedded cameras. *International Journal of Multimedia Data Engineering and Management (IJMDEM)*, 10(1):22–39, 2019.

2. Lingchao Kong, Ademola Ikusan, Rui Dai, and Jingyi Zhu. Blind image quality prediction for object detection. In *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 216–221. IEEE, 2019.

3. Lingchao Kong and Rui Dai. Efficient video encoding for automatic video analysis in distributed wireless surveillance systems. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 14(3):72, 2018.

4. Lingchao Kong, Jingyi Zhu, Rui Dai, and Mohammad Nazmus Sadat. Impact of distributed caching on video streaming quality in information centric networks. In *2017 IEEE International Symposium on Multimedia (ISM)*, pages 399–402. IEEE, 2017.

5. Lingchao Kong and Rui Dai. Object-detection-based video compression for wireless surveillance systems. *IEEE MultiMedia*, 24(2):76–85, 2017.

6. Lingchao Kong and Rui Dai. Temporal-fluctuation-reduced video encoding for object detection in wireless surveillance systems. In *Multimedia (ISM), 2016 IEEE International Symposium on*, pages 126–132. IEEE, 2016.

7. Lingchao Kong, Rui Dai, and Yuchi Zhang. A new quality model for object detection using

compressed videos. In *Image Processing (ICIP), 2016 IEEE International Conference on*, pages 3797–3801. IEEE, 2016.

8. Mohammad Nazmus Sadat, Rui Dai, Lingchao Kong, and Jingyi Zhu. QoE-aware multi-source video streaming in content centric networks. *revised for journal publication*, April 2019.

9. Lingchao Kong, Ademola Ikusan, Rui Dai, and Dara Ros. An image quality adjustment framework for object detection on embedded cameras. *submitted for journal publication*, June 2019.

Chapter 1

Introduction

Wireless camera networks are ubiquitously deployed in various distributed sensing applications such as surveillance, traffic monitoring, and environmental monitoring. Automatic video analysis can efficiently extract useful information from a huge amount of videos collected by these sensing applications without human intervention. In current wireless camera networks, each sensor node is usually equipped with an embedded camera, a wireless transceiver, and a battery supply[10]. The basic functions of each node include video capture, video encoding or local video processing, and data transmission. A typical automatic video analysis in wireless camera networks includes the following stages: object detection, classification of objects, tracking, understanding and description of behaviors, and final human identification [11]. Object detection is the first and the most essential step of the entire procedure, because detecting objects provides a focus of attention for later processes such as tracking and behavior analysis. After detection, the goal of tracking is to estimate the states of the target in the subsequent frames, given the initial state (e.g., position and extent) of a target object in the first image. Behavior understanding involves the analysis and recognition of motion patterns, and the production of high-level description of actions and interactions. Final human identification can be treated as a special behavior-understanding task. Object detection and tracking could be done automatically, and understanding of behaviors and other high-level visual tasks are still application dependent and require much human/user involvement or feedback. Thanks to abundant multimedia information provided by camera and the development of computer vision and related techniques, automatic video analysis in wireless distributed systems is applied further and wider. However, traditional network quality measures, such as QoS (Quality-of-Service) and (Quality-of-Experience), do not necessarily reflect the quality of automatic video analysis in wireless camera networks. There is a lack of solutions for provisioning the quality of video analysis in wireless camera networks. **The overall goal of this dissertation is to propose new quality measures that could reflect the quality of automatic video analysis in wireless camera networks and to design efficient video processing and encoding schemes for wireless cameras that could boost the quality of automatic video analysis.**

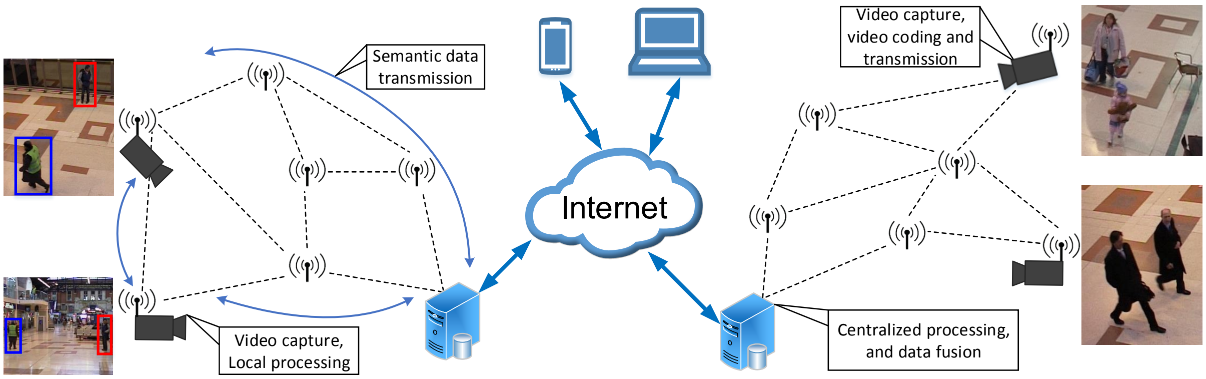

Semantic data transmission
Video capture, video coding and transmission
Internet
Video capture, Local processing
Centralized processing, and data fusion

Figure 1.1: Typical architecture of wireless camera networks

The typical architecture of wireless camera networks is shown as Figure 1.1. The process of video analysis for different applications is implemented either in the central server or in the camera sensor node, depending on their computational capability, energy supply and the purpose of applications. In the right implementation, raw videos acquired by camera sensors are usually preprocessed, encoded, and compressed before being delivered to the base station [12]. A powerful central server or a data center at the base station can fully utilize its powerful computing capability and perform data fusion from multiple cameras to obtain much better understanding of the surveillance videos than individual cameras [11, 13]. In the left implementation, wireless camera node can perform object detection and object tracking locally, and communicate semantic information with other nodes in real time for collaboration. In this dissertation, we address the modeling of video quality and its applications for both types of implementations.

1.1 Background

1.1.1 Image and video quality

Digital images are subject to a rich variety of distortions during acquisition, processing, compression, storage, transmission and reproduction, any of which may result in a degradation on image quality. Since human eyes usually are the ultimate receivers in most image related applications, the only correct method of quantifying visual image quality is through subjective evaluation. However, subjective evaluation is usually too inconvenient, time-consuming and expensive. Image Quality Assessment (IQA) is to develop quantitative measures that can objectively predict perceived image quality [14].

IQA can be classified into three categories according to the availability of an original image: 1). Full-Reference (FR), meaning that a complete reference image is assumed to be known; 2). Reduced-Reference (RR), meaning that reference image is only partially available in the form of a set of extracted features; 3). No-Reference (NR) or ”blind”, meaning that the reference image is not available, which is common in many practical applications.

Full-Reference IQA

The most widely used full-reference quality metric is the mean squared error (MSE), computed by averaging the squared intensity differences of distorted and reference image pixels, along with the related quantity of peak signal-to-noise ratio (PSNR). These have many attractive features because they are simple to calculate, have clear physical meanings, and are mathematically convenient in the context of optimization [15]. However, PSNR is not good representative for perceived visual quality.

Taking advantage of known characteristics of the human visual system (HVS) in recent years, Structural Similarity (SSIM) index [14] is proposed. Under the assumption that human visual perception is highly adapted for extracting structural information from a scene, the authors introduce an alternative complementary framework for quality assessment based on the degradation of structural information. The framework separates the task of similarity measurement into three independent comparisons: luminance, contrast and structure. The three components are calculated in the local windows first, then a mean SSIM index is used to evaluate the overall image quality. In summary, SSIM metric is simple to compute and consistent with perceptual quality.

Over several years, numerous variations of IQA algorithms which estimate quality based on structural similarity and/or structural degradation have been proposed. In [16], a complex wavelet SSIM version which adds robustness to small affine transformations of the distorted image is proposed. In [17], a feature similarity index is proposed based on the fact that HVS understands an image mainly according to its low-level features, which include the phase congruency and the image gradient magnitude. Another way to measure changes in structure is to compute changes in local image gradients. In [18], contraststructural changes, which can be effectively captured by gradient similarity, and luminance change in image are combined together to effectively assess image quality. In [19], the global variation of gradient magnitude similarity based local quality map for overall image quality prediction is proposed.

Many IQA algorithms share a common two-stage structure: local quality/distortion measurement followed by pooling. While significant progress has been made in measuring local image quality/distortion, the pooling stage is often done in straightforward ways. In [20], the information content weighting, which can be estimated in units of bit using advanced statistical models of natural images, for perceptual image quality assessment is explored. The visual saliency/ attention models can provide guides (weights) during pooling stag of IQA algorithms. In [21], an exhaustive statistical evaluation is conducted to justify the added value of computational saliency in objective image quality assessment, using 20 state-of-the-art saliency models and 12 best-known IQAs.

No-Reference and Reduced-Reference IQA

Although FR IQA provides a useful and effective way to evaluate quality differences, in many applications the reference image is not available or only limited information is available. Thus RR and NR metrics are highly desirable.

In the application of RR IQA, for example, real-time video quality monitoring over multimedia communication networks, feature extractor is applied to the reference visual signal at the sender side first. Then, the extracted features are transmitted to the receiver as side information to evaluate the quality of the distorted signal [22, 23]. In [24], an RR IQA method based on a set of extracted statistical features, which consider both primary visual information and unpredictable uncertainty with negligible transmission overhead, from the perspective of screen content images visual perception is proposed.

Since natural images have strong statistical regularities across different visual content, natural scene statistic (NSS) models are used to capture those statistical properties for NR or blind IQA. In [25], a simple Bayesian inference model to predict image quality scores is proposed, given exacted features from NSS model of discrete cosine transform (DCT) coefficients. In [26], an effective no-reference quality assessment of contrast distorted images based on the principle of NSS using support vector regression is proposed.

Recently, deep learning techniques are also applied to NR IQA. In [27], the blind IQA is reorganized as a five-grade classification problem to facilitate learning the qualitative descriptions given by humans via deep learning, and then a quality pooling is applied to produce numerical outputs. In [28], a multi-task end-to-end optimized deep neural network, which consists of two sub-networks-a distortion identification network and a quality prediction network-sharing the early layers, for blind image quality assessment is proposed.

1.1.2 Automatic image and video analysis

Intelligent video surveillance have been substantially growing from practical needs in the past decade, being driven by a wide range of applications. This section summarizes the state-of-the-art, and various practical systems for intelligent video surveillance.

The prerequisites for intelligent video surveillance using a single camera include the following stages: moving objects detection, object tracking, understanding and description of behaviors, and object recognition and identification [11]. In order to extend the surveillance area and overcome occlusion, fusion of data from multiple cameras is needed. This fusion can involve all the above stages. We classify these stages by low level, middle level and high level. Among them, moving objects detection is the low level, objects tracking is middle level, understanding and description of behaviors, and object recognition and identification are high level.

Automatic video analysis techniques

Moving objects detection aims at segmenting regions corresponding to moving objects from the rest of an image. The process of moving objects detection usually involves environment modeling, motion segmentation, and object classification, which intersect each other during processing. The active construction and updating of environmental models are indispensable to visual surveillance. Motion segmentation in image sequences aims at detecting regions corresponding to moving objects such as vehicles and humans. At present, most segmentation methods use either temporal or spatial information in the image sequence. Three main categories includes temporal differencing, background subtraction and optical flow. Temporal differencing makes use of the pixel-wise differences between two or three consecutive frames in an image sequence to extract moving regions [29]. Background subtraction detects moving regions in an image by taking the difference between the current image and the reference background image in a pixel-by-pixel fashion. Gaussian Mixture Model (GMM) is a popular method [30]. Optical flow motion segmentation uses characteristics of flow vectors of moving objects over time to detect moving regions in an image sequence. However, most flow computation methods are computationally complex and very sensitive to noise, and cannot be applied to video streams in real time without specialized hardware [31]. Different moving regions may correspond to different moving targets in natural scenes. To further track objects and analyze their behaviors, it is essential to correctly classify moving objects.

After moving object detection, surveillance systems generally track moving objects from one frame to another in an image sequence. The tracking algorithms usually have considerable intersection with motion detection during processing. Tracking methods are divided into four major categories: region-based tracking, active-contour-based tracking, featurebased tracking, and model-based tracking. Tracking over time typically involves matching objects in consecutive frames using features such as points, lines, blobs or Histogram of Oriented Gradients (HOG) [32]. Useful methods for tracking include the Kalman filter [33], the hidden Markov models [34], Condensation algorithm [35] and Particle filter[36].

On the other hand, collaborative object tracking using multiple-camera is also explored in some recently works. For example, a new collaborative tracking algorithm is put forward to track multiple objects in video streams in [37]. A framework of a collaborative multiple-camera tracking system for seamlessly object tracking across fixed cameras in overlapping and non-overlapping fields of view (FOVs) is presented in [38]. An approach to track several subjects from video sequences acquired by multiple cameras in real time is presented in [39]. Pedestrian re-identification is a difficult problem due to the large variations in a person's appearance caused by different poses and viewpoints, illumination changes, and occlusions. A new approach that takes the video of a walking person as input and builds a spatiotemporal appearance representation for pedestrian re-identification is proposed in [40].

After successfully tracking the moving objects from one frame to another in an image sequence, the problem of understanding-object behaviors from image sequences follows naturally. Behavior understanding involves the analysis and recognition of motion patterns, and the production of high-level description of actions and interactions. Dynamic time warping [41], finite-state machine [42] and hidden Markov model [43] are useful methods. Recognition and identification of objects, such as human beings, vehicles, or animals, can be treated as a special behavior-understanding problem. Human face and gait are now regarded as the main biometric features that can be used for personal identification in visual surveillance systems. Bags-of-keypoints model [44] and deep learning model [45] are most popular recognition methods in recent years.

Distributed embedded cameras

Most of vision-computing techniques require powerful processors and large memory resources if they are not customized for implementation in distributed smart camera platforms [46]. Most of the platforms utilize background subtraction, frame differencing, and edge detection as the main in-network processing techniques [47]. For example, Meerkats [48], Cyclops [49], MeshEye [50], and MicrelEye [51] perform background subtraction and frame differencing as object detection mechanisms. MicrelEye takes a fixed background frame at the beginning and then performs pixel-by-pixel differencing between each frame and the background frame. Cyclops use moving average filters to smooth background changes. Firefly Mosaic [52] uses a Gaussian Mixture Model (GMM) to separate foreground from background, and it also support face detection function. Edges could be considered as boundaries between dissimilar regions in an image. Computation of edges are fairly cheap and recognition of an object is easy since it provides strong visual clues. Cyclops supports Sobel libraries to perform edge detection. Delay incurred by edge detection as a function of image size is investigated for XYZ platform [53].

In [54], CITRIC, a distributed smart camera platform, is introduced and used to perform distributed object recognition. A wireless embedded smart-camera system is presented for cooperative object tracking and detection of composite events spanning multiple camera views in [55]. Each camera is a CITRIC mote consisting of a camera board and a wireless mote. Lightweight and robust foreground detection and tracking algorithms are implemented on the camera boards. In [56], adaptive methodologies are introduced for energy-efficient object detection and tracking with battery-powered embedded smart cameras, which is also based on CITRIC platform.

There are many commercial applications of intelligent video surveillance. UTC Fire Safety and Security (former GE Security) offers integrated security management, intrusion and property protection, and video surveillance [57]. IBM smart surveillance system (S3) can provide not only the capability to automatically monitor a scene but also the capability to manage the surveillance data, perform event based retrieval, receive real time event alerts through standard web infrastructure and extract long term statistical patterns of activity [58]. FLIR Intelligent Transportation

Systems (ITS) provide unique, field-proven solutions help keep vehicles, pedestrians, and bicycles moving safely and smoothly [59]. For example, the TrafiCam series combines a CMOS camera and a video detector in one, which is used for detection and monitoring of moving and stationary vehicles at signalized intersections. C-Walk / SafeWalk product can improve safety and efficiency at signalized intersections and pedestrian crossings. AID Boards product can integrate automatic incident detection, data collection, recording of pre and post incident image sequences and streaming video in one board. DAHUA security can provide illegal behavior detecting and recording in road intersection [60].

1.1.3 Image and video quality for automatic analysis

The use of video and image analytics to automatically detect and track object, identify activities and anomalous behaviors is very important. The accuracy of video and image analytics depends on the input video quality. However, the video quality evaluated by automatic analysis may be different with the perceptual quality perceived by human being.

Quality of automatic analysis on compressed video

In [61], one method of predicting accuracy for pedestrian detection on compressed video without actually executing detection is proposed. The video dataset is transcoded 18 times, each time using a unique quantization parameter (QP), which ranges from 15 to 55. The accuracy prediction has three components: texture descriptor extraction, spatial and temporal averaging and random forest regression. The spatial texture descriptors are used to capture video quality degradation caused by video compression. In [62], a system to predict videos optimal compression rate for the task of activity recognition is proposed. Videos are encoded into five different compressed versions (five constant QPs: 20, 26, 32, 38, 44) in H.264 format. The system starts with feature extraction from all compressed versions of the input video, and then applies the visual word assignment pipeline to assign words to each descriptor. The resulting histogram represents the video. After that, the histogram is input to the trained Random Forest to receive a classification result on whether the detection performance is success or failure for the given QP value. The final step is to collect the classification results from the previous step and select the optimal QP.

In [63], an importance map that can guide bit allocation to areas that are important for object detection for HEVC video encoding standard is created by using the initial convolutional layers of a state-of-the-art object detector. Similarly in [64], the impact of both near-lossless and lossy compression of feature data using HEVC standard on collaborative object detection is studied. The relationship between human, object detection, and vision related application accuracy and video coding rate mainly controlled by QP is explored and applied in rate allocation for video analysis in mobile surveillance networks [65], in joint source and channel coding for human detections in a mobile surveillance cloud [66], in license plate recognition and medical diagnosis [67], and in edge computing for cooperative video processing in multimedia IoT systems [68].

In summary, all these works focus on exploring the quality of video analysis on compressed video and they try to find the optimal compression rate for multimedia communication.

Quality of automatic analysis without compression

In [69], a detection probability model to estimate the quality of target detections by integrating the target location uncertainty over polygonal domains, which represent the fields of view (FOV) of the cameras, is proposed. Quadrature-based integration is combined with importance sampling to provide accurate quality estimations while reducing the computational cost. This detection probability models the miss-detection rate and accounts, over time, for the number of undetected targets that are within the fields of view FOV. In [70], image quality assessment for face recognition is investigated. First, a number of techniques that measure image quality factors namely, contrast, brightness, focus, sharpness, and illumination, are evaluated. Second, via a set of experiments measuring the sensitivity of each matric to quality change, the most practical measure(s) for each quality factor are selected. Finally, a novel face image quality index that combines the five aforementioned quality factors is proposed.

In [71], the impact of common image distortions on infrared face recognition is explored, Natural Scene Statistics (NSS) is used to detect degradation of infrared images, and a method for aggregating perceptual quality-aware features to improve the identification rates is proposed. In [72], the degradation in performance of face detector is quantified when human-perceived image quality is degraded by distortions commonly occurring in capture, storage, and transmission of facial images, including noise, blur, and compression. In [73], a framework for predicting the performance of a vision algorithm given the input image or video so as to maximize the algorithms ability to provide the desired output is proposed. The input image/video quality is measured by a combination of objective image quality measures.

There are a series of works about video and image quality of target detection, target tracking and event detection for airborne reconnaissance applications by the same group from Charles Stark Draper Laboratory. In [74], the applicability of the National Imagery Interpretability Ratings Scale (NIIRS) to an automated target detection algorithm is examined. The findings indicate that NIIRS is not a good predictor of target detection performance. In [75] and [76], the impacts of video frame rate and two spatial factor (noise and spatial resolution) on the tracker performance are investigated, respectively. In [77] and [78], the video quality models, which rest on a suite of image metrics computed in real-time from the videos, are proposed for enhanced event detection.

Salient object detection or saliency estimation for common detection is also investigated in [79, 80, 81]. For example, a novel quality constrained co-saliency estimation method for common detection is proposed in [79]. A quality-based dynamic feature selection method for improving salient object detection is introduced in [80]. A quality metric to measure the focus of the detected object in quad-copters is proposed in [81].

In general, automatic video and image analysis techniques have the ability to provide a successful output dependent on the input characteristics. Unfortunately, the acceptable input variability for analysis techniques is not known a priori. Recent investigations have shown that human observers and automated processing methods are sensitive to different aspects of image quality. The Detection quality model in [69] is the probability of a target to be detected within the FOV of a camera without considering the impact of actual image or video captured quality on detection performance. The proposed IQAs in [70, 71, 72] are specified to face recognition and face detection. The series of works in [74, 75, 76, 77, 78] are specified to video and image quality for airborne reconnaissance applications. The quality model in [73] is realized for object tracking, however, it does not consider the comprehensive image and video distortions. These literatures either propose some specific video and image quality models for certain applications, such as, face detection, recognition and airborne reconnaissance, or some video quality models for tracking and event detection. There are still lack of general image and video quality models on comprehensive distortions for automatic video and image analysis.

1.2 Overview of research objectives

The impact of video quality on object detection and its applications in wireless camera networks are systematically investigated in this dissertation.

Firstly, in many distributed wireless surveillance applications, compressed videos are used for performing automatic video analysis tasks. The accuracy of object detection, which is essential for various video analysis tasks, can be reduced due to video quality degradation caused by lossy compression. We introduces a video encoding framework with the objective of boosting the accuracy of object detection for wireless surveillance applications. The proposed video encoding framework is based on systematic investigation of the effects of lossy compression on object detection. It has been found that current standardized video encoding schemes cause temporal domain fluctuation for encoded blocks in stable background areas and spatial texture degradation for encoded blocks in dynamic foreground areas of a raw video, both of which degrade the accuracy of object detection. Two measures, the sum-of-absolute frame difference (SFD) and the degradation of texture in 2-D transform domain (TXD), are introduced to depict the temporal domain fluctuation and the spatial texture degradation in an encoded video, respectively. The proposed encoding framework is designed to suppress unnecessary temporal fluctuation in stable background areas and preserve spatial texture in dynamic foreground areas based on the two measures, and it introduces new mode decision strategies for both intra and inter frames to improve the accuracy of object detection while maintaining an acceptable rate-distortion performance. The quality model for object detection on compressed video is introduced in Chapter 2, and the efficient video encoding for object detection is presented in Chapter 3.

Video analysis at network edges in a distributed manner can alleviate bandwidth pressure, enable better real-time response and achieve higher system reliability. We investigate the quality of distributed in-network video analysis. The accuracy of automatic analysis methods relies on the quality of images that are processed, which could be degraded by factors such as noise and blur during the imaging process. It is therefore essential to predict the quality of images as evaluated by automatic analysis algorithms and to understand the types of distortion in an image, such that measures could be taken to enhance image quality accordingly. We proposes a quality adjustment framework to provide satisfactory object detection performance on embedded cameras. Key components of the framework include a blind regression model for predicting the performance of object detection and two classifiers for determining the type of distortion in an image. A video data set is constructed that considers different factors related to quality degradation in the imaging process. The performances of common low-complexity object detection algorithms are obtained for the data set. Based on the data set and utilizing features that can be easily extracted from an image, a regression model and two classifiers are trained and tested. The proposed framework achieves accurate estimations of both image quality and image distortion types with low computational complexity and it can effectively enhance the performance of object detection on embedded cameras. The quality model of object detection for local processing is introduced in Chapter 4, and the quality adjustment framework to provide satisfactory object detection performance on embedded cameras is presented in Chapter 5.

Chapter 2

Modeling of object detection quality on compressed videos

Wireless embedded camera sensors are ubiquitously deployed in many distributed sensing applications such as surveillance, environmental monitoring, and remote health care. In many distributed wireless surveillance systems [47], camera sensors report their video observations to a central base station through wireless communication, the typical architecture is shown as Figure 2.1. Due to the low computing power and limited energy and bandwidth on embedded cameras, raw videos acquired by camera sensors are usually preprocessed, encoded, and compressed before being delivered to the base station [10]. A powerful central server or a data center at the base station can fully utilize its powerful computing capability and perform data fusion from multiple cameras to obtain much better understanding of the surveillance videos than individual cameras [11, 13]. A typical automatic surveillance system includes the following stages: object detection, classification of objects, tracking, understanding and description of behaviors, and final human identification [11]. Object detection is the first and the most essential step of the entire procedure, because detecting objects provides a focus of attention for later processes such as tracking and behavior analysis. While it is inevitable to introduce quality degradation in lossy compression, encoders should properly control video quality to maintain satisfactory performance for object detection.

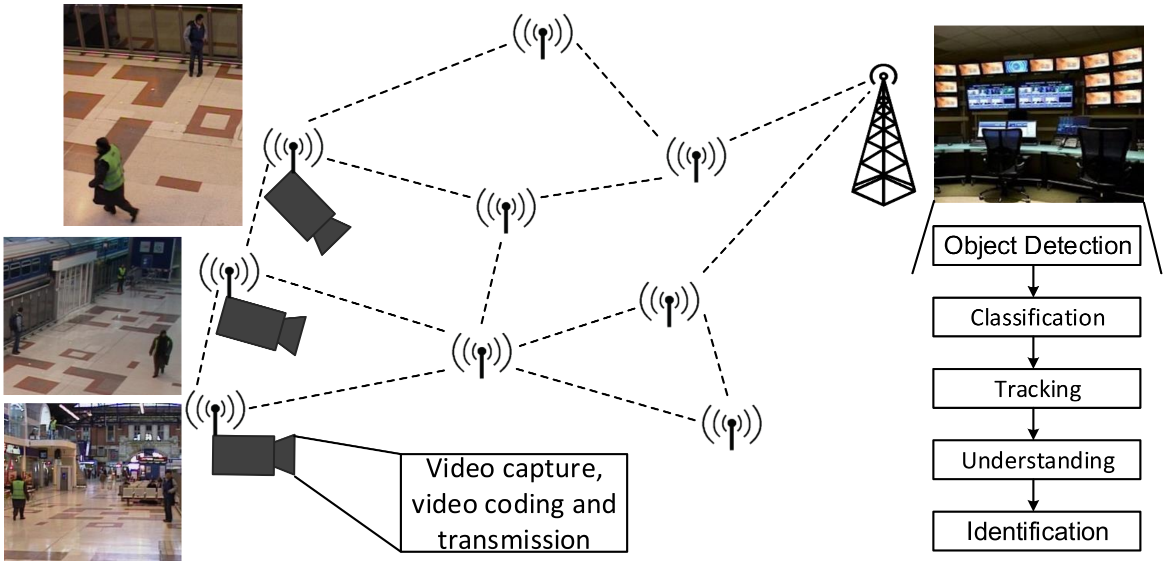


Figure 2.1: Typical architecture of distributed wireless surveillance systems

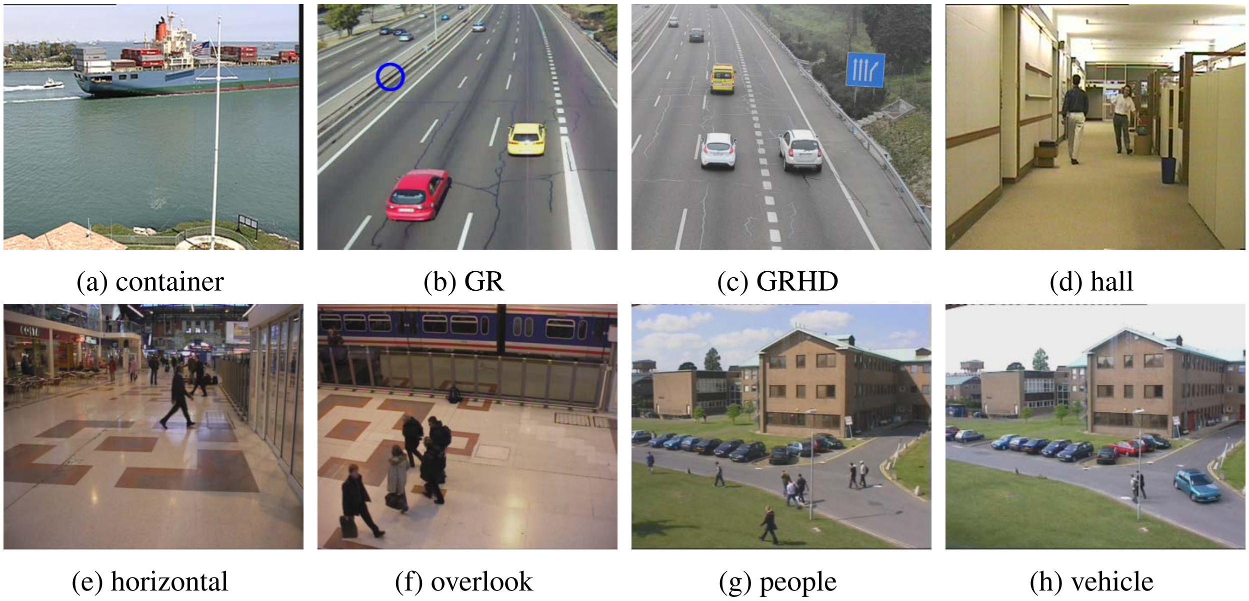
Various video quality assessment (VQA) measures have been applied to monitor and control video quality. The industrial standard measure Peak-Signal-to-Noise-Ratio (PSNR) characterizes the Mean Square Error (MSE) between a compressed image and an original image, which has been applied in rate-distortion optimization for various video encoders. A lot of VQA studies have aimed at modelling the quality of videos as perceived by human users, including the classical measures like Structural Similarity (SSIM) [14] and VQM [82], and several recent models such as MOVIE [83], STRRED [84], and V-CORNIA [85].

The video quality as judged by an automatic vision algorithm, however, is not necessarily sensitive to the same factors that drive human perceptions. Perceptual image quality assessment solutions usually emulate known characteristics of the human visual system (HVS), such as the contrast sensitivity and the visual attention mechanisms. The contrast sensitivity mechanism means that the HVS is sensitive to the relative luminance change rather than the absolute luminance change [14]. The visual attention mechanism is that only a local area in the image can be perceived with high resolution by the human observer at one time instance at typical viewing distances, due to the foveation feature of the HVS [86]. On the other hand, automatic analysis methods run by machines can "perceive" the absolute luminance change precisely and have a better global "view". For example, the problem of evaluating motion imagery quality for tracking in airborne reconnaissance systems was studied in [75]. It has been found that automated target detection algorithms are less sensitive to spatial resolution than humans, but factors such as jitter in the temporal domain, texture complexity, edge sharpness, and level of noise have a strong effect on the performance of target detection. The study in [75] focused on evaluating the quality of raw videos without distortion introduced by compression. It is also worthwhile to investigate how lossy compression can affect the accuracy of object detection. Studying this problem can provide insight on the design of encoders to improve the accuracy of object detection under rate constraints.

The goal of this chapter is to build a new video quality model to estimate the performance object detection using parameters that could be easily obtained in the encoding process, based on which video encoders could be adjusted to enhance the accuracy of object detection. To achieve this goal, we establish a distorted video database by applying several object detection algorithms on a variety of videos that are encoded using different parameters. From statistical analysis results, we construct a parametric model to predict the accuracy of object detection on compressed videos in relative to uncompressed raw videos.

2.1 Description of dataset and tests

A distorted video database was constructed to study the impact of lossy compression on the performance of object detection. Eight video sequences with different spatial and temporal details were chosen. The snapshots of these videos are shown in Figure 2.2. Among them, *container*, *GR* and *GRHD* [87] are typical test videos for traffic monitoring, *hall*, *horizontal* and *overlook* [88] are indoor scenes, and *people* and *vehicle* [89] are outdoor scenes. The open source H.264/AVC encoder x264 [90] was used to compress the raw videos, and compression settings are summarized in Table 2.1. Each raw video was compressed using 19 different QPs ranging from 22 to 40, which resulted in a total number of 152 compressed videos.


(a) container (b) GR (c) GRHD (d) hall

(e) horizontal (f) overlook (g) people (h) vehicle

Figure 2.2: Snapshots of video sequences

Table 2.1: Video compression parameters

Encoder	x264	Resolution	CIF (352×288)
Frame rate	25	Duration	20 sec
GOP structure	IPPP	GOP size	20
Rate control	constant QP	QP range	22-40

Object detection algorithms can be classified into two main groups: optical flow and background subtraction [91, 11]. Background-subtraction-based object detection algorithms attract the most attention due to their high accuracy and moderate complexity. As suggested in [92], background subtraction algorithms can be summarized into several categories based on their principles. Three algorithms from different categories were selected to be executed on the compressed videos: the GMM algorithm [93] from the statistical category, the GMG algorithm [94] from the non-parametric category, and the ABL algorithm [92] from the basic category.

It is unlikely to have prior knowledge on what object detection algorithm will be used by a certain application. Therefore, it is hard to estimate the absolute accuracy of object detection at the encoder side. We propose to estimate the relative performance of object detection on compressed videos in comparison to uncompressed raw videos. Object detection results from the raw videos are regarded as *ground truth*, and results from the compressed videos are *algorithm result*. *Recall* and *Precision* are common metrics to evaluate the performance of object detection [95]. *Recall* denotes the percentage of detected true positive pixels compared with the total number of true positive pixels in the *ground truth*, and *Precision* denotes the ratio of detected true positive pixels to the total number of pixels detected in the *algorithm result*, which are given by:

$$\begin{aligned} Recall &= \frac{TP}{TP + FN}, \\ Precision &= \frac{TP}{TP + FP}, \end{aligned} \tag{2.1}$$

where *TP*, *FN*, *FP* stand for the amount of true positive pixels, false negative pixels, and false positive pixels, respectively. Since *Recall* and *Precision* selectively assess the level of missing TP and mistaking TP, it is hard to evaluate the performance of algorithms using one of these metrics alone. Therefore, the overall performance of detection algorithms could be measured by their harmonic mean F_1 [96], which is given by:

$$F_1 = 2 \cdot \frac{Recall \cdot Precision}{Recall + Precision}. \tag{2.2}$$

2.2 Proposed quality model

A typical automatic vision system includes the following stages: object detection, classification of objects, tracking, understanding and description of behaviors, and final human identification [11]. Object detection is the first and the most essential step of the entire procedure, because detecting object provides a focus of attention for later processes such as tracking and behavior analysis. Based on the definition of *Recall* and *Precision*, the accuracy of object detection is related to TP, FN, and FP, and the union of TP and FN is the *ground truth*, a constant value if given the raw video and the detection algorithm. Therefore, it is sufficient to characterize the relative object detection performance of a compressed video by estimating the average values of FP and FN over different detection algorithms. To enable such estimation, in the following analysis, we consider

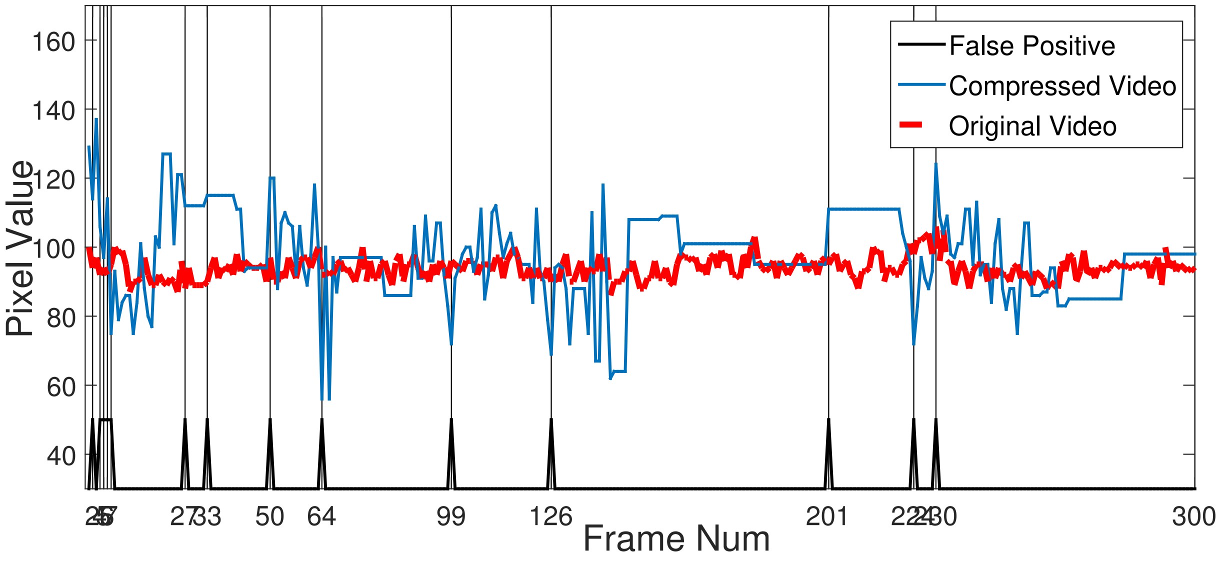


Figure 2.3: An example of temporal fluctuation in background

the scenario that a coarse-grained classification of foreground and background MBs can be done in the encoder side. A simple frame differencing method was applied before encoding to label coarse-grained foreground and background MBs in our distorted video database.

2.2.1 Estimating false positive

Unlike human beings that can easily extract and focus on a moving object from a blurred background, the performance of computer vision algorithms can be affected by the quality of the background. The background should be stable in the temporal domain to facilitate object detection; however, the procedure of video coding might introduce temporal fluctuations of the background that can cause FP. We take the traffic video *GR* as an example to demonstrate the variation of a specific pixel in the background. The center of the blue circle in Figure 2.2 (b) indicates the location of this background pixel. Figure 2.3 shows the intensity of this pixel in the encoded video and the original video in the temporal domain. Figure 2.3 also displays the detection result for this pixel using impulses of black line, where 1 indicates FP and 0 is correct. There is not much fluctuation in the intensity of the pixel in the original video, but there are several sharp changes of the intensity in the encoded video. When the intensity in the encoded video fluctuates intensively,

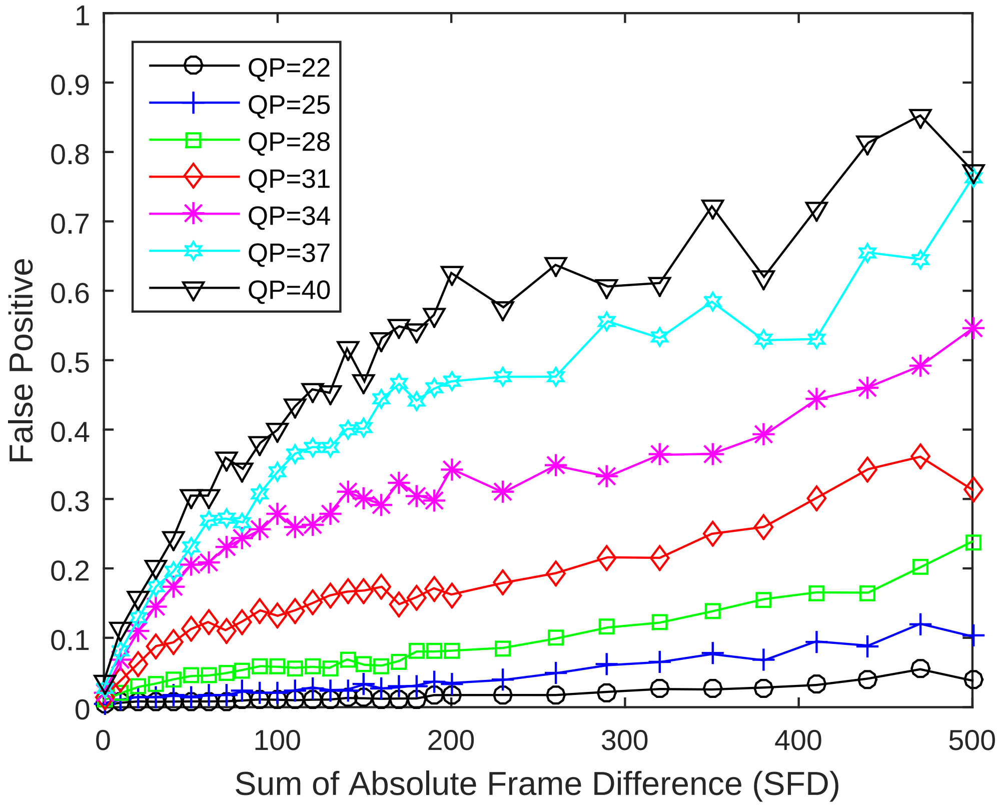


Figure 2.4: The trends of FP vs. SFD

it often causes FP. This situation occurs commonly in our distorted video database.

To describe the temporal fluctuation of a background, we introduce *SFD*, *the Sum-of-absolute Frame Difference in MB unit between the current frame and the previous frame*, which is given by:

$$SFD = \sum_{i,j=1}^{i,j=16} |m_t(i,j) - m_{t-1}(i,j)|, \tag{2.3}$$

where $m_t(i,j)$ is the reconstructed pixel value at location (i,j) in an MB of the current frame and $m_{t-1}(i,j)$ is the reconstructed pixel value at the same location in the corresponding MB of the previous frame.

To establish statistical relationship between SFD and FP, the FP and SFD values were recorded for constant background MBs in our entire dataset. The values of FP were obtained using the aforementioned three different object detection algorithms. Analysis of Variance (ANOVA) was conducted to the pairs of FP and SFD, where a small p-value ($p \leq 0.01$) means significant correla-

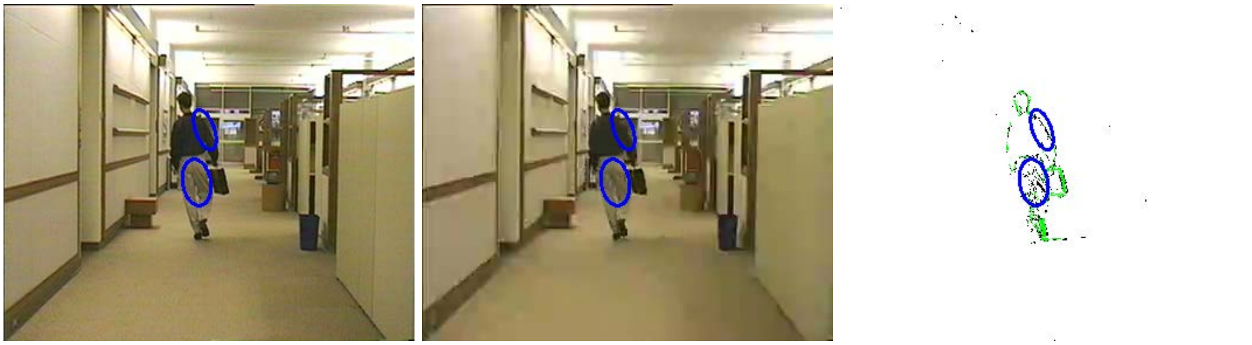


Figure 2.5: An example of texture degradation

tion [97]. The resulting p-values are very close to 0 and much smaller than 0.01, indicating that FP is closely associated with SFD. For each object detection algorithm, the average value of FP over all the background MBs were normalized based on $x' = (x - min(x))/(max(x) - min(x))$. The relationships between FP and SFD for the three detection algorithms are similar, and the averages for the three algorithms are shown in Figure 2.4. To provide a better view, this figure only displays the results under 7 different QPs.

From Figure 2.4, we can find that when SFD increases, FP also grows, and the curve looks like a power function. Moreover, FP goes up quickly when QP becomes large, indicating that higher compression can introduce more artifacts. Base on these observations, FP could be estimated by:

$$FP = a \cdot SFD^b, \tag{2.4}$$

where *a* and *b* are function parameters related to QP. *a* and *b* are quadratic functions of ***QP*** as $a = p_0 + p_1QP + p_2QP^2$, $b = p_3 + p_4QP + p_5QP^2$, and the detailed value of the parameters can be found in Table 2.4.

2.2.2 Estimating false negative

Edge and texture are the key elements for object detection. If there is no clear boundary between the foreground and the background, it is difficult to detect an object accurately. After comparing

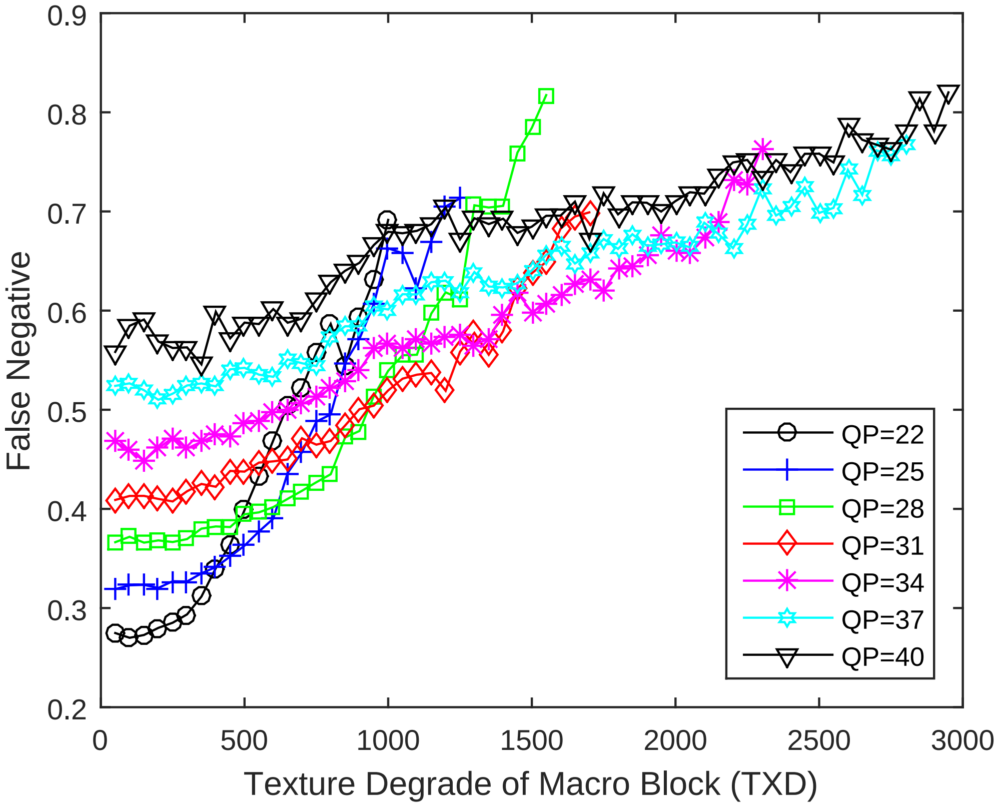


Figure 2.6: The trends of FN vs. TXD

algorithm results with *ground truth* in our entire dataset, we find that foreground areas with large texture deterioration are highly likely to be detected as FN. As an example, Figure 2.5 presents detection results for the 69^{th} frame of the *hall* sequence at 65 kbps. Figure 2.5 (a) and (b) are snapshots of the original frame and the encoded frame. In Figure 2.5 (c), the TP points are marked in green and the FN points in black. The blue ellipses on the shoulder and legs of the person indicate the locations where texture details are largely lost, and there are considerable FN points in these areas.

Based on this phenomenon, we introduce *TXD*, *the absolute difference of texture in MB unit between the encoded frame and the original frame*, to describe texture degradation as:

$$TXD = |\sum_{i,j=1}^{i,j=16} g_t(i,j) - \sum_{i,j=0}^{i,j=16} G_t(i,j)|, \tag{2.5}$$

where $\sum_{i,j=0}^{i,j=16} g_t(i,j)$ is the texture value in an MB of the encoded frame and $\sum_{i,j=0}^{i,j=16} G_t(i,j)$ is

Table 2.2: Goodness of fitting the model parameters

	a	*b*	*c*	*d*	*e*
adjR^2	0.939	0.881	0.9	0.983	0.968
RMSE	0.00557	0.0918	7.1e-08	2.0e-05	0.0162

the texture value in the corresponding MB of the original frame.

To obtain texture information, we have applied a simple texture analysis method that uses the range value of the 3-by-3 neighborhood around the corresponding pixel to represent the pixel's texture [98]. Values of TXD and FN were obtained for each foreground MB in our entire dataset.

ANOVA analysis was also conducted to the pairs of FN and TXD. The resulting p values are tiny numbers much smaller than 0.01, indicating that FN is significantly correlated with TXD. Similar with the post-process of FP vs. SFD, we took the average value of FN for a given TXD value, normalized the results for each algorithm, and obtained the average of the three algorithms. As shown in Figure 2.6 (b), FN increases when TXD grows, and the curve looks like a quadratic function. When QP becomes larger, the starting point of the curve is higher, and FN grows slower. In a small QP mode (high bit rate) the encoded video has less distortion, the overall level of FN is low and a little texture degradation would cause a relatively large increase of FN. Whereas in a large QP mode (low bit rate) the encoded video has higher distortion, the overall degree of FN is high and the growth ratio of FN is slow.

Based on these observations, FN can be estimated by:

$$FN = c \cdot TXD^2 + d \cdot TXD + e, \tag{2.6}$$

where *c*, *d*, and *e* are function parameters related to QP. *c* and *d* are cubic and quadratic functions of ***QP*** as $c = p_6 + p_7 QP + p_8 QP^2 + p_9 QP^3$, $d = p_{10} + p_{11} QP + p_{12} QP^2 + p_{13} QP^3 + p_{14} QP^4$, and *e* is a linear funciton of ***QP*** as $e = p_{15} + p_{16} QP$. The values of the parameters are listed in Table 2.4.

In summary, at the encoder end, using a simple frame differencing operation, we can easily

Table 2.3: Goodness of fitting in two models

	$FP = f(SFD, QP)$			
	max	min	average	std
adjR^2	0.973	0.866	0.94	0.0282
RMSE	0.0397	0.00429	0.0198	0.0109
	$FN = f(TXD, QP)$			
	max	min	average	std
adjR^2	0.995	0.915	0.971	0.0189
RMSE	0.0259	0.00526	0.0131	0.0054

Table 2.4: The list of model parameters

p_0	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8
5.55e-2	-6.13e-3	1.57e-4	4.42	-0.205	2.61e-3	-1.34e-5	1.45e-6	-4.94e-8
p_9	p_{10}	p_{11}	p_{12}	p_{13}	p_{14}	p_{15}	p_{16}	—
5.37e-10	8.91e-2	-1.13e-2	5.26e-4	-1.08e-5	8.15e-8	-5.53e-2	1.59e-2	—

label coarse-grained foreground and background MBs of the current frame. And then we can obtain SFD and TXD in real time, since the computing complexity of SFD is comparable with MSE used in RDO and the texture analysis for obtaining TXD is lightweight. An encoder could easily make use of the model to optimize the performance of object detection.

2.3 Performance evaluation

For performance evaluation, we first evaluated the accuracy of the model using adjusted R^2 (adjR^2) and Root-Mean-Squared Error (RMSE). We gathered adjR^2 and RMSE values for estimating the parameters in Eq. (2.4) and (2.6) (a, b, c, d, e) and for estimating FP and FN based on these parameters. A good curve fitting result is supposed to have adjR^2 close to one and RMSE close to zero. In Table 2.2, the results for estimating parameters a-e are listed, where all the adjR^2 values are fairly close to one and all the RMSE values are close to zero. The performance for estimating FP and FN was evaluated for each QP value in our dataset, and the results are shown in Table 2.3. The average values for adjR^2 are above 0.9, and the average RMSE values are close to 0.

Table 2.5: Correlation for the estimation of FP

Model	LCC	SROCC	KRCC
$f(SFD, QP)$	0.977	0.988	0.940
PSNR	-0.146	-0.136	-0.081
SSIM	-0.153	-0.134	-0.092

Table 2.6: Correlation for the estimation of FN

Model	LCC	SROCC	KRCC
$f(TXD, QP)$	0.971	0.972	0.885
PSNR	-0.451	-0.436	-0.318
SSIM	-0.493	-0.456	-0.333

The adjR^2 and RMSE values for different QPs are consistent, as indicated by the small standard deviation values in the table.

The performance of the proposed model was also compared with PSNR and SSIM, in terms of the Linear Correlation Coefficient (LCC), the Spearman Rank Order Correlation Coefficient (SROCC), and the Kendall Rank Correlation Coefficient (KRCC). The average results on our entire video dataset are shown in Table 2.5 and Table 2.6. The correlation values for both PSNR and SSIM are negative, and this is because higher PSNR or higher SSIM values can indicate better video quality, resulting in lower values of FP and FN. However, the absolute values of correlation can be used for comparison between the proposed method and PSNR/SSIM. PSNR and SSIM are both computed between an encoded frame and an original frame, but they could not capture the temporal fluctuation of reconstructed frames, which is a contributing factor for FP. This explains the low absolute correlation values between PSNR/SSIM and FP in Table 2.5. On the other hand, PSNR and SSIM could reflect the degradation of texture, which explains the relatively higher correlation between PSNR/SSIM and FN. The proposed model outperforms both PSNR and SSIM in predicting FP and FN. It can potentially be applied in encoders to achieve better performance in object detection on compressed videos.

2.4 Conclusion

In this chapter, we systematically investigate the effects of lossy compression on object detection, and propose a new quality model to predict the performance of object detection. Firstly, we introduce two features that are highly correlated with object detection performance: sum-of-absolute frame difference and texture degradation. Then, the parametric quality prediction model is built using these two features that can be easily obtained during the encoding process. Experimental results show that the model can achieve high accuracy in predicting the performance of object detection. The model introduces low computation cost and can be easily integrated in video encoders for rate-quality optimization.

Chapter 3

Efficient video encoding for object detection in wireless camera networks

3.1 Motivation

In Chapter 2, we have systematically investigated the effects of lossy compression on object detection, and proposed a new quality model to predict the performance of object detection using two features that can be easily obtained during the encoding process. The model introduces low computation cost and can be easily integrated in video encoders for rate-quality optimization. Since the inevitable degradation of video quality caused by lossy compression at embedded cameras has a significant impact on object detection [99, 100], video encoders for surveillance systems should be designed to improve the performance of object detection.

The block-based hybrid approach (intra-/inter-picture prediction and 2-D transform coding) is employed in all modern video compression standards such as H.264/AVC [101] and the latest HEVC (also known as H.265). As shown in Figure 3.1, this approach measures the encoding distortion by comparing the encoded video with the original video (*A* direction) using the metric *SSD*, namely *Sum of Squared Differences*, which is obtained by the sum of squared differences of the intensity between the encoded video and the original video in the macroblock (MB) unit.

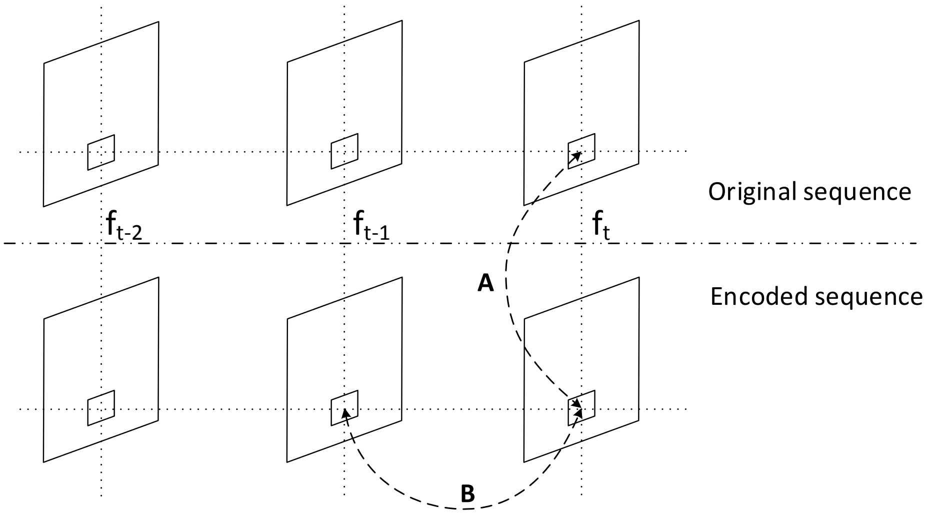


Figure 3.1: Schematic diagram of encoding distortion calculation

This strategy can result in two problems: 1) temporal domain fluctuation in the encoded video (*B* direction in Figure 3.1) when co-located regions of consecutive frames (e.g., f_{t-1} to f_t) are not consistently encoded, especially when intra frames are periodically inserted at low and medium bit rates; and 2) spatial texture degradation, since *SSD* could not effectively reflect the degradation status of spatial texture. In Chapter 2, we have studied the effects of lossy compression on object detection in depth, and we have found that the temporal domain fluctuation in stable background areas and the spatial texture degradation in dynamic foreground areas degrade the accuracy of object detection in a compressed video.

In this chapter, we propose an efficient video encoding framework for distributed wireless surveillance systems with the objective to improve the performance of object detection on compressed videos. The proposed framework uses the sum-of-absolute frame difference (SFD) to depict the temporal domain fluctuation, and the degradation of texture (TXD) to quantify the degree of spatial texture degradation in an encoded video. Both measures have been demonstrated to be highly correlated with the accuracy of object detection in Chapter 2. For the encoding of background areas in a raw video, we introduce a *Temporal-Fluctuation-Reduced video Encoding* scheme (TFRE) based on the SFD, and for the encoding of dynamic foreground areas, we introduce a *Spatial-Texture-Preserved video Encoding* scheme (STPE) based on the TXD in 2-D transform domain (TXD^{SIT}). Both schemes are standard-compliant, in which new mode decision strategies are incorporated in the standardized encoding procedure to optimize the performance of object detection. Our preliminary results on the TFRE scheme have been presented in our recent work [6, 5]. Unique contributions of this chapter includes: 1) the STPE scheme is designed based on a new spatial textual descriptor in the 2-D transform domain, which is presented for the first time in this chapter; 2) the STPE scheme is integrated with the TFRE scheme to a standard-compliant video encoding framework; 3) in addition to the original dataset in our preliminary work, a new dataset is introduced to evaluate the proposed algorithms; 4) using both the original dataset and the new dataset, the performance of the proposed algorithms are thoroughly evaluated in terms of computational complexity, pixel level detection accuracy, and object level detection accuracy.

There exist several encoding algorithms especially designed for improving the performance of object detection. In [102], regions of individual frames containing high-frequency spatial features, corners, and edges, which are detected by FAST and Sobel detectors, are preserved while other regions are smoothed in the encoding process. For efficient video processing and analysis in the compressed domain, a coding method is proposed that optimizes the accuracy of motion information embedded in a code stream based on the affine motion model [103]. In [67], two typical usage of task-based video, license plate recognition and medical diagnosis, are studied, and a task-based video quality optimization approach is proposed, which is driven by object recognition rates during encoding process. A model of human detection accuracy based on object area and video compression ratio is established in [65], and based on this model, an appropriate amount of bit rate is allocated to each moving camera in mobile surveillance networks. Although these existing encoding algorithms could improve the performance of object detection, they have not addressed the problem of temporal fluctuation in background areas that can reduce the accuracy of object detection.

On the other hand, the problem of temporal fluctuation has been investigated with the objective to improve the perceptual quality of compressed videos. The temporal fluctuation perceived by human is defined as *flicker*, which usually refers to frequent luminance or chrominance perceptual changes that does not appear in uncompressed raw videos [104]. A temporal low-pass filtering scheme is proposed that smooths the luminance changes on a block-by-block basis in [104]. A two-pass coding scheme is proposed in [105], which involves a first pass of simplified P-frame coding to derive a no-flicker reference of the current frame, and a second pass of actual I-frame coding with small QPs for closely approaching the no-flicker reference. A modified distortion measure that considers the distortions in both A and B directions in Figure 3.1 to reduce flicker is applied during intra prediction mode rate distortion optimized selection process in [106]. For the flicker artifact in HEVC, a region-classification-based rate control for Coding Tree Units in I-frames is proposed to improve the reconstructed quality of I-frames to suppress flicker in [107]. Different from these methods that are designed to optimize human visual perception, our proposed work addresses the temporal fluctuation problem to improve the performance of object detection. It is worthwhile to address this problem since the human vision system and the computer vision system may have different responses to an encoded video.

The conservation of spatial texture has been studied in several video encoding solutions. A region-based rate control scheme for better subjective quality is proposed in [108], in which each frame is firstly divided into complex textural regions, flat regions and moving regions, based on their inter-frame rate-distortion behaviors. Then, the regions containing complex texture are treated as one basic unit for rate control. In [109], a perspective motion model is employed to warp static textures and utilize texture synthesis to encode dynamic textures, which results in bitrate savings at the same video quality. For facilitating visual retrieval, textural features in spatial domain, such as gradient-based features like SIFT and SURF are better preserved by designing specific rate control strategies in [110]. A HEVC framework of jointly compressing the visual feature descriptors and video content is proposed for visual retrieval in [111], in which the high efficiency coding is achieved by exploiting the interactions between video features and visual content. While the purposes of these methods are to improve either objective and subjective quality or the performance of visual retrieval, our proposed work utilize the relationship between spatial texture and the 2-

D transform encoding to preserve spatial texture for the better performance of automatic video analysis.

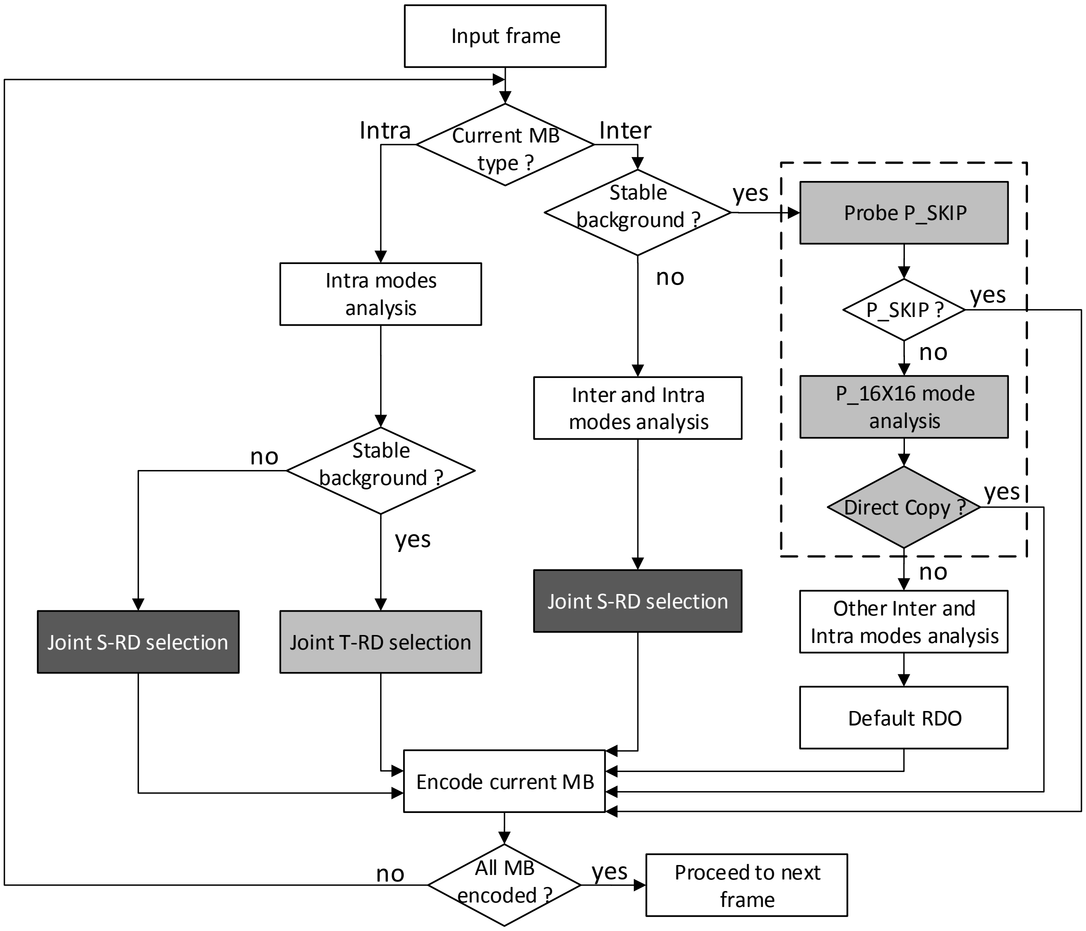


Figure 3.2: Flow chart of proposed video encoding framework

3.2 Proposed video encoding framework

To obtain better performance of object detection on compressed videos, we propose an efficient video encoding framework for distributed wireless surveillance systems. This framework includes a *Temporal-Fluctuation-Reduced video Encoding* scheme (TFRE) for the encoding of stable background areas and a *Spatial-Texture-Preserved video Encoding* scheme (STPE) for the encoding of dynamic foreground areas. Both schemes are designed to comply with the hybrid block-based video encoding architecture, in which new mode decision and Rate-Distortion Optimization (RDO) strategies are applied for intra and inter frames. The current implementation of this framework is based on the H.264/AVC standard. We consider the case that a coarse-grain classification of dynamic foreground and stable background MBs is obtained by a simple frame-differencing-based method at the encoder. The entire process for the proposed video encoding framework is illustrated in the flow chart in Figure 3.2, which includes two main branches, Intra and Inter MB encoding processes, for encoding the current MB of an input frame. Depending on whether the current MB is a stable background block or a dynamic foreground block, the TFRE scheme (gray color blocks) or the STPE scheme (black color blocks) is applied.

3.2.1 Temporal-fluctuation-reduced video encoding scheme

The TFRE scheme is designed to encode stable background areas to suppress unnecessary temporal fluctuation in these areas. For stable background MBs in intra frames, during the RDO process for deciding type mode and prediction mode, SFD is calculated and jointly optimized with the RDO cost. For the inter frame analysis process, new strategies are introduced in the analysis of P_SKIP and P_16×16 type modes, which are highlighted in the dashed box in Figure 3.2, with the objective to reduce temporal fluctuation for inter blocks while maintaining acceptable distortion.

Intra frame coding/mode selection

The intra frame RDO process of H.264/AVC consists of two steps: type mode decision from I_16×16, I_8×8, I_4×4 and I_PCM based on RDO cost, and then prediction mode decision from nine prediction options, such as vertical prediction, horizontal prediction, etc., based on RDO cost. And the RDO cost C is calculated by:

$$C = D + \lambda \times R, \tag{3.1}$$

where D denotes the distortion of a candidate encoding option, R denotes the total bits of this option, and λ is the Lagrange multiplier that controls the trade-off of rate and distortion.

We formulate a joint Temporal-fluctuation and RD (joint T-RD) mode selection problem, as follows:

$$\begin{aligned} \text{Given :} \quad & \{M_i, C_i, SFD_i\}, \\ \text{Find :} \quad & M^*, \\ \text{Minimize :} \quad & C, \\ \text{Subject to :} \quad & SFD_i \leq SFD_{th}, \end{aligned} \tag{3.2}$$

where M_i denotes the i-th available type mode or prediction mode, C_i is the corresponding RDO cost, and SFD_i is the SFD value of mode i. The problem seeks to minimize the RDO cost C from a set of available modes that satisfy the SFD constraint $SFD_i \leq SFD_{th}$. SFD_{th} is the N_{top}-th SFD in the ascending-order sorted array of SFD_i, and N_{top} is given by:

$$N_{top} = \lceil N \times P_{top} \rceil, \tag{3.3}$$

where N is the total number of available modes, and P_{top} is a custom parameter that stands for how many top percent of total available modes N will be considered in joint T-RD selection.

Algorithm 1 is designed to solve this problem for both type mode and prediction mode selection. For a stable background MB, first, all available type modes are tried, the corresponding RDO costs and SFD values are recorded (lines 2-5 in Algo. 1), then the best type mode is determined based on the SFD threshold (lines 6-9); second, all available prediction modes of the selected type mode are tried, the corresponding RDO costs and SFD values are recorded (lines 10-13), and then the best prediction mode is determined based on the SFD threshold (lines 14-17). We take the x264 encoder [90] as our reference encoder. From above description, the complexity of our proposed Algorithm 1 can stay comparable with the corresponding algorithm in the reference encoder, since the extra computations related with SFD are embedded in the original loops of the reference encoder.

Algorithm 1: Intra frame joint T-RD selection

```
if current MB belongs to stable background then
    for available type mode $M_{t_i}$ do
        encode current MB and store $C_{t_i}$;
        calculate and store $SFD_{t_i}$;
    end
    sort records in ascending order based on SFD value, and obtain valid number of records
     ($N_t$);
    obtain $SFD_{t_{th}}$ based on $N_{t_{top}} = \lceil N_t \times P_{top} \rceil$;
    find the minimum $C_t$, subject to $SFD_{t_i} \leq SFD_{t_{th}}$;
    output the corresponding $M_t^*$ as the selected type mode;
    for available prediction mode $M_{p_i}$ of the selected type $M_t^*$ do
        encode current MB and store $C_{p_i}$;
        calculate and store $SFD_{p_i}$;
    end
    sort records in ascending order based on SFD value, and obtain valid number of records
     ($N_p$);
    obtain $SFD_{p_{th}}$ based on $N_{p_{top}} = \lceil N_p \times P_{top} \rceil$;
    find the minimum $C_p$, subject to $SFD_{p_i} \leq SFD_{p_{th}}$;
    output the corresponding $M_p^*$ as the selected prediction mode;
end
```

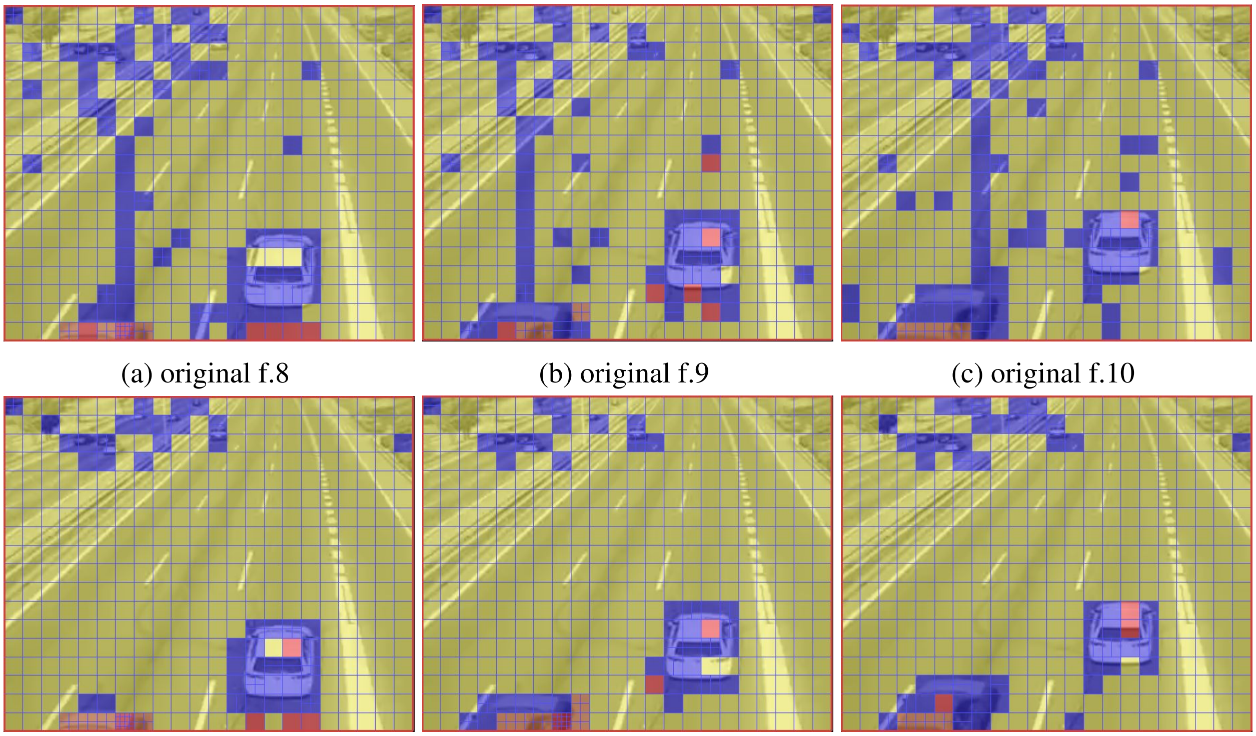
Inter frame coding/mode selection

A typical inter frame analysis process includes three steps, shown as below:

1. Probe P_SKIP mode–that is, encode the current MB assuming no encoding residuals and no Motion Vector (MV) difference, and use only the predictive MV. The *decimate score* is computed, which indicate whether we could set the DCT coefficients to 0 given the DCT coefficients after the actual encoding of this inter MB [112]. If the decimate score of the current MB is less than 6, then the current MB can be encoded as P_SKIP and return [90].

2. Otherwise, other inter type modes, including P_16×16, P_8×16, P_16×8, P_8×8, P_4×8, P_8×4, and P_4×4 modes, are all tried and the corresponding MVs are estimated, and also search is performed on those intra modes.

3. Run the RDO process and determine the best mode from all available modes.


(a) original f.8 (b) original f.9 (c) original f.10

(d) proposed f.8 (e) proposed f.9 (f) proposed f.10

Figure 3.3: Fluctuation of P_SKIP distribution

However, the typical inter frame analysis process can result in temporal fluctuation for stable background areas, which will reduce the accuracy of object detection. For example, three consecutive inter frames (frame 8, 9, and 10) of the *GR* video clip is shown in the first row of Figure 3.3. In this figure, each block represents one MB unit, and yellow, blue, and red colors denote P_SKIP mode, other inter mode, and intra mode, respectively. Obviously, there is fluctuation of P_SKIP location distribution in these consecutive inter frames. For an MB in the stable background area, when the inter mode changes between P_SKIP and other inter prediction modes in consecutive frames, there will be temporal fluctuation in the encoded frames, and such fluctuation might result in FP for object detection due to mistaking for new object appearing.

We propose to reduce temporal fluctuation in inter frames by designing new criteria in the analysis of inter type modes. Specifically, we expect to classify more MBs in stable background areas as P_SKIP or set the MVs of these MBs to zeros, meanwhile we expect to maintain acceptable traditional distortion *SSD*, the *Sum of Squared Differences* between the intensities of an original

Algorithm 2: Inter frame Probe P_SKIP

```
Input: decimate score of current MB.
if decimate score of current MB < 6 then
    current MB is set as P_SKIP;
    return
else if current MB belongs to stable background then
    encode current MB based on predictive MV;
    calculate $SSD_r$ and $SFD_r$ based on the reconstructed MB;
    calculate $SSD_s$ and $SFD_s$ assume current MB as P_SKIP;
    if $SSD_s \leq d\_w \times SSD_r$ and $SFD_s \leq s\_w \times SFD_r$ then
        current MB is set as P_SKIP;
        return
    end
end
```

MB and the intensities of an encoded MB. Based on the typical inter MB analysis process, we design new schemes in the probe P_SKIP process and the analysis of P_16×16 mode.

In probe P_SKIP process, for MBs dissatisfied with the original criterion in [112], we compare the encoding option of P_SKIP with the encoding option of using predictive MV, and if the P_SKIP option brings less SFD while maintaining acceptable SSD, the current MB will be set as P_SKIP. The detailed steps are described in Algorithm 2, where SSD_r and SFD_r are SSD and SFD of the reconstructed MB based on predictive MV, SSD_s and SFD_s are SSD and SFD of the current MB assuming P_SKIP encoding, and d_w and s_w are weight variables that can be customized by encoders. Compared with the x264 reference encoder, Algorithm 2 is additional; however, the overall computational complexity of inter frame coding/mode selection can be reduced because more MBs can be set as P_SKIP that do not need any other inter type modes analysis and RDO.

Furthermore, for the analysis of P_16×16 mode, we design an inter frame *P_16×16 Direct Copy* mode: *direct copy from the corresponding MB in the previous frame due to negligible motion in the stable background area.* If the distortion brought by assuming no motion is comparable with the distortion of reconstructed MB after motion estimation, the process will skip other inter modes analysis and jump to Encode current MB process without RDO, as shown in the flow chart in Figure 3.2. The detailed steps of inter frame P_16×16 Direct Copy mode are described in Algorithm 3, where SSD_{me} is the distortion of the MB based on MV_{me} after motion estimation,

Algorithm 3: Inter frame P_16×16 Direct Copy mode

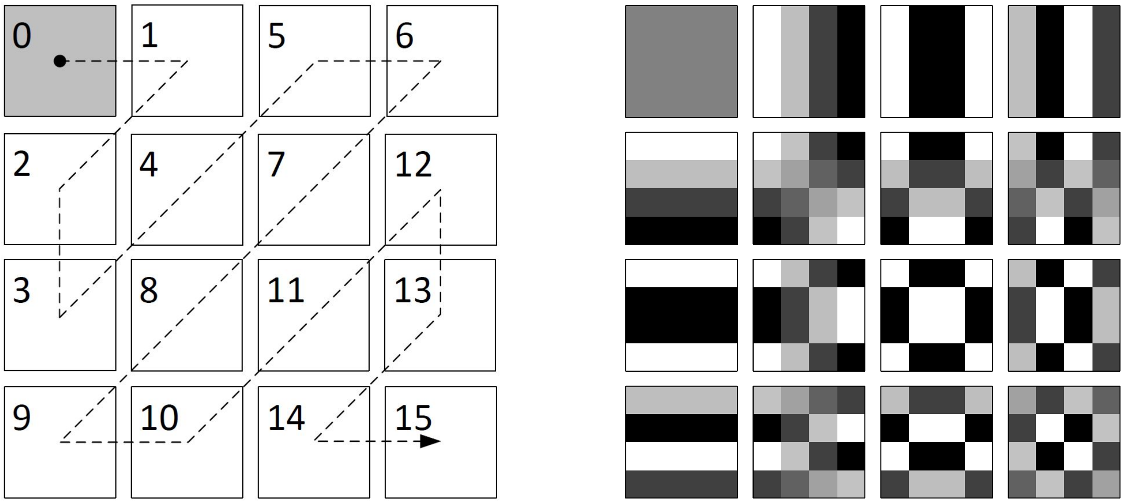
```
Input: $MV_{me}$ after motion estimation in P_16×16 inter analysis.
if current MB belongs to stable background then
    encode current MB based on $MV_{me}$;
    calculate $SSD_{me}$ based on the reconstructed MB;
    calculate $SSD_{dc}$ assume current MB as Direct Copy mode;
    if $SSD_{dc} \leq d\_w \times SSD_{me}$ then
        current MB is set as P_16×16 Direct Copy mode;
        return
    end
end
```

SSD_{dc} is the distortion of the MB based on the assumption that there is no motion and that a direct copy from the corresponding MB in the previous frame is applied, and d_w is a custom weight parameter that restricts SSD_{dc} inside a threshold of $d\_w \times SSD_{me}$. Encoding an MB in Inter P_16×16 Direct Copy mode could skip other Inter modes analysis and RDO, which reduces the overall computational complexity of inter frame coding/mode selection.

An example of the proposed inter coding scheme (combining Algo. 2 and Algo. 3) is shown in the second row of Figure 3.3. Compared with the first row which shows results from the standard inter analysis process, after applying the proposed scheme, more background MBs are encoded as P_SKIP modes, and the distribution of P_SKIP stays stable for consecutive frames. The video snapshots from the two rows look similar, both with acceptable video quality.

3.2.2 Spatial-texture-preserved video encoding scheme

Spatial texture also plays a critical role in automatic object detection. Since 2-D transform encoding, such as the Discrete Cosine Transform (DCT), is indispensable in the modern block-based hybrid video encoders, it is natural to explore the properties of spatial texture in the 2-D transform domain. The texture features of an image are extracted from DCT coefficients for saliency detection in the JPEG bit-stream adaptive image retargeting applications [113]. In [114], the texture features in video saliency are detected through DCT coefficients in the MPEG4 compressed domain. Recent progresses of perceptual image coding with DCT are summarized in [115], in which each



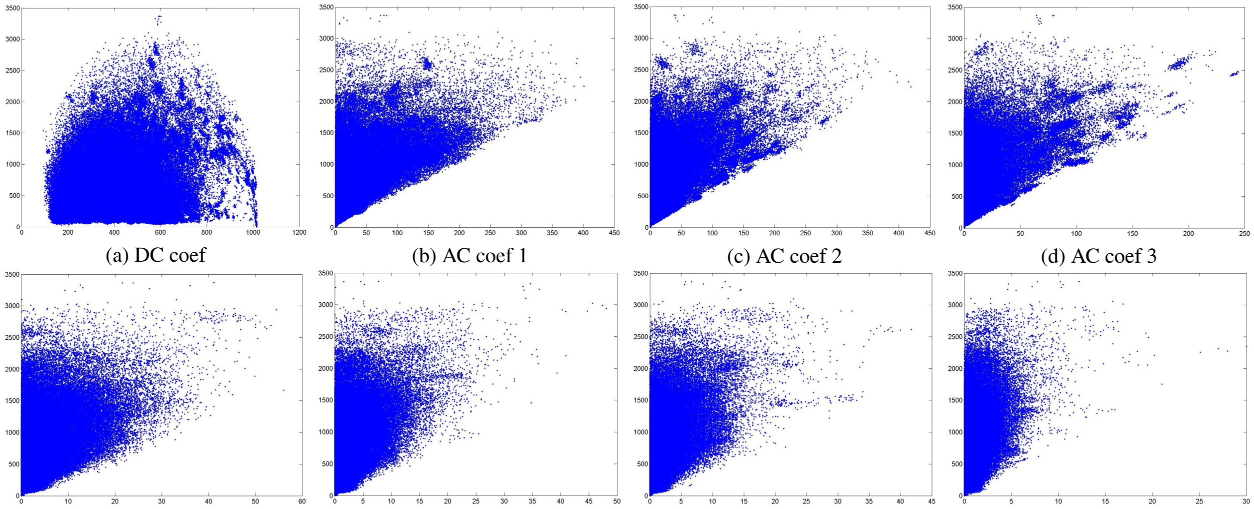
(a) 4×4 SIT coefficients

(b) 4×4 SIT basis patterns

Figure 3.4: 2-D transform in H.264/AVC (4×4 SIT)

image block is classified into plain, edge, or texture class based on the sum of DCT absolute coefficients. The above works are all based on 8×8 DCT in JPEG or MPEG4 but not 4×4 transform in H.264 or H.265. The features of 4×4 transform are studied in [116] with the purpose of designing a tracking-aware H.264 video compression algorithm for transportation surveillance: It has been observed that each coefficient's corresponding basis in the 4×4 transform of the H.264/AVC sharpens vertical and/or horizontal edges to varying degrees, and a new quantization table is designed that can help to identify and concentrate compression bit rate on frequencies useful to tracking, at the cost of bit rate allocated to frequencies confusing or useless to tracking.

From the above works, we learn that the coefficients of 2-D transform are highly related with spatial texture. In JPEG and MPEG4 standards, the DCT coefficients in a 8×8 block includes one DC coefficient and 63 AC coefficients. Among them, the DC coefficient is the average energy over all the 64 pixels in this block, and the left AC coefficients characterize the properties of the block in the frequency domain. Previous studies [113, 114, 115] show that the DCT AC coefficients can be used to represent the texture information for a block. For example, in [113] the DCT AC coefficients are classified into three parts: low-frequency (LF), medium-frequency (MF), and high-frequency (HF) parts. However, H.264/AVC uses a simplified Separable Integer 4×4 Transform (SIT) instead of 8×8 DCT [101]. The coefficients of 4×4 SIT in H.264/AVC standard is shown



(a) DC coef (b) AC coef 1 (c) AC coef 2 (d) AC coef 3

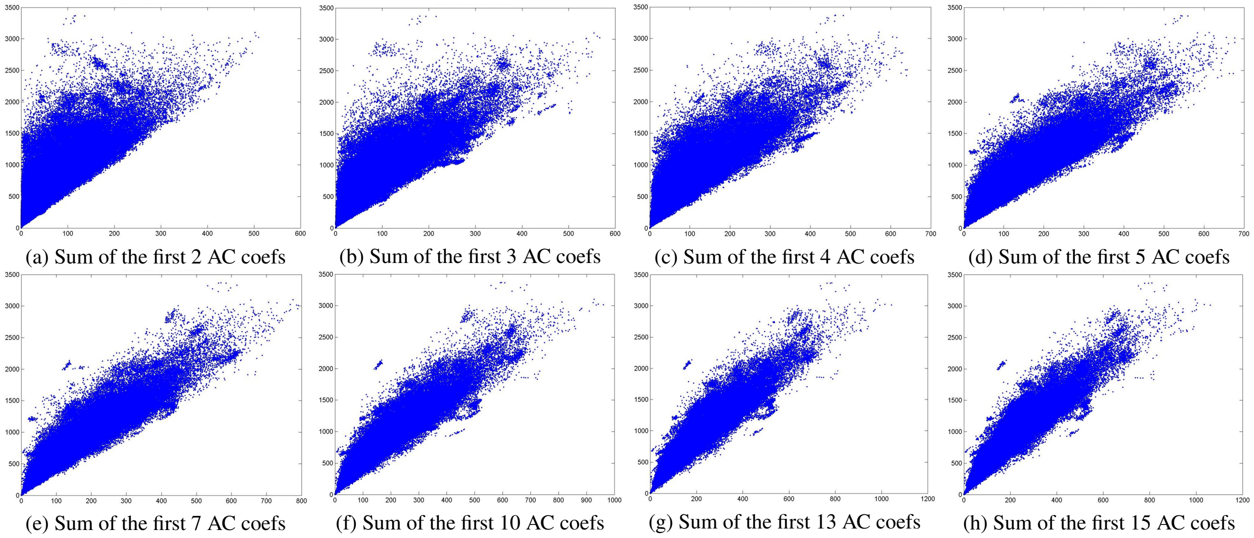
(e) AC coef 12 (f) AC coef 13 (g) AC coef 14 (h) AC coef 15

Figure 3.5: Scatter figure of SIT coefficients with spatial texture information

in Figure 3.4 (a), in which the first block (No. 0) with gray color denotes DC coefficient, and the rest fifteen blocks (No. 1-15) denote 15 AC coefficients. Figure 3.4 (b) shows the standard basis patterns for 4×4 SIT, and the coefficients of SIT can be considered as weighting factors of a set of these basis patterns. Any image block can be reconstructed by combining the sixteen basis patterns with the appropriate weight.

Firstly, we inspect the correlation between each coeff of SIT with spatial texture information. The original Y channel images of *hall* are used as examples to analyze the dynamic foreground regions. All foreground MBs are applied in 4×4 SIT and the texture analysis method mentioned in Chapter 2.2.2. Scatter figures of the absolute value of single SIT coeff with texture are inspected, DC coeff and AC coeffs 1, 2, 3, 12, 13, 14, and 15 with spatial texture information are shown in Figure 3.5. We can find that DC coeff could not reflect the spatial texture level and AC coeffs are related with spatial texture in a certain degree. With the number of AC coeff increasing, the absolute value of AC coeff decreases generally and even close to zero (e.g., AC coeffs 13, 14 and 15), which indicates that texture details in too high frequency are in the minority. However, any single AC coeff is not significant correlated with spatial texture status.

We further investigate the relationship between AC coefficients of SIT with spatial texture information. We use the sum of the first x AC coefficients to represent spatial texture information,



(a) Sum of the first 2 AC coefs (b) Sum of the first 3 AC coefs (c) Sum of the first 4 AC coefs (d) Sum of the first 5 AC coefs

(e) Sum of the first 7 AC coefs (f) Sum of the first 10 AC coefs (g) Sum of the first 13 AC coefs (h) Sum of the first 15 AC coefs

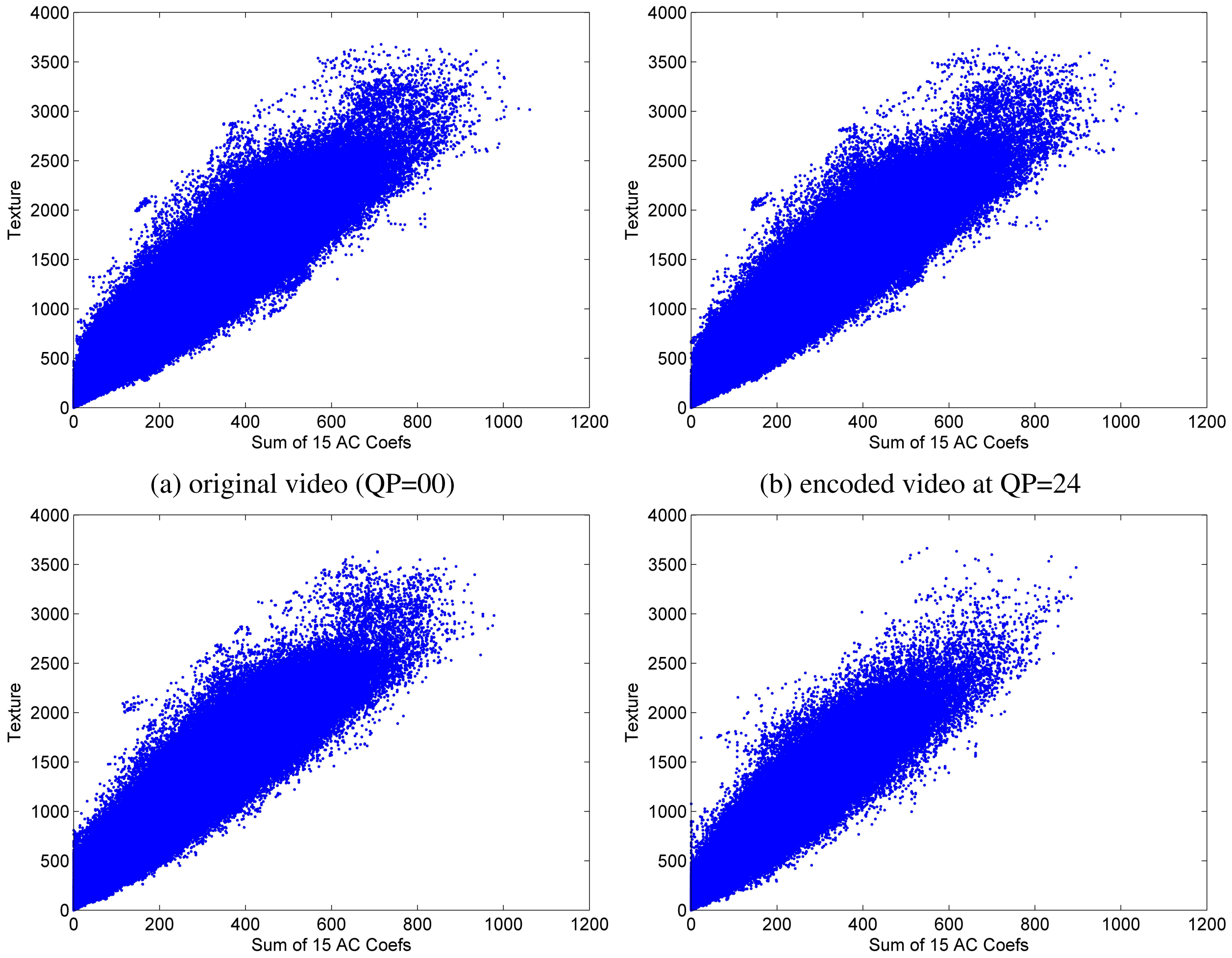
Figure 3.6: Scatter figure of sum of the first x AC coefficients with spatial texture information

where x is an integer more than 0 and less than 16. In Figure 3.6, scatter figures are shown for the sum of the first 2, 3, 4, 5, 7, 10, 13 and 15 AC coefficients with spatial texture information, respectively. And the scatter figure of the first 1 AC coefficient has been already shown in Figure 3.5 (b). We can find that the correlation become more clear when more AC coefficients are considered with x increasing from 1 to 5, and as x continues to increase, there is no obvious improvement of the correlation. It is well known that different computer vision algorithms work based on different levels of features, and therefore, we prefer not to select to preserve specific frequencies, in other words, we try to protect all the frequency details as the original ecology.

Consequently, we introduce the measure *SSAC*, *Sum-of-absolute 15 SIT AC Coefficients*, to depict the spatial texture information of a 4×4 image block as:

$$SSAC = \sum_{i=1}^{i=15} |AC_i|, \tag{3.4}$$

where AC_i is the i-th SIT AC coefficient of one 4×4 image block in an MB. We investigate the entire video dataset, which includes different compression level videos (QP from 24 to 48, step size is 2) and the original raw videos. The scatter figures of the original raw videos (where QP is marked as 00) and the encoded video using QP 24, 36 and 48 are shown in Figure 3.7. Based on



(a) original video (QP=00)

(b) encoded video at QP=24

(c) encoded video at QP=36

(d) encoded video at QP=48

Figure 3.7: Scatter figure of SSAC with spatial texture information

the scatter figures of the entire video dataset, the distribution become more and more concentrated and regular as the compression ratio (QP) increases. We use SSAC value 10 as intervals to average data points in the scatter figure and then obtain curves for all QP setting, which are shown Figure 3.8. We can find that there is a positive linear correlation between SSAC and spatial texture, no matter how much compression is introduced.

The correlation between spatial texture and SSAC is inspected using the Linear Correlation Coefficient (LCC), the Spearman Rank Order Correlation Coefficient (SROCC), and the Kendall Rank Correlation Coefficient (KRCC), respectively. The correlation coefficients are summarized in Table 3.1, in which QP 00 denotes the original raw video. The results of LCC are all above 0.96

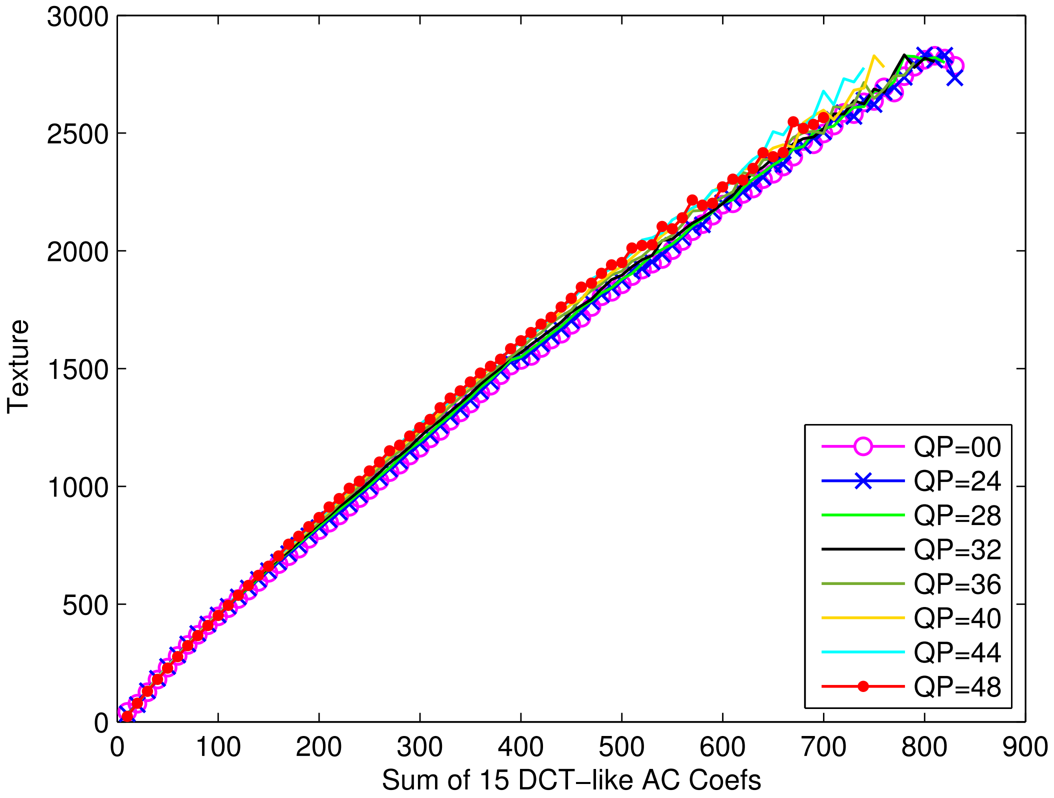


Figure 3.8: The relationship between SSAC and spatial texture information

(average value 0.965), the ones of SROCC are all higher than 0.97 (average value 0.978), and the ones of KRCC are all higher than 0.87 (average value 0.879). These results indicate that there is a significant positive linear correlation between SSAC and spatial texture.

Table 3.1: Correlation coefficients between spatial texture information and SSAC

QP	00	24	26	28	30	32	34	36	38	40	42	44	46	48
LCC	0.962	0.964	0.964	0.964	0.965	0.965	0.965	0.965	0.966	0.966	0.966	0.966	0.966	0.965
SROCC	0.975	0.977	0.977	0.977	0.978	0.978	0.978	0.978	0.978	0.979	0.979	0.979	0.979	0.980
KRCC	0.871	0.874	0.875	0.875	0.876	0.877	0.878	0.879	0.881	0.882	0.883	0.884	0.886	0.889

By far, we have proposed the ideal descriptor in 2-D transform domain, SSAC, to depict the spatial texture information in video encoding scenario. Inheriting the concept of texture degradation from TXD, which is defined in (2.5), we introduce a new TXD^{SIT} as a basic unit to represent texture degradation in a 4×4 block, which is given by:

$$TXD^{SIT} = \left| \sum_{i=1}^{i=15} |AC_i| - \sum_{i=1}^{i=15} |ac_i| \right|, \tag{3.5}$$

where $\sum_{i=1}^{i=15} |AC_i|$ is texture information of a 4×4 block in an MB of the original frame and $\sum_{i=1}^{i=15} |ac_i|$ is texture information of the corresponding 4×4 block in the same MB of the encoded frame. The descriptor of texture degradation TXD^{SIT} brings unique benefits in video encoding context, including: 1) convenient calculation; 2) low computational complexity; 3) finer-grained tuning than original TXD in MB unit.

The block-based video coding can be summarized as Intra/Inter type mode (macroblock partitions) decision and Intra/Inter prediction (nine prediction options or motion compensated prediction) mode decision. To protect spatial texture during the video encoding process, we formulate a joint Spatial-texture and RD (joint S-RD) mode selection problem for both Intra and Inter frames as follows:

$$\begin{aligned} \text{Given :} \quad & \{M_i, C_i, TXD_i^{SIT}\}, \\ \text{Find :} \quad & M^*, \\ \text{Minimize :} \quad & C, \\ \text{Subject to :} \quad & TXD_i^{SIT} \leq TXD_{th}^{SIT}, \end{aligned} \tag{3.6}$$

where M_i denotes the i-th available type mode or prediction mode in Intra/Inter frame, C_i is the corresponding RDO cost, and TXD_i^{SIT} is the TXD^{SIT} value of mode i. The problem seeks to minimize the RDO cost C from a set of available modes that satisfy the TXD^{SIT} constraint $TXD_i^{SIT} \leq TXD_{th}^{SIT}$. TXD_{th}^{SIT} is the N_{top}-th TXD^{SIT} in the ascending-order sorted array of TXD_i^{SIT}, and N_{top} is given by:

$$N_{top} = \lceil N \times P_{top} \rceil, \tag{3.7}$$

where N is the total number of available modes, and P_{top} is a custom parameter that stands for how many top percent of total available modes N will be considered in joint S-RD selection.

Algorithm 4 is designed to solve this problem for both Intra/Inter type mode and Intra/Inter

Algorithm 4: Intra/Inter MB joint S-RD selection

```
Input: Intra prediction options or Inter motion vectors based on SATD scores
if current MB belongs to dynamic foreground then
    4×4 SIT on the original video, and store the original SSAC;
    for available type mode $M_{t_i}$ of Intra/Inter MB do
        encode current MB based on Intra/Inter prediction, store $C_{t_i}$ and the corresponding
          SSAC;
        calculate and store $TXD_{t_i}^{SIT}$;
    end
    sort records in ascending order based on $TXD^{SIT}$ value, and obtain valid number of
      records ($N_t$);
    obtain $TXD_{t_{th}}^{SIT}$ based on $N_{t_{top}} = \lceil N_t \times P_{top} \rceil$;
    find the minimum $C_t$, subject to $TXD_{t_i}^{SIT} \leq TXD_{t_{th}}^{SIT}$;
    select the corresponding $M_t^*$ as the optimal type mode;
    for available prediction mode $M_{p_i}$ based on the selected type $M_t^*$ do
        encode current MB or , store $C_{p_i}$ and the corresponding SSAC;
        calculate and store $TXD_{p_i}^{SIT}$;
    end
    sort records in ascending order based on $TXD^{SIT}$ value, and obtain valid number of
      records ($N_p$);
    obtain $TXD_{p_{th}}^{SIT}$ based on $N_{p_{top}} = \lceil N_p \times P_{top} \rceil$;
    find the minimum $C_p$, subject to $TXD_{p_i}^{SIT} \leq TXD_{p_{th}}^{SIT}$;
    output the corresponding $M_p^*$ of the selected type $M_t^*$ as the optimal mode;
end
```

prediction mode selection. For a dynamic foreground MB, first, the SSAC of the original video is calculated and stored as a benchmark, then all the available Intra/Inter type modes are tried, the corresponding RDO costs and TXD^{SIT} values are recorded (lines 4-7 in Algo. 4), and the best type mode is determined based on the TXD^{SIT} threshold (lines 8-11); second, all available Intra/Inter prediction modes of the selected type mode are tried, the corresponding RDO costs and TXD^{SIT} values are recorded (lines 12-15), and then the best prediction mode is determined based on the TXD^{SIT} threshold (lines 16-19). Based on above procedure, our proposed Algorithm 4 can maintain the same level as the x264 reference encoder in the computational complexity, since the computing related with TXD does not bring extra loops to the reference encoder.

3.3 Performance evaluation

Table 3.2: Video information for data set 2

Video name	AD right	AD left	AD ahead	AD behind	ID stern	ID sv 1	ID sv 2	ID sv 3
SI index	83.71	91.78	95.62	123.63	75.69	53.37	47.08	59.64
TI index	21.14	61.45	42.73	47.37	10.47	12.86	16.99	8.86
Length (sec)	20	60	40	40	60	100	70	90

Table 3.3: Video compression parameters

GOP structure	IPPP	GOP size	20
Rate control	constant QP	QP range	28-46
Intra/Inter custom parameters	P_{top}	d_w	s_w
	0.1	6	0.1

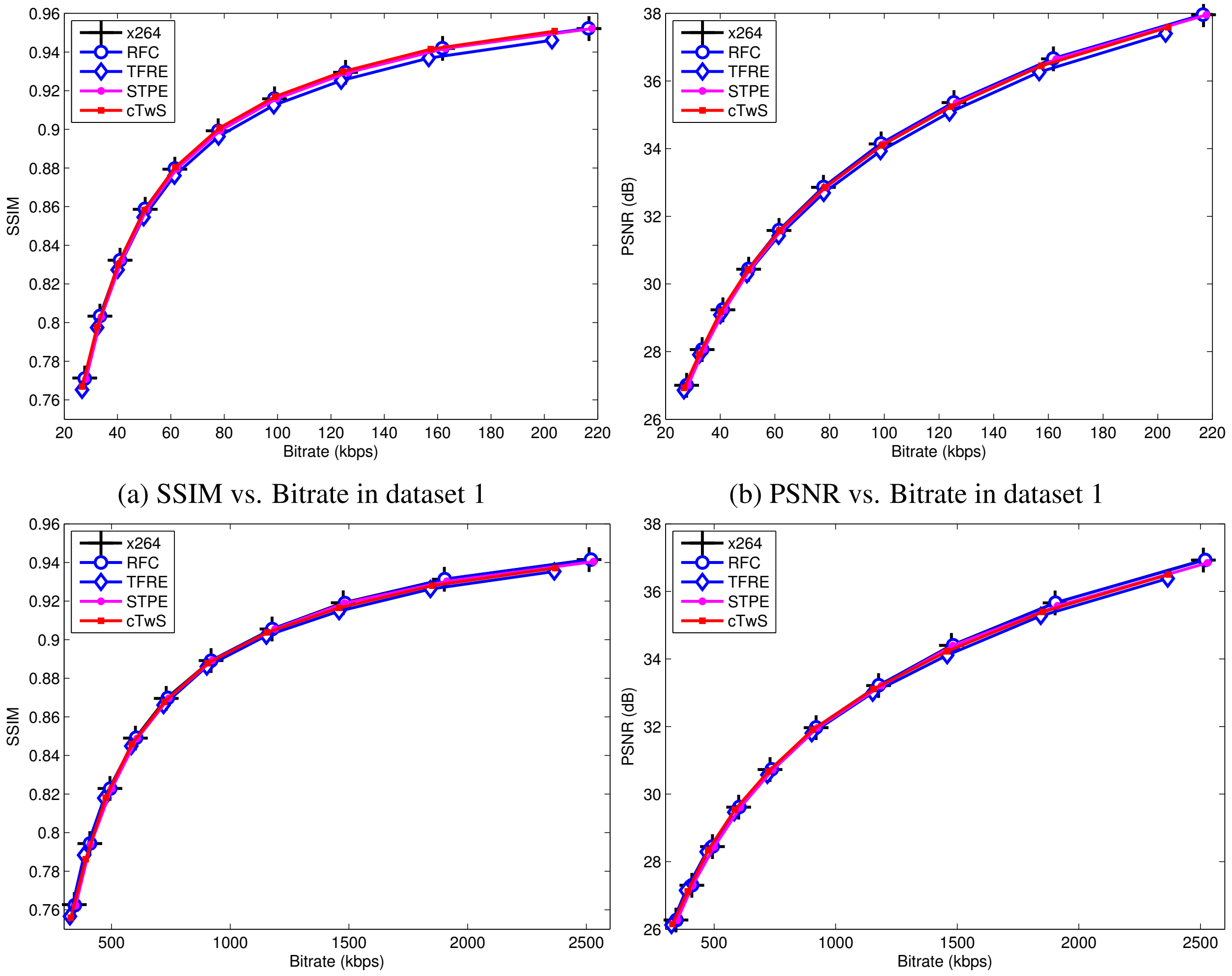
We evaluate the proposed video encoding framework by applying object detection algorithms on a variety of compressed videos. The eight raw videos shown in Figure 2.2 and a new video dataset from PETS 2017 datasets [117] are used for this test. There are uniform resolution (352×288), frame rate (25 fps), and duration (12 sec) in the dataset 1. Dataset 2 contains the ARENA dataset (AD), which includes 4 non-overlapping field of views at each corner of a truck outdoor, and the maritime IPATCH dataset (ID), which includes 1 view at stern and 3 starboard views (sv) of one ship. Different from dataset 1, eight videos in dataset 2 covers higher resolution (1280×960), higher frame rate (30 fps), and longer durations (20∼100 sec), and more details can be found in Table 3.2. Spatial Information (SI) index and Temporal Information (TI) index of a sequence, which are defined by ITU-T P.910 [118] and are directly related to video compression complexity, are also includes in Table 3.2. The x264 encoder (version 0.142.x) is configured to encode videos using one-pass mode with medium speed, and the compression settings are summarized in Table 4.3. The aforementioned three object detection algorithms (GMM, GMG, and ABL) are applied on these compressed videos. One motivation to include relatively higher QP values in our tests is that medium and high compression ratios are used in many wide-area, large-scale, or sparse wireless camera networks with limited bandwidth and energy constraints. For example, a wide-area and large-scale camera network is implemented in [119], a long-duration and large-scale environmental monitoring application is introduced in [120], the deployment of sparse sensor networks in large areas is studied in [121], and the deployment of airborne camera networks is introduced in [122]. These practical systems operate in a bandwidth range of 40 kbps - 300 kbps, providing video observations with around 0.01-0.1 bits per pixel (BPP). The QP values in our experiments could produce videos with bandwidth and BPP ranges consistent with these practical wireless camera systems. Moreover, a similar QP range (28-44) was adopted by other researchers for studies on subjective video quality in [123] and [124], which demonstrated that the perceptual quality of videos encoded with medium and high QP is acceptable.

We evaluate the performance of the proposed algorithms in terms of both pixel level detection accuracy and object level detection accuracy on the two datasets. Evaluation of detection at object level is straightforward, while more precise detection at pixel level provides more insight into strengths and weaknesses of detection performance [125, 126], based on which solutions could be designed to improve object detection performance.

3.3.1 Evaluation of the proposed algorithms in pixel level

The performances of the proposed TRFE scheme, STPE scheme and the combined TFRE with STPE scheme (short for cTwS) are compared to the H.264/AVC-based open source encoder x264 and the Reducing Flicker video Coding approach (RFC) [106]. The objective of RFC is to improve perceptual video quality by reducing flicker effects, and it considers the distortions not only between the encoded video and the original video but also in the temporal domain in the encoded video during intra rate distortion optimization process.

First, we compare the objective video quality and the corresponding bit rate of the five schemes in dataset 1 and dataset 2. The industrial standard PSNR and Structural Similarity (SSIM) are applied to the compressed videos. We evaluate the average performance of the eight different video sequences in dataset 1 and dataset 2 separately at the same QP. The resulting PSNR and SSIM with the corresponding bit rates are shown in Figure 3.9. The R-D performances of RFC



(a) SSIM vs. Bitrate in dataset 1

(b) PSNR vs. Bitrate in dataset 1

(c) SSIM vs. Bitrate in dataset 2

(d) PSNR vs. Bitrate in dataset 2

Figure 3.9: Rate-Distortion curves of proposed schemes

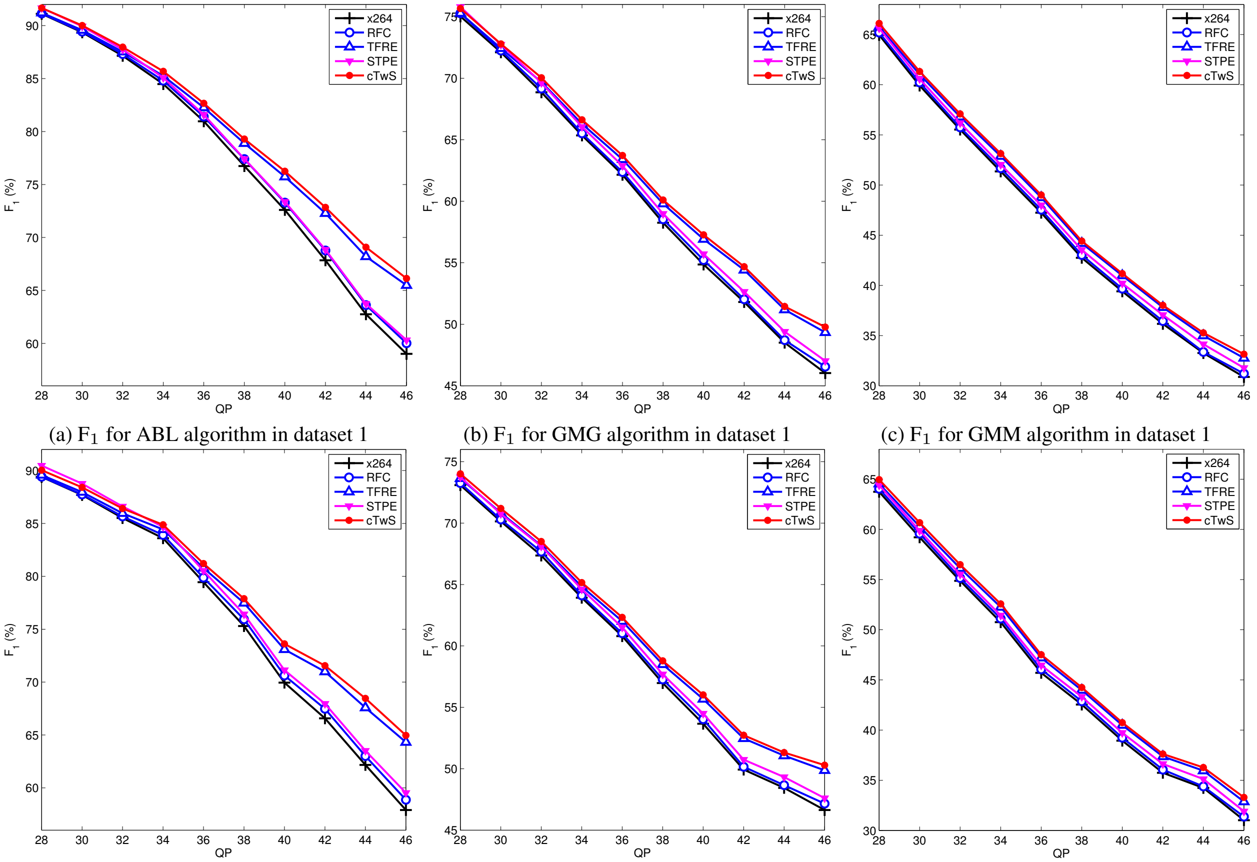
are nearly identical with those of x264 in every QP for both datasets. The curves of STPE almost overlap completely with the ones of x264 for both SSIM and PSNR, which indicates that the proposed STPE scheme has little impact on the Rate-Distortion performance. The PSNR and SSIM values of TFRE scheme decrease slightly, whereas the bit rate is saved compared with ones of x264 encoding. The slight decrease in bit rate is due to the fact that TFRE encodes more inter MBs in P_SKIP modes. For the combined cTwS scheme, comparing with x264 encoding: in dataset 1, the PSNR and SSIM values of cTwS decrease slightly by 0.102 dB and 0.001 on average, respectively, whereas cTwS brings down the bit rate by 2.04 kbps on average; in high resolution videos of dataset 2, its PSNR and SSIM decrease by 0.157 dB and 0.004 on average, respectively, whereas it saves the bit rate by 31.67 kbps on average. Overall speaking, the Rate-Distortion performances of the proposed algorithms are comparable to those of the x264 encoder.

Next, we evaluate the overall performance of object detection in pixel level through F_1 scores. The average F_1 scores of the eight videos in each dataset for the three object detection algorithms are shown in Figure 3.10. Though the three object detection algorithms have different ranges of F_1 in two datasets, the detection performance degrades when QP increases. The curves of RFC scheme always nearly overlap with those of x264 except negligible improvements of ABL algorithm. The performance gains of STPE scheme are larger than ones of RFC and distributed evenly over different QPs. The benefits of TFRE scheme upon x264 for three algorithms on both datasets are noticeable, and the gain of TFRE is higher with larger QP values. The cTwS scheme results in the largest F_1 scores for different QP values in every algorithm on both datasets. More specifically, there are noticeable gains of cTwS over x264 for ABL (average 2.96 and 2.99% for two datasets), and modest gains for GMG (1.92 and 1.94%) and GMM (1.72 and 1.76%).

Finally, we summarize the average F_1 scores of the three object detection algorithms on two datasets in Table 3.4. The numbers in the Δ rows denote the gains of cTwS over the x264 encoder. Three points could be reached based on the summary table and above figures: 1) both the R-D and the object detection performances of RFC are nearly identical with that of x264; 2) the R-D performance of cTwS is comparable to that of the x264 encoder and RFC; 3) the improvement on detection performance of cTwS at every QPs is obvious, and the gain of cTwS is higher with larger QP values. These results indicate that, by reducing temporal fluctuation in stable background areas and preserving spatial texture in foreground areas, the proposed video encoding framework could effectively improve the accuracy of object detection in pixel level for different types of detection algorithms with no impact on the R-D performance.

3.3.2 Evaluation of the proposed algorithms in object level

To evaluate the performance of the proposed algorithms in object level, a uniform post-processing procedure is adopted to the pixel-level results of three detection algorithms. The post-processing



(a) F_1 for ABL algorithm in dataset 1

(b) F_1 for GMG algorithm in dataset 1

(c) F_1 for GMM algorithm in dataset 1

(d) F_1 for ABL algorithm in dataset 2

(e) F_1 for GMG algorithm in dataset 2

(f) F_1 for GMM algorithm in dataset 2

Figure 3.10: F_1 scores of test videos in proposed schemes

modules includes:

1. Median filtering (5×5 rectangular aperture);

2. Morphological operations (first opening then closing with 3×3 square structure);

3. Connected-component labeling (8-way connectivity); and

4. Region thresholding (240 pixels).

For the object-level detection accuracy, we calculate the Configuration Distance (CD) [96], which measures the difference between the amount of GT objects and AR objects according to their presence. For one given frame, the CD_f can be calculated by:

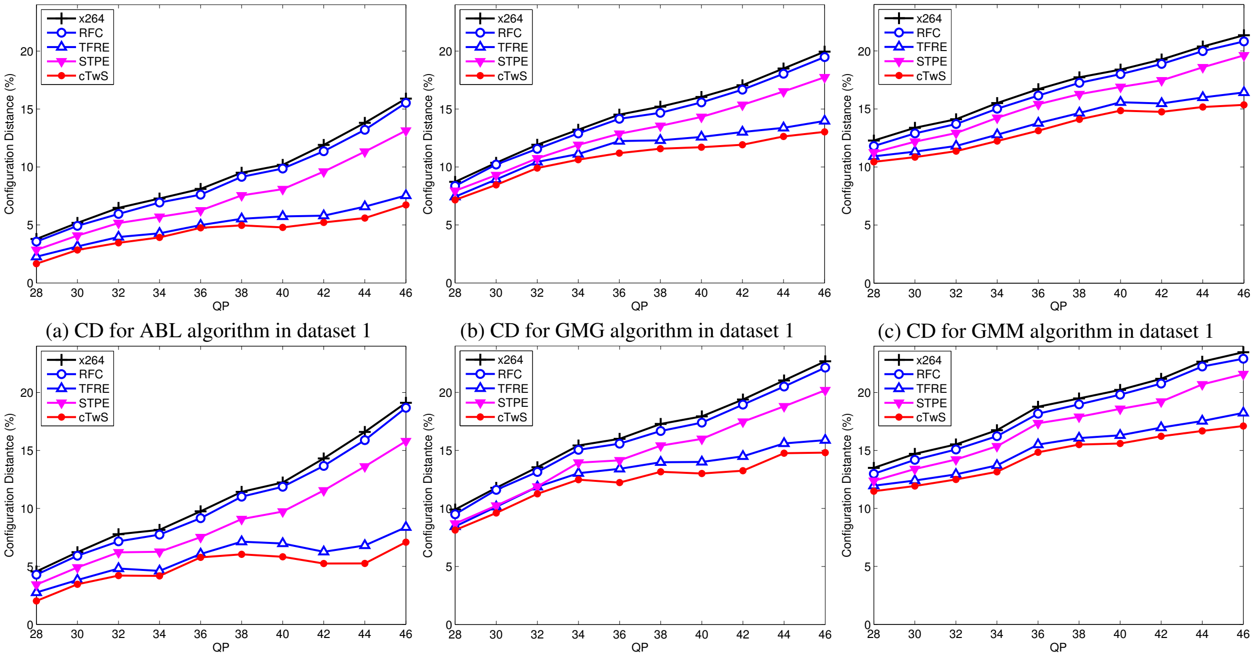
$$CD_f = \left| \frac{AR_o - GT_o}{max(GT_o, 1)} \right|, \tag{3.8}$$

Table 3.4: Average results of proposed algorithms in pixel level

QP		28	30	32	34	36	38	40	42	44	46
Dataset 1 F$_1$ (%)	x264	77.03	73.80	70.51	67.06	63.45	59.25	55.62	51.94	48.17	45.31
	RFC	77.21	73.97	70.74	67.33	63.77	59.65	56.06	52.42	48.56	45.92
	TFRE	77.70	74.42	71.19	67.72	64.16	59.98	56.43	52.85	49.10	46.37
	STPE	77.43	74.34	71.34	68.14	64.81	60.96	57.88	54.83	51.46	49.20
	cTwS	77.81	74.71	71.70	68.48	65.13	61.27	58.23	55.18	51.94	49.68
	Δ	0.78	0.91	1.19	1.42	1.68	2.02	2.61	3.24	3.77	4.37
Dataset 2 F$_1$ (%)	x264	75.41	72.35	69.25	66.09	61.98	58.27	54.19	50.75	48.28	45.18
	RFC	75.57	72.54	69.49	66.38	62.32	58.69	54.63	51.23	48.68	45.80
	TFRE	75.92	73.03	70.08	67.17	63.33	59.98	56.42	53.60	51.53	49.02
	STPE	76.21	73.13	70.08	66.91	62.83	59.14	55.12	51.77	49.32	46.35
	cTwS	76.34	73.43	70.47	67.54	63.68	60.31	56.79	53.97	52.02	49.52
	Δ	0.93	1.08	1.22	1.45	1.70	2.04	2.60	3.22	3.74	4.34

where AR_o and GT_o are the numbers of AR objects and GT objects in the frame. The CD of a video sequence is obtained by the average of CD_f in each frame. In our experiments, object detection results from the raw videos are regarded as GT, and results from the compressed videos are AR. Ideally, if a video is compressed in a lossless way, the corresponding CD is 0. Lossy compression inevitably degrades the performance of object detection, and both false positives and false negatives (i.e., detection mistaking and detection missing) could result in an increase of CD.

The performance of proposed algorithms in object level through CD value in two datasets is shown in Figure 3.11. Despite different ranges of CD value in two datasets, the trends of the three detection algorithms are similar. The curves of RFC are always very close to the ones of x264 for three algorithms. The minimum gains of STPE over x264 for ABL, GMG, and GMM algorithms (-0.94%, -0.77% and -1.03% in dataset 1, -1.13%, -1.21% and -1.12% in dataset 2) are larger than the maximum gains of RFC (-0.60%, -0.53% and -0.56% in dataset 1, -0.71%, -0.61% and -0.60% in dataset 2). The TFRE scheme improves the detection performance significantly. The combined cTwS scheme attains the remarkable improvement (average gains: -4.82%, -3.73% and 3.68% in dataset 1, -6.10%, -4.22% and -4.12% in dataset 2, respectively). The average CD values of the three object detection algorithms on two datasets are also summarized in Table 3.5. The values in the Δ rows denote the gains of cTwS over the x264 encoding. In summary,



(a) CD for ABL algorithm in dataset 1
(b) CD for GMG algorithm in dataset 1
(c) CD for GMM algorithm in dataset 1
(d) CD for ABL algorithm in dataset 2
(e) CD for GMG algorithm in dataset 2
(f) CD for GMM algorithm in dataset 2

Figure 3.11: Configuration Distance (CD) of testing videos in proposed schemes

the average gains of RFC over x264 is quite limited (average -0.42% in dataset 1 and -0.47% in dataset 2, respectively), STPE achieves better improvement than RFC (with average gains -1.60% and -1.86%), the improvement of TFRE is considerable (with average gains -3.42% and -4.04%), and cTwS achieves the maximum average gains (-4.08% and -4.82%).

3.3.3 Evaluation of the computational complexity

The computational complexity of algorithms is a crucial design factor for distributed wireless surveillance systems. All video encoding in this chapter was performed exclusively on a computer based on Intel Xeon E5-2637 v3 (3.50 GHz) processor running on Windows 7 Enterprise operating system. Computational complexity was measured by the encoding time for x264 encoder, RFC approach, TFRE scheme, STPE scheme, and the combined TFRE with STPE scheme (cTwS). Computational complexity is evaluated by the average encoding time of running separately three times for both the 80 test cases in dataset 1 and the 80 test cases dataset 2, which is summarized in Table 3.6. Each dataset consists of 8 different videos in 10 different QP configurations.

Table 3.5: Average results of proposed algorithms in object level

QP		28	30	32	34	36	38	40	42	44	46
Dataset 1 CD (%)	x264	8.26	9.64	10.84	12.00	13.11	14.15	14.87	16.07	17.56	19.06
	RFC	7.91	9.34	10.41	11.62	12.64	13.69	14.48	15.64	17.08	18.61
	TFRE	7.35	8.52	9.61	10.61	11.51	12.46	13.10	14.14	15.48	16.83
	STPE	6.87	7.80	8.74	9.40	10.34	10.83	11.30	11.43	11.98	12.65
	cTwS	6.42	7.38	8.24	8.93	9.69	10.22	10.45	10.63	11.13	11.70
	Δ	-1.84	-2.26	-2.60	-3.07	-3.42	-3.93	-4.42	-5.44	-6.43	-7.36
Dataset 2 CD (%)	x264	9.32	10.91	12.28	13.44	14.84	16.07	16.80	18.29	20.09	21.75
	RFC	8.93	10.57	11.79	13.01	14.30	15.55	16.36	17.79	19.54	21.24
	TFRE	8.17	9.52	10.77	11.85	13.00	14.13	14.77	16.08	17.70	19.19
	STPE	7.72	8.79	9.87	10.45	11.66	12.39	12.43	12.57	13.32	14.17
	cTwS	7.22	8.33	9.32	9.94	10.95	11.57	11.48	11.57	12.23	13.00
	Δ	-2.10	-2.58	-2.96	-3.50	-3.89	-4.50	-5.32	-6.72	-7.86	-8.75

Table 3.6: The computational complexity of algorithms

Algorithms	x264	RFC	TFRE	STPE	cTwS
Dataset 1 complexity (ms)	1934.985	2131.977	1278.990	2021.941	1309.986
Gains (%)	—	+10.18	-33.90	+4.49	-32.30
Dataset 2 Complexity (ms)	18743.861	20622.032	14814.014	19655.053	15391.520
Gains (%)	—	+10.02	-20.97	+4.86	-17.89

The computational complexity of x264 encoder in Table 3.6 is regarded as a benchmark. The RFC approach increases more than 10% in complexity due to extra computing introduced during intra RDO process. The proposed TFRE scheme reduces computational complexity significantly (-33.90% and -20.97%, respectively) thanks to avoiding unnecessary Inter modes analysis and RDO process. The proposed STPE scheme maintains comparable complexity (less than 5%). Finally the combined TFRE with STPE scheme achieves -32.30% and -17.89% reductions in computational complexity for dataset 1 and dataset 2, respectively. The reduction in complexity will provide considerable benefits for distributed wireless surveillance applications.

3.4 Conclusion

In this chapter, we have proposed an efficient video encoding framework that aims at improving the performance of object detection on compressed videos in distributed wireless surveillance systems. This framework includes the *Temporal-Fluctuation-Reduced video Encoding* scheme (TFRE) for the encoding of stable background areas and the *Spatial-Texture-Preserved video Encoding* scheme (STPE) for the encoding of dynamic foreground areas. Besides, this framework is compliant with the block-based hybrid encoding architecture, and its computational complexity of H.264-based implementation is reduced significantly to that of common H.264 encoding schemes. Experimental results on a variety of encoder settings and object detection algorithms indicate that, compared with traditional encoding schemes, the framework improves the accuracy of object detection and results in lower bit rate and significantly reduced complexity with comparable video quality in terms of PSNR and SSIM. This standard-compliant video encoding framework can promote the development and applications of many distributed wireless surveillance systems.

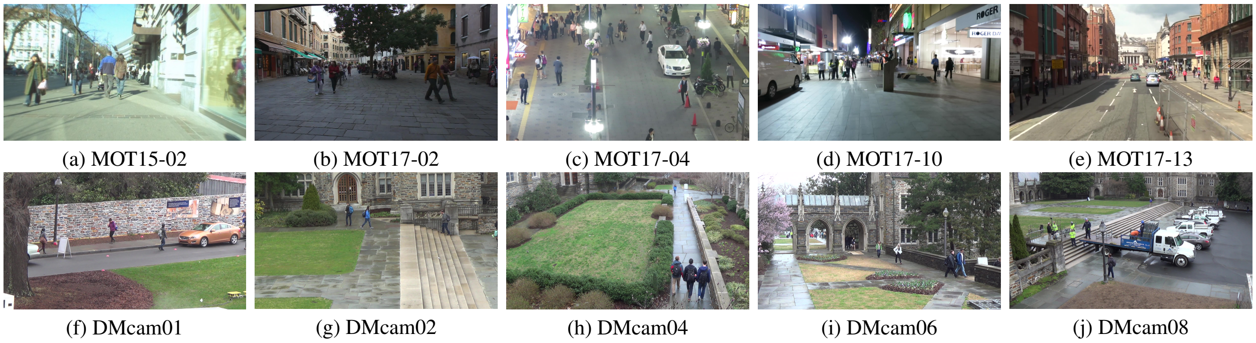
Chapter 4

Modeling of object detection quality for local processing on embedded cameras

Apart from the distortion introduced by compression, the quality of an image or a video could be degraded during the data acquisition or sensing process, e.g., distortion caused by noise or motion blur, or reduced image resolution due to storage or bandwidth constraints on embedded cameras. These factors should also be taken into consideration to evaluate the quality of an image.

Object detection is the first and the most important step in the process of automatic analysis, because the detected objects provide a focus of attention for the following tasks such as tracking and recognition. In this chapter, we propose a blind regression model based on a bagging ensemble of trees to predict the performance of object detection on an image. The model utilizes local features in an image such as edge and oriented gradient and global features including image gradient and estimated object size, which could be easily extracted from an image. The model is trained using a large number of images with different scene characteristics and four types of distortions including noise, Gaussian blur, motion blur, and reduced spatial resolution. The accuracy of the proposed model is evaluated on a separate test data set and compared against commonly used IQA measures.

There are only a few studies on the problem of quality evaluation for automatic analysis algo-


(a) MOT15-02 (b) MOT17-02 (c) MOT17-04 (d) MOT17-10 (e) MOT17-13

(f) DMcam01 (g) DMcam02 (h) DMcam04 (i) DMcam06 (j) DMcam08

Figure 4.1: Snapshots of video data set.

rithms. Image quality assessment for face recognition applications was studied in [70, 71, 72]. Five quality factors were evaluated, including contrast, brightness, focus, sharpness, and illumination, and a face image quality index combining the five factors was proposed in [70]. In [71], natural scene statistics was used to detect degradation of infrared images for face recognition. In [72], the degradation in the performance of face detectors were quantified considering different factors including noise, blur, and compression.

There are also a few studies on the quality for target detection, target tracking, and event detection for airborne reconnaissance applications. In [74], the applicability of the National Imagery Interpretability Ratings Scale (NIIRS) to an automated target detection algorithm was examined, and it was found that NIIRS is not a good predictor of target detection performance. In [75] and [76], the impacts of video frame rate and two spatial factors (noise and spatial resolution) on the tracker performance were investigated.

The aforementioned studies investigated the performance of automatic analysis on specific applications like face recognition and airborne reconnaissance. Our work advances the state of the art by addressing the challenge of building a more general quality prediction model for a wide range of object detection algorithms and diverse scene characteristics. Moreover, our model considers four common types of distortions during the imaging process.

4.1 Data set and object detection measure

In this section, we introduce the data set, the object detection algorithms, and the measure used for evaluating object detection accuracy in our study.

4.1.1 Data set

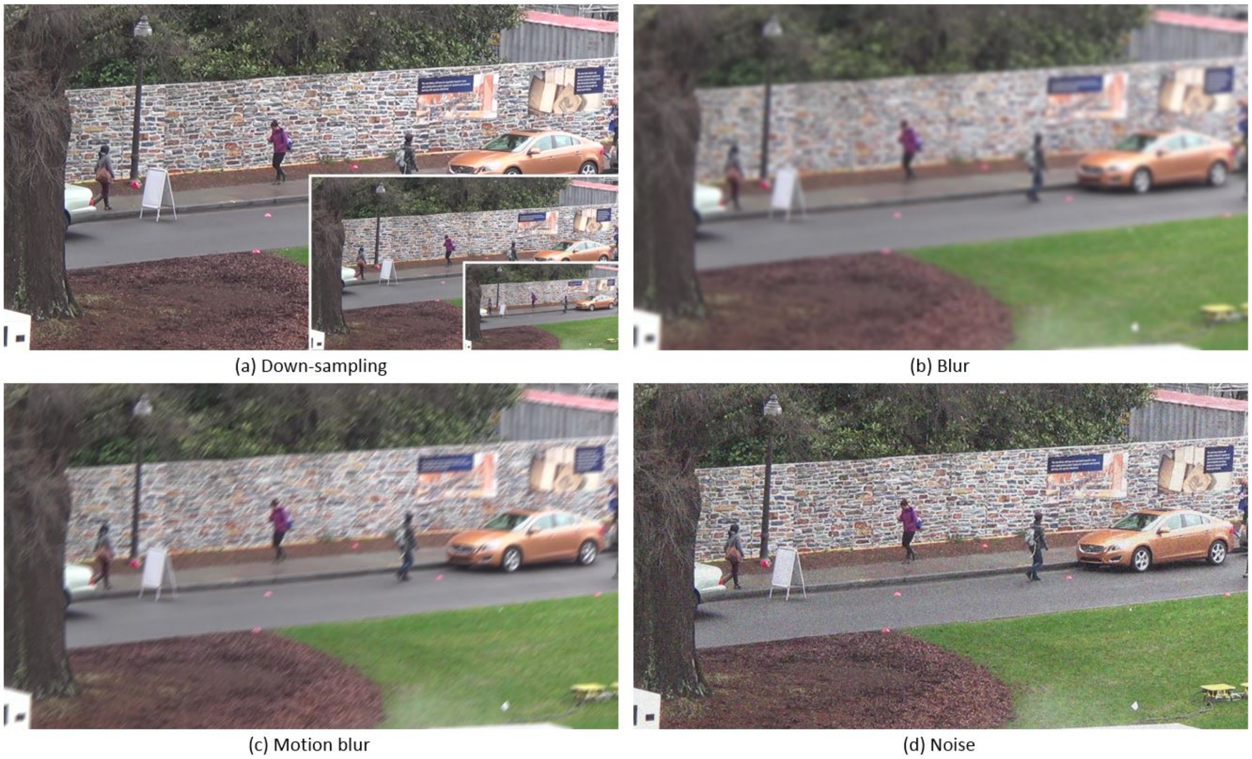
We have selected 10 high resolution original video sequences with different scene characteristics, illumination levels, and object scales. Among them, 5 videos are chosen from the Multiple Object Tracking (MOT) dataset [127], and 5 videos are chosen from the Duke Multi-Target Multi-Camera Tracking (DM) dataset [128]. The resolutions of these videos are mostly 1920×1080 except for one video with 640×480 resolution, and the average number of frames is 741. The snapshot of these videos are shown in Figure 4.1.

Blur and noise are major factors that degrade imaging quality for surveillance or mobile cameras. To understand how the performance of object detection could be affected by blur and noise during the image sensing process, we have generated distorted video sequences based on the original videos, including videos with Gaussian blur, motion blur, and imaging noise. For each type of distortion, distortion levels are set to low, medium, high, higher, extreme level, and the simulation parameters and setting are selected based on the experiments conducted in recent works [129, 130]. We have also included reduced spatial resolution versions of the original videos to study the effect of spatial resolution on object detection.

- The blurring effect of a video is generated by 2D circularly symmetric Gaussian blur kernels with standard deviations of [1.2, 2.5, 6.5, 15.2, 33.2] for five levels, respectively.

- The motion blur is simulated to approximate the linear motion of a camera by [5, 12, 20, 40, 100] pixels with an angle of 45 degrees.

- White Gaussian noise is added to the original images, where variances are set to be [0.001, 0.006, 0.022, 0.088, 1].


(a) Down-sampling

(b) Blur

(c) Motion blur

(d) Noise

Figure 4.2: Samples of different distortions.

- For reduced spatial resolution, 1:2 and 1:4 down-sampling rates are applied in both horizontal and vertical directions on the original images.

Samples for the four types of distortion are shown in Figure 4.2, in which the original image frame is the 581_{th} frame of DMcam01 video. Figure 4.2 (a) shows the image with reduced spatial resolutions, which includes 3 resolutions overlaying in one image, corresponding to original, half, and quarter resolutions in both horizontal and vertical directions. Figure 4.2 (b) is a sample of blur to simulate out-of-focus blur, Figure 4.2 (c) is a sample of motion blur to simulate camera shake during exposure, and Figure 4.2 (d) is a sample of white noise to simulate imaging noise in low-light scenarios. For each original video sequence shown in Figure 4.1, we have generated a total number of 17 distorted videos, including 2 videos from reduced spatial resolution and 5 videos from each type of other three distortions. This results in a total number of 180 video sequences (including the original ones), and it correspond to a total number of 133344 images in our data set.

4.1.2 Object detection algorithms

There are two categories of object detection algorithms in the field of computer vision: one based on building models of backgrounds and the other based on building models for objects. Algorithms based on background modeling require multiple frames to build a stable background, while methods based on object modeling could generate detection results on a single image. In this Chapter, we aim at evaluating the quality of single images, such that the wireless embedded camera could adjust its sensing strategy based on the predicted quality and energy supply. Therefore, we focus on object modeling methods. Furthermore, we consider the scenario that the embedded camera perform local object detection in a fast manner, so low-complexity object detection algorithms are preferred. We use the following three representative lightweight algorithms based on object modeling:

1. Histograms of Oriented Gradients (HOG) [32];
2. Discriminatively Part Models (DPM) [131]; and
3. Locally Decorrelated Channel Features (LDCF) [132].

4.1.3 Object detection measure

The evaluation measures for object detection could be either sequence-based or image-based. Since our goal is to predict and adjust the performance of object detection once an image is taken, we evaluate the object detection accuracy of each frame in a video. The Frame Detection Accuracy (FDA) is a comprehensive metric that accounts for important measures of system performance (such as number of objects detected, missed objects, false positives, and localization error of detected objects) in a single score [133]. For a given frame, the optimal matching pairs is assigned firstly by computing the spatial overlap between ground truth and detected objects. Then, the FDA measure calculates the spatial overlap between the ground truth and system output objects as a ratio of the spatial intersection between the two objects and the spatial union of them. The sum of all of the overlaps is normalized over the average number of ground truth and detected objects.

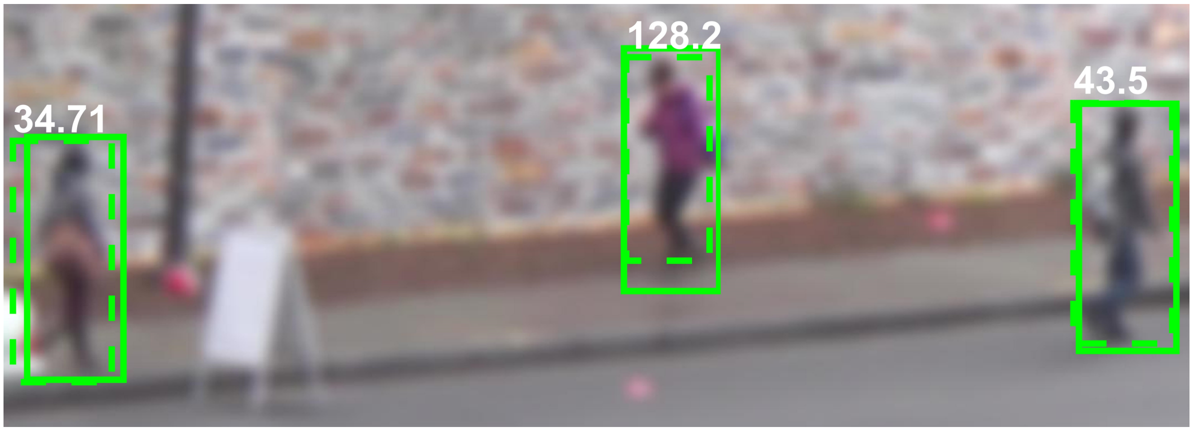


Figure 4.3: Detection sample.

For one image, where there are N_G ground-truth objects G and N_D detected objects D, N_m is the number of mapped object pairs, FDA is defined as:

$$FDA = \frac{\sum_{i=1}^{N_m} \frac{G_i \cap D_i}{G_i \cup D_i}}{(N_G + N_D)/2}. \tag{4.1}$$

A detection system needs to take an image and return a bounding box and a confidence for each detection. The provision of a confidence level allows results to be ranked such that the trade-off between false positives and false negatives can be evaluated, without defining arbitrary costs on each type of classification error [134]. However, the original FDA measure does not reflect the trade-off between false positives and false negatives. Thus, we introduce a revised FDA measure, rFDA for short, which is the average of FDA based on different thresholds (T) of detection confidence levels (C). $rFDA$ is defined as:

$$rFDA = \sum_{j=1}^{N_m} \left(\frac{\sum_{i=1}^{N_{T_j}} \frac{G_i \cap D_i}{G_i \cup D_i}}{\frac{N_G + N_D}{2}} \right) / N_m, \tag{4.2}$$

where N_m is the number of mapped object pairs, N_{T_j} is the number of true positives when the threshold of detection confidence T_j equals to C_j, $j \in \{1, ..., N_m\}$, and C_j denotes the detection confidence level of the j-th mapped detected object.

We use the detection sample shown in Figure 4.3 to explain how to compute rFDA. Figure 4.3 corresponds to a part of the 581^{th} frame of high blur distorted DMcam01 video by the LDCF detector. The ground truth is highlighted in solid line, and three detected objects in dash line with confidence levels 34.71, 128.2, and 43.5, respectively. When the threshold T equals to the minimum confidence $C(min)$, i.e., $T = 34.71$, the three detection results are all true positives, which is the same with the original FDA definition; when $T = 43.5$, only two detection results are regarded as true positives. On the other hand, the SSIM (0.51) and PSNR (21.14 dB) values of this image actually are quite low and poor, however, detection performance is pretty good, which indicates that the popular image quality assessments can not reflect the detection quality. The original FDA measure in (4.1) can be regarded as $FDA_{T(min)}$, which uses the minimum detection confidence level $C(min)$ in mapped pairs as threshold such that all mapped object pairs are true positives.

In order to validate the proposed rFDA measure, we compare the correlation of FDA and rFDA with Average Miss Rate (AMR), which is the most popular metric used in the object detection area. The AMR of an image sequence [135] can be determined as follows: first, a detected object and a ground truth form a match if they overlap sufficiently, which is evaluated by the ratio of the intersection between two objects and the union of them, and a threshold ratio of 0.5 is commonly used; then, the miss rates against false positives per image (FPPI) is plotted (using log-log plots) by varying the threshold on detection confidence; finally, the log-average miss rate is used to summarize the detector performance by averaging miss rate at nine FPPI rates evenly spaced in log-space in the range of 0.01 to 1. Since AMR is calculated based on an entire image sequence, we measure the detection performance for the whole sequence using Sequence Frame Detection Accuracy (SFDA) introduced in [133]. SFDA is an average of the FDA measured over all frames in sequence. The average is normalized to the number of frames in the sequence where at least a ground truth or a detected object exists. SFDA is formulated as:

$$SFDA = \frac{\sum_{t=1}^{N_{frames}} FDA(t)}{\sum_{t=1}^{N_{frames}} \exists(N_G^t \; OR \; N_D^t)}, \tag{4.3}$$

Table 4.1: Comparison of rFDA and FDA in correlation and variation

Detectors	DPM	HOG	LDCF	Average	Δ
Correlation of SFDA vs. AMR	0.9183	0.6807	0.8872	0.8288	–
Correlation of SrFDA vs. AMR	0.9536	0.7734	0.9307	0.8859	0.0571
Variation (σ) of FDA	0.2600	0.1657	0.2347	0.2202	–
Variation (σ) of rFDA	0.1818	0.1187	0.1669	0.1558	-0.0643

where N_G^t and N_D^t denote the number of ground-truth objects and the number of detected objects in frame t, respectively, N_{frames} is the number of frames in the sequence, and $FDA(t)$ is the FDA value for frame t.

Table 4.2: The consistency between different detectors

DPM vs. HOG	DPM vs. LDCF	HOG vs. LDCF	Average
0.8715	0.8266	0.8352	0.8445

Similarly, we can compute the sequence level measure of the proposed rFDA, which could be referred to as SrFDA. We evaluate the correlation of AMR with the sequence level FDA measures on our entire data set. The correlation coefficients of SFDA with AMR and the ones of SrFDA with AMR for the three different detectors are summarized on Table 4.1. We can find that there are obvious improvements comparing SrFDAs correlation with SFDAs for all the three detectors. Specifically, gains are 0.0353, 0.0927, and 0.0435 for DPM, HOG, and LDCF detectors, respectively. The average of correlation coefficients between SrFDA and AMR for the three detectors reaches 0.8859, and the average gain is 0.0571, which indicates that SrFDA is more consistent with AMR. It indicates that rFDA can depict the performance of object detection and it is a better metric for single image detection performance. In addition, the variations of each frames FDA and rFDA in the image sequences are inspected based on standard deviation, and the results are summarized in Table 4.1. We can notice that the variations of rFDA are always smaller than the ones of

FDA for the three different detectors, which indicates that rFDA can reduce arbitrary fluctuations and maintain more stable measurements.

The correlation between the different object detectors are also investigated, as shown in Table 4.2. The correlation coefficients are all above 0.82, and the average of them reaches 0.8445. Although the operating principles of the three detectors are different, the correlation results indicate that their detection performances are consistent. Therefore, we target at predicting the average performance of the three detectors in the proposed image quality adjustment framework.

4.2 Blind model for object detection quality

We build a blind/no-reference regression model to predict the performance of object detection on an image, given by rFDA in the previous section. The regression model utilizes four categories of features: gradient, edge, compact HoG, and estimated object size.

Boundary information in an image plays an important role in object detection and pattern recognition, since boundaries represent the transition regions between objects and background where the image intensities vary abruptly or have discontinuities. Gradient is a good indicator for the variance of image intensities. For an image $f(x, y)$, the gradient of f at location (x, y) is defined as the two dimensional column vector: $[\partial f/\partial x, \partial f/\partial y]^T$, where $\partial f/\partial x = f(x+1, y) - f(x-1, y)$, and $\partial f/\partial y = f(x, y+1) - f(x, y-1)$ using finite difference filters. The magnitude and direction of this gradient at location (x, y) are given by:

$$mag(x) = \sqrt{(\partial f/\partial x)^2 + (\partial f/\partial y)^2}, \tag{4.4}$$

$$dir(x, y) = tan^{-1}\left[\frac{\partial f/\partial y}{\partial f/\partial x}\right]. \tag{4.5}$$

One sample of image gradient is shown in Figure 4.4. The original image in Figure 4.4 (a) is a portion of the 581_{th} frame of DMcam01 video, and Figure 4.4 (b) and (c) show the corresponding image gradient directions and magnitudes, respectively. We can observe that the image gradient

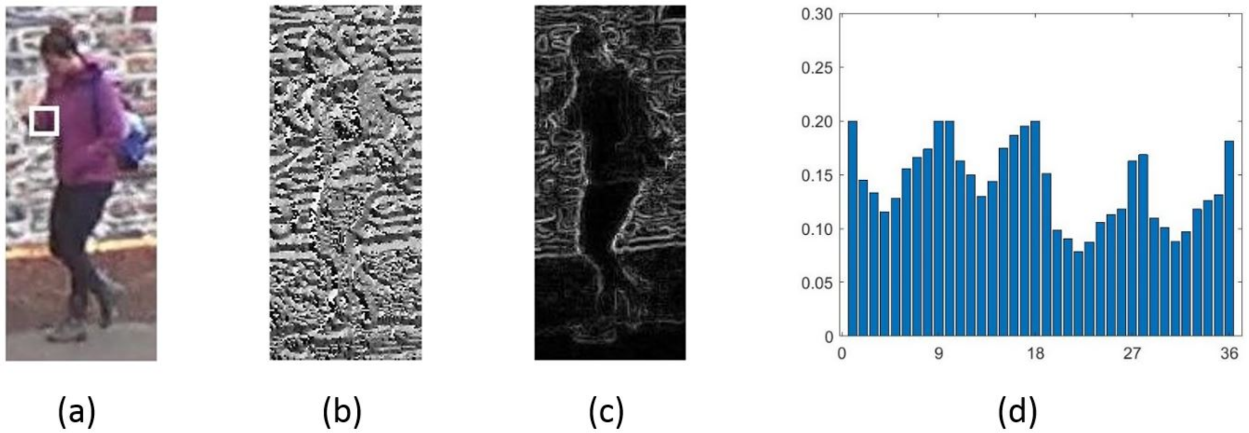


Figure 4.4: Sample of image gradient and HOG descriptor.

direction and magnitude can depict the boundary of objects precisely. Thus, the statistical properties of gradient could be used to depict the characteristics of an image. We calculate 4 related features: (1) meanGmag: the average of gradient magnitude; (2) stdGmag: the standard deviation of gradient magnitude; (3) meanGdir: the average of gradient direction; and (4) stdGdir: the standard deviation of gradient direction.

The local oriented gradient can describe object appearance and shape through counting occurrences of gradient orientation in localized portions of an image based on the HOG descriptor defined in [32]. The local window for one HOG descriptor is set as 16×16 pixels, and the number of orientation bins for one HOG descriptor is set as 9. For each histogram with 9 orientation bins, 4 different normalizations using adjacent histograms are employed, which results in a 36-dimensional feature vector. In order to summarize the information provided, we simplify the HOG descriptor through the average frequency w_m and the frequency's variation level w_s of the histogram's bins, which are defined to one window as follows:

$$w_m = \sum\nolimits_{i=1}^{N_b} h_i / N_b, \tag{4.6}$$

$$w_s = \sqrt{\sum\nolimits_{i=1}^{N_b} (h_i - w_m)^2 / (N_b - 1)}, \tag{4.7}$$

where h_i is the frequency of the i_{th} bin in a local window, and N_b is the number of bins in a local window. Based on two statistical values for one local window, we introduce 4 compact HOG features: (5) hog_mm: the average of every blocks' w_m; (6) hog_ms: the standard deviation of every blocks' w_m; (7) hog_sm: the average of every blocks' w_s; and (8) hog_ss: the standard deviation of every blocks' w_s.

The boundary or edge, representing transition areas between objects and background, is obtained by Sobel operator through convolving the image with two 3x3 kernels in the horizontal and vertical directions. The local information of edge is collected based on a block of 16×16 pixels, and 4 related features are calculated: (9) edge_mm: the average of every blocks' average; (10) edge_ms: the standard deviation of every blocks' average; (11) edge_sm: the average of every blocks' standard deviation; and (12) edge_ss: the standard deviation of every blocks' standard deviation.

If the size of an object is too small or too large in the image, it is hard to detect the object from the background. We introduce (13) estimated object size as another feature. This feature is obtained based on Otsu's method [136] with low computational overhead. Four feature channels are employed together to determine the optimal threshold for binarizing the gray image. In a gray image, let O_{T_i} and B_{T_i} denote the pixel sets of potential objects and background obtained with a threshold T_i. The optimal threshold T^* is determined through searching for a possible T_i that maximizes the total variance of four features between the potential objects and background according to

$$\begin{gathered} T^* = \underset{T_i}{\operatorname{argmax}} \sum_{j=1}^{N=4} \omega_O \omega_B (\overline{O^j_{T_i}} - \overline{B^j_{T_i}})^2, \\ s.t.\ 0 < \omega_O < 1,\ \omega_O + \omega_B = 1, \end{gathered} \tag{4.8}$$

where ω_O and ω_B denote the percentages of potential objects and background in the image, $\overline{O^j_{T_i}}$ and $\overline{B^j_{T_i}}$ represent the average values of the potential objects and background in feature channel j, $j \in \{f_{gm}, f_r, f_g, f_b\}$, and f_{gm}, f_r, f_g, and f_b are the gradient's magnitude, red color channel, green color channel, and blur color channel of the image, respectively.

We use the bootstrap aggregating, or bagging, ensemble of trees to train a regression model to predict detection performance based on the aforementioned 13 features on a single image. Every decision tree in the bagging ensemble is grown on an independently drawn bootstrap replica of input observations [137]. The ensemble tree prediction is formed by taking the average over base learners. The tuning parameters of ensemble trees include the number of trees and the minimum leaf size to control the tree depth.

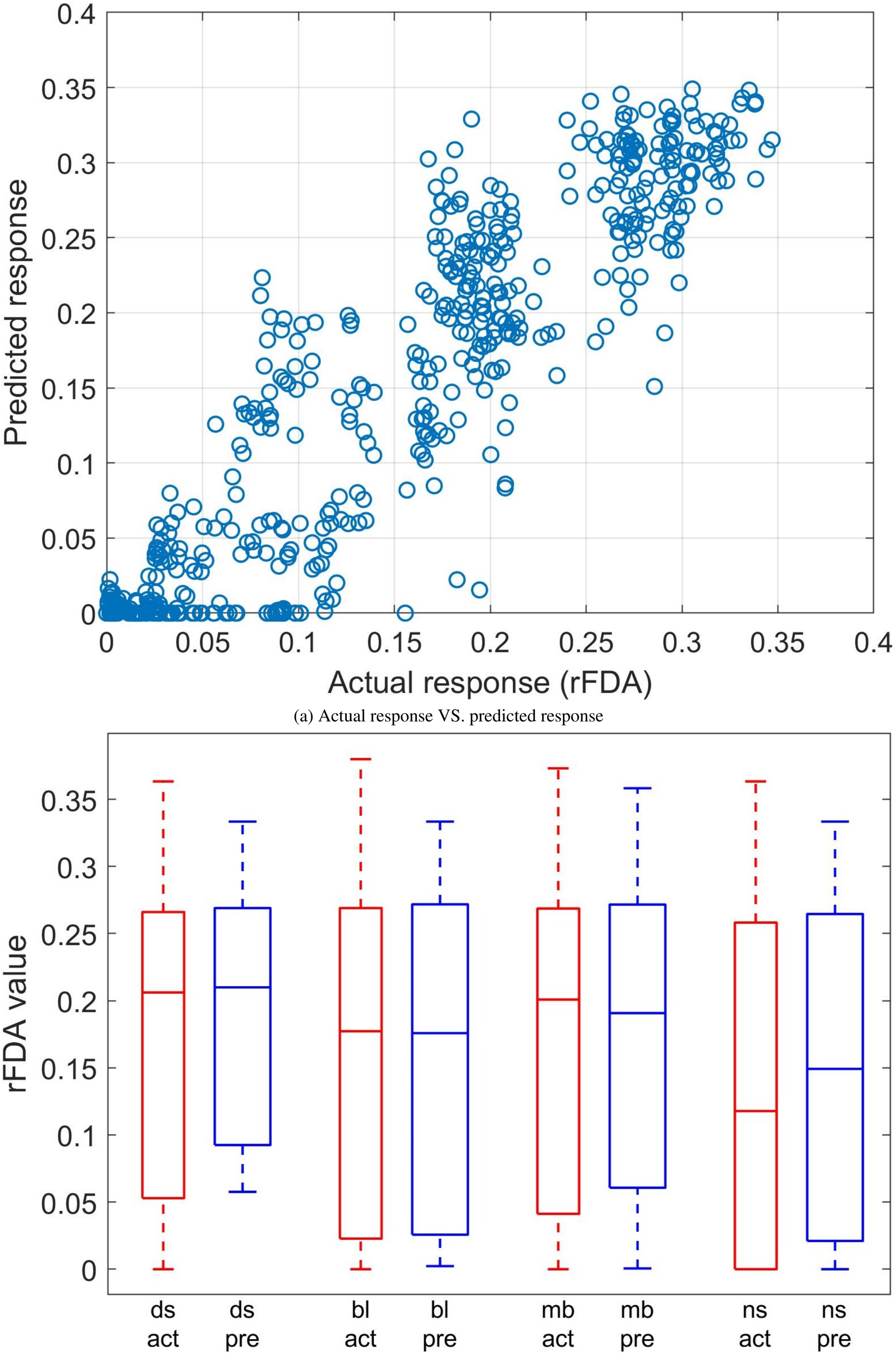
4.3 Performance evaluation

To evaluate the performance of the proposed quality model and classifiers of distortion types, we divide the entire data set into a training set and testing set, which are described in Table 4.3. The total number of images in our data set is 133344. The images from 8 raw videos and their distorted versions are used for training (75.03%), and the images from the remaining 2 raw videos and their distorted versions are used for testing. Through 5-fold cross validation during the training procedure, 30 base learners and a minimum leaf size of 8 are used to build the ensemble of trees for the proposed quality model.

Table 4.3: The setting of training and testing sets

Category	video name	image number	percentage
Training set	MOT17-02, MOT17-10, MOT17-13, MOT15-02, DMcam01, DMcam02, DMcam04, DMcam08	100044	75.03%
Testing set	MOT17-04, DMcam06	33300	24.97%

First, the regression performance of the proposed quality model is investigated. Figure 4.5 (a) shows the scatter figure of the actual response VS. the predicted response. There are a huge number of observations (33300 images) in the testing data set, and one point is selected from every 50 observations to generate a clear figure. The perfect regression results should be all on the diagonal line, and most of the predictions in our proposed model are near or on the diagonal line, which indicates that the regression of proposed model can depict the image quality for object



(a) Actual response VS. predicted response

(b) Distribution comparison in different distortion categories

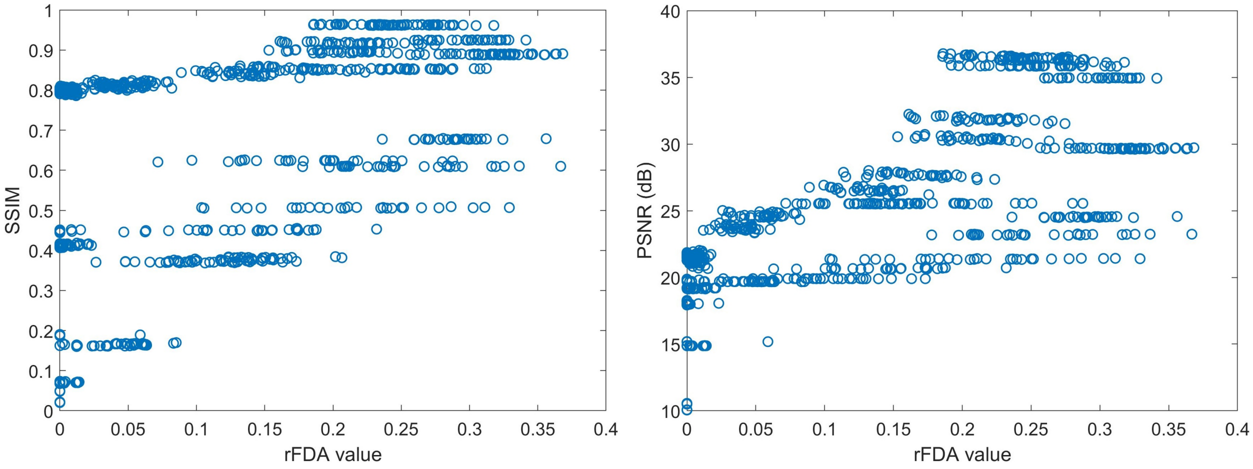
Figure 4.5: Regression performance of the proposed quality model.

Table 4.4: Regression metrics of the proposed quality model

Metrics	RMSE	R^2	$adjR^2$	MSE	MAE
Overall performance	0.0461	0.8416	0.8416	0.0021	0.0347
Down-sampling	0.0511	0.7896	0.7892	0.0026	0.0425
Blur	0.0412	0.8715	0.8713	0.0017	0.0293
Motion blur	0.0502	0.8104	0.8102	0.0025	0.0382
Noise	0.0433	0.8465	0.8463	0.0019	0.0320

detection quite well. Figure 4.5 (b) illustrates the distributions of the actual response and the predicted response in different categories of images, down-sampling in the spatial domain (ds), blur (bl), motion blur (mb), and imaging noise (ns), in which the actual responses (act) are in red color and the predicted responses (pre) are in blue color with wider boxes. We can find that the 25_{th} and 75_{th} quartiles and the medians of the predicted responses are all close to the actual responses in the distribution of the four categories, indicating that the proposed model can accurately predict image quality for object detection for different types of distortions.

The regression performance of the proposed quality model on the testing data set is measured in terms of Root Mean Square Error (RMSE), R^2, $adjR^2$, Mean Squared Error (MSE), and Mean Absolute Error (MAE), as shown in Table 4.4. Among these metrics, smaller values of RMSE, MSE and MAE indicate better performance. R^2, or coefficient of determination, is always smaller than 1 and usually larger than 0. Adjusted R^2, short for $adjR^2$, adjusts R^2 for the number of explanatory terms (features) in a model relative to the number of observations. R^2 and $adjR^2$ values close to 1 indicates good regression performance. From Table 4.4, we can find that the overall performance in terms of RMSE, MSE and MAE are all quite close to 0, and both R^2 and $adjR^2$ reaches 0.8416 after rounding, which indicates that the proposed model fits data well and that only a few features can explain the observations. The performances of specific distortion categories are also inspected and summarized in Table 4.4. For the blur and the noise categories, the values of RMSE, MSE, and MAE are all less than the ones of the overall performance, and the values of R^2 and $adjR^2$ are larger than the ones of the overall performance. For the down-sampling category, the values of RMSE, MSE, and MAE are slight larger to the ones of the overall



(a) rFDA VS. SSIM

(b) rFDA VS. PSNR

Figure 4.6: Full-reference IQAs performance.

performance, and the values of R^2 and $adjR^2$ are close to 0.8. For the motion blur category, the values of RMSE, MSE, and MAE are close to the ones of the overall performance, and both of R^2 and $adjR^2$ values reach above 0.81. Generally speaking, the proposed model can handle different distortion categories and achieve a decent overall performance.

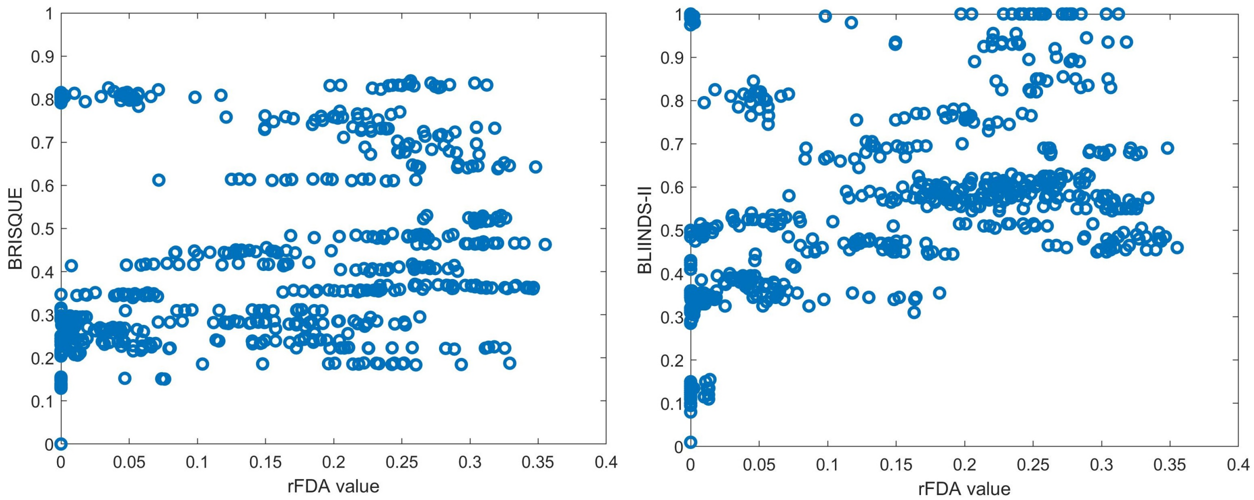
The performance of the proposed model is compared with two well-known full-reference IQAs: PSNR and SSIM. The performance of these models are evaluated in terms of the Linear Correlation Coefficient (LCC), the Spearman Rank Order Correlation Coefficient (SROCC), and the Kendall Rank Correlation Coefficient (KRCC). Because full-reference PSNR and SSIM measures could not evaluate the quality of down sampling versions and original video sequences, results from these images are excluded in this comparison. The correlation results are shown in Table 4.5. The correlation coefficients of LCC and SROCC for the proposed model reach above 0.90, while the ones for SSIM and PSNR fall between 0.6 and 0.8; the correlation coefficients of KRCC for the proposed model also reach above 0.70, which is 17.9% and 40.8% higher over the ones for PSNR and SSIM, respectively. The results show that the proposed model is a good predictor for the image quality for object detection, and SSIM and PSNR can not be good indicators for the image quality for object detection. The conclusion also can be drawn from Figure 4.6, in which scatter figures between PSNR, SSIM and rFDA values are plotted. From Figure 4.6, we can find that there is no significant relationship between either PSNR or SSIM and rFDA values. The reason is that

SSIM and PSNR are designed for the perceptual quality but not for the quality evaluated by object detection algorithms.

Table 4.5: Full-reference correlation coefficients

Algorithms	LCC	KRCC	SROCC
SSIM	0.6187	0.5198	0.7068
PSNR	0.7792	0.6208	0.8165
Proposed	0.9205	0.7319	0.9049

The proposed quality model is also evaluated against two popular no-reference IQAs: BRISQUE and BLIINDS-II. BRISQUE [138] is a distortion-generic no-reference IQA model, which exploits scene statistics of locally normalized luminance coefficients in spacial domain to quantify possible losses of naturalness in the image. BLIINDS-II [25] is a blind IQA algorithm using a Bayesian inference approach on extracted features that are based on a natural scene statistics model of discrete cosine transformation coefficients. All images in the testing set are included in this comparison thanks to the no-reference property of these two IQAs. Since both algorithms regard a quality score (QS) of 100 as the worst quality, and a QS of 0 as the best quality, we convert the QS using $QS_{new} = 1 - QS/100$ and compare QS_{new} from the two algorithms with the predictions of rFDA from our proposed quality model. The correlation results are presented in Table 4.6. For the proposed model, compared with the results obtained from the reduced testing set (shown in Table 4.5), the correlation coefficients on this complete testing set keep the same magnitude order, just slight decreasing with 0.2%, 1.6%, and 0.5% for LCC, KRCC, and SROCC, respectively. This indicates that the proposed model can also achieve good performance on the original videos and the down-sampled versions. The correlation coefficients of BRISQUE and BLIINDS-II are quite low, among them, the maximum value is 0.6007 and the mimnum value is 0.3737. The scatter figures between BRISQUE, BLIINDS-II and rFDA values are shown in Figure 4.7, in which a certain level of perceptual quality indicated by BRISQUE or BLIINDS-II could correspond to a diverse range of rFDA values. These results indicate that the proposed model is a good image quality estimator of object detection for various kinds of distoriton; however, BRISQUE and BLIINDS-II



(a) rFDA VS. BRISQUE

(b) rFDA VS. BLIINDS-II

Figure 4.7: No-reference IQAs performance.

Table 4.6: No-reference correlation coefficients

Algorithms	LCC	KRCC	SROCC
BRISQUE	0.4371	0.3737	0.5342
BLIINDS-II	0.5392	0.4337	0.6007
Proposed	0.9184	0.7204	0.9002

are limited in predicting image quality for object detection since they are intended for predicting perceptual quality.

It is important to factor computational complexity into an algorithm selection for embedded cameras, which are usually constrained in processing capabilities. Three algorithms, i.e., BLIINDS-II, BRISQUE, and our proposed one, are inspected. Computational complexity is measured exclusively on a computer based on an Intel Xeon E5-2637 v3 (3.50GHz) processor running on a Windows 7 Enterprise operating system. It is evaluated by the average computational time over 800 images of the original 1080P resolution (1920×1080), half resolution (960×540), and quarter resolution (480×270) of DMcam06 video. In Table 4.7, it can be noticed that the time consumption is significantly long of up to 159 seconds for the BLIINDS-II algorithm to compute a quality score of each original-sized input image; meanwhile the BRISQUE algorithm took approximately 0.785 seconds, which doubles the time of our proposed quality model. As input video

Table 4.7: Average computational complexity measured in second

Algorithms	1920×1080	960×540	480×270
BLIINDS-II	158.987 (±1.491)	40.065 (±0.698)	10.140 (±0.202)
BRISQUE	0.785 (±0.070)	0.324 (±0.061)	0.235 (±0.053)
Proposed	0.436 (±0.021)	0.210 (±0.011)	0.136 (±0.010)

resolution decreases, quality assessment time also drops. Using the proposed quality model, it takes only 0.21 seconds to process a half resolution frame and 0.136 seconds for a quarter resolution frame. In conclusion, among the three algorithms, our proposed quality model requires the least time for evaluating the quality of the input frames, which is suitable for implementation on embedded cameras.

4.4 Conclusion

In this chapter, we have proposed a no-reference image quality model based on a wide range of object detection algorithms that can be executed on embedded cameras. The proposed model could predict image quality for object detection by considering different types of quality degradation in the imaging process, including reduced resolution, noise, and blur. The proposed model is built based on a diverse range of scene characteristics. Utilizing easily extracted local and global features, the model achieves more accurate predictions of image quality for object detection than common full-reference image quality measures, such as PSNR and SSIM, and popular no-reference IQAs.

Chapter 5

Image quality adjustment framework for object detection on embedded cameras

In this chapter, we propose an image quality adjustment framework to provide satisfactory object detection performance for imaging applications based on embedded cameras. The framework includes a blind regression model based on a bagging ensemble of trees for predicting the performance of object detection on an image and two distortion type classifiers based on support vector machines for determining whether or not there is noise or blur in the image. The regression model and the classifiers utilize local features in an image such as edge and oriented gradient and global features including image gradient, image contrast, and estimated object size. All of the features could be easily obtained from an image, providing a light-weight solution for embedded cameras. The regression model and the classifiers are trained using a large number of images with different scene characteristics and three types of distortions including noise, out-of-focus blur, and motion blur. Their performances are evaluated through extensive experiments on a separate test data set. The effectiveness of the entire image quality adjustment framework is also evaluated using images with different types and levels of distortions. Our preliminary results on the regression model has been presented in Chapter 4. New contributions in this article include: a systematic image quality adjustment framework, two distortion type classifiers, and comprehensive experimental results and discussions based on a larger data set.

There is a rich literature on enhancing the perceptual quality of images. An image fusion and enhancement framework based on spectral total variation was proposed in [139]. This framework extracted the main features of the input images and achieved improvement for edge details and contrast. For low-light image enhancement, the robust Retinex model in [140] additionally considered a noise map compared with the conventional Retinex model, to improve the performance of enhancing low-light images accompanied by intensive noise. A guided image contrast enhancement framework based on retrieved images in cloud was proposed in [141], in which context-sensitive contrast and context-free contrast were jointly improved via solving a multi-criteria optimization problem. A no-reference IQA model through analysis of contrast, sharpness, brightness and etc., was proposed in [142]. Then, a robust image enhancement framework through histogram modification to rectify image brightness and contrast is established based on the NR-IQA model.

There are a few works on adjusting image quality for automatic analysis tasks. In [72], based on the quantification of the degradation in the performance of face detectors, a new set of features were proposed for robust face detection that could augment face-indicative features with perceptual quality-aware spatial natural scene statistics features. A similar enhancement strategy, based on aggregating IQA features that are more robust to image quality degradation, was employed by face recognition in infrared images in [71]. A closed-loop computing framework of enhancing image steganography through optimizing picture quality was proposed in [143]. For poor license plate recognition due to low quality of images, a new mathematical model based on Riesz fractional operator for enhancing details of edge information in license plate images was proposed to improve the performances of text detection and recognition methods in [144].

The aforementioned studies proposed quality enhancing or control approaches for either perceptual quality or specific applications like face recognition, image steganography, and plate recognition. However, two more challenges in image quality adjustment remain unsolved. First, there are different types of distortion during the imaging process, and the type of distortion (e.g., noise or blur) should be determined before applying image enhancement algorithms. Second, the estimation of distortion should be achieved through low complexity considering the limitation of processing on embedded cameras. Existing studies on the estimation of noise and blur involves intensive computation [145, 146]. Our work aims to advance the state of the art by addressing these two challenges.

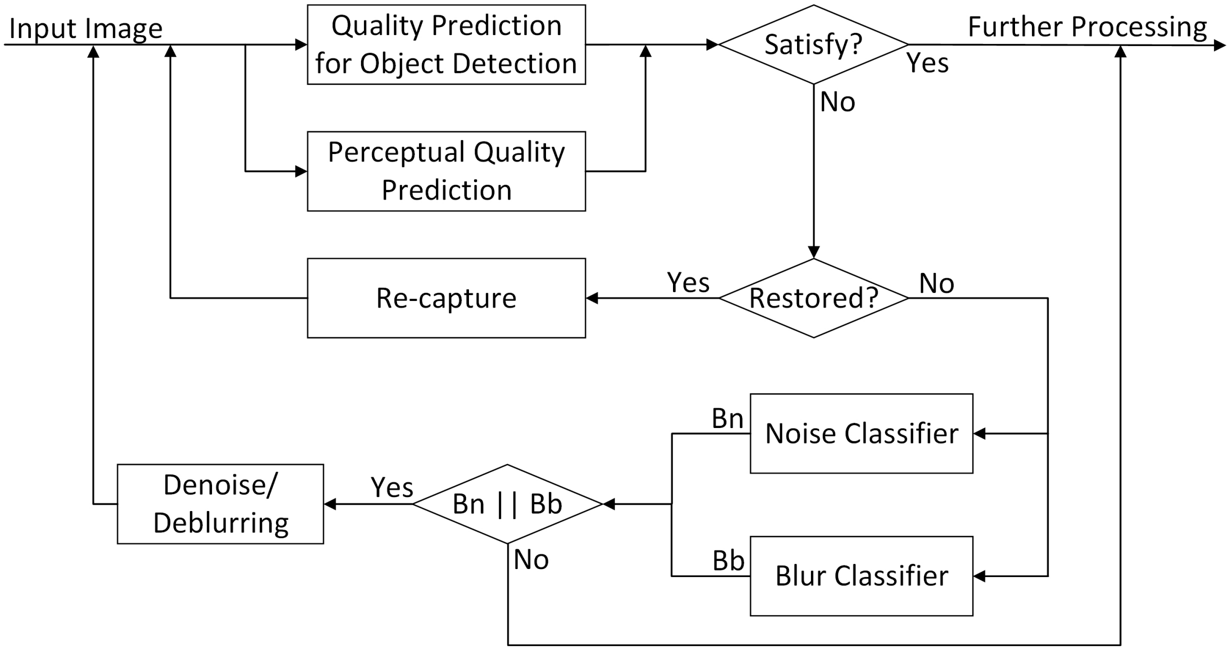


Figure 5.1: The quality adjustment framework.

5.1 Proposed image quality adjustment framework for object detection

We propose a quality adjustment framework to provide satisfactory image quality for object detection during the image sensing process. The components of the framework are shown in Figure 5.1. Once an image is captured, both the quality for object detection and the perceptual quality of the image are estimated, as the image may be used for further automatic analysis or be delivered in front of human users. The prediction of perceptual quality could be achieved through existing no-reference perceptual quality models, such as BRISQUE [138] and BLIINDS-II [25]; however, the evaluation of object detection quality requires further studies. We propose to build a new quality model for predicting the performance of object detection. If the image has satisfactory perceptual and object detection quality, it will be further processed or analyzed. Otherwise, the framework will determine if there is any noise or blur in the image, which are common types of distortions in the imaging process. If so, pre-processing methods for removing noise or blur will be applied to enhance its quality. If the restored image still cannot provide satisfactory quality, an image re-capturing action will be executed. Although several existing denoise and deblurring algorithms could be applied here to restore the images [147, 148, 149, 150], there is a lack of mechanisms to distinguish noisy or blurred images with normal ones. To solve this problem, we propose to build a noise classifier and a blur classifier.

In the rest of this section, we explain the three core components in the proposed framework: quality prediction for object detection, blur classifier, and noise classifier. Using the aforementioned data set, we apply supervised learning algorithms to build these components. We introduce a total number of 18 local and global features, all of which could be obtained from an image with low computational complexity. Figure 5.2 illustrates the high level relationship of the features. All of them are extracted from the converted gray image and the RGB image, and they can be shared among the three core components. The "Edge" and "Compact HOG" modules with grids in the background denote that these features are collected locally. Other features are summarized over an entire image.

5.1.1 Features for blur classification

Blur, including motion blur and out-of-focus blur, can smooth the boundary information in an image. Thus, statistical features based on image gradient, a good indicator for the variance of image intensities, are extracted from the gray image. Except for the average and standard deviation previously introduced in Section 4.2, we introduce two concepts known as Skewness and Kurtosis to better describe the nature of the distribution. Skewness means lack of symmetry. A distribution, or data set, is symmetrical when the data points are uniformly distributed around the mean. The

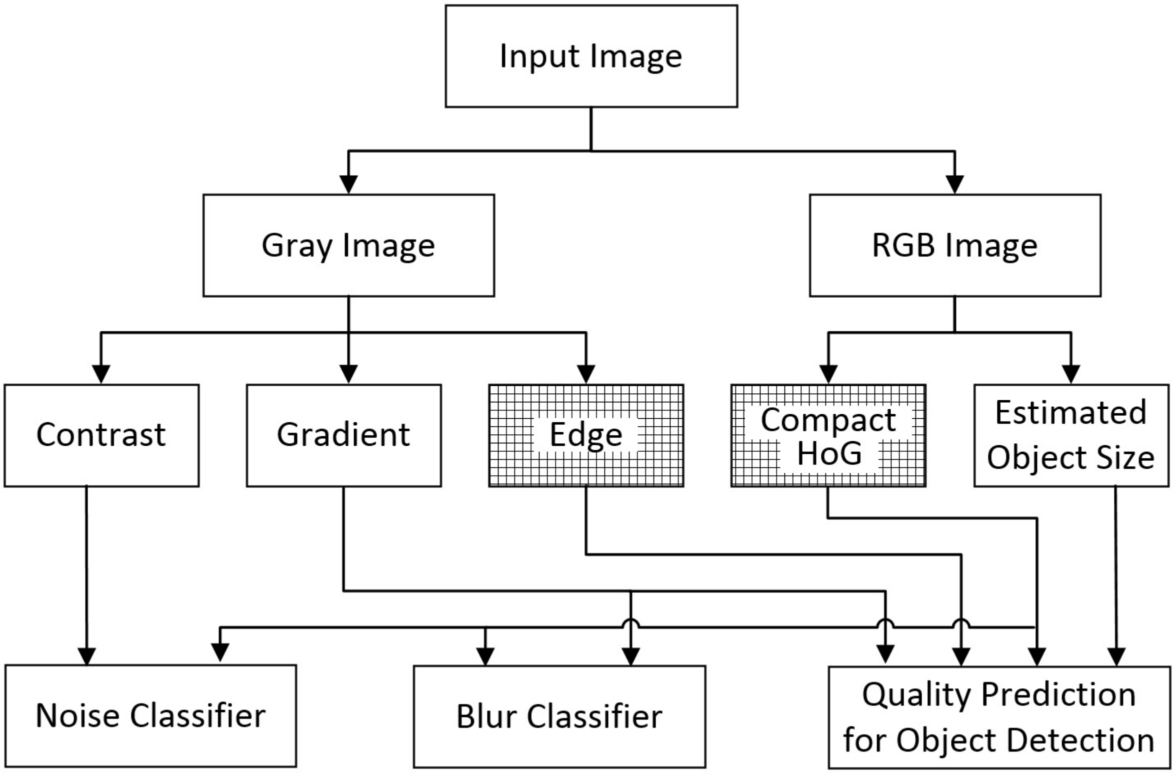


Figure 5.2: The architecture of feature extraction.

skewness of a random variable X is the third standardized moment, defined as:

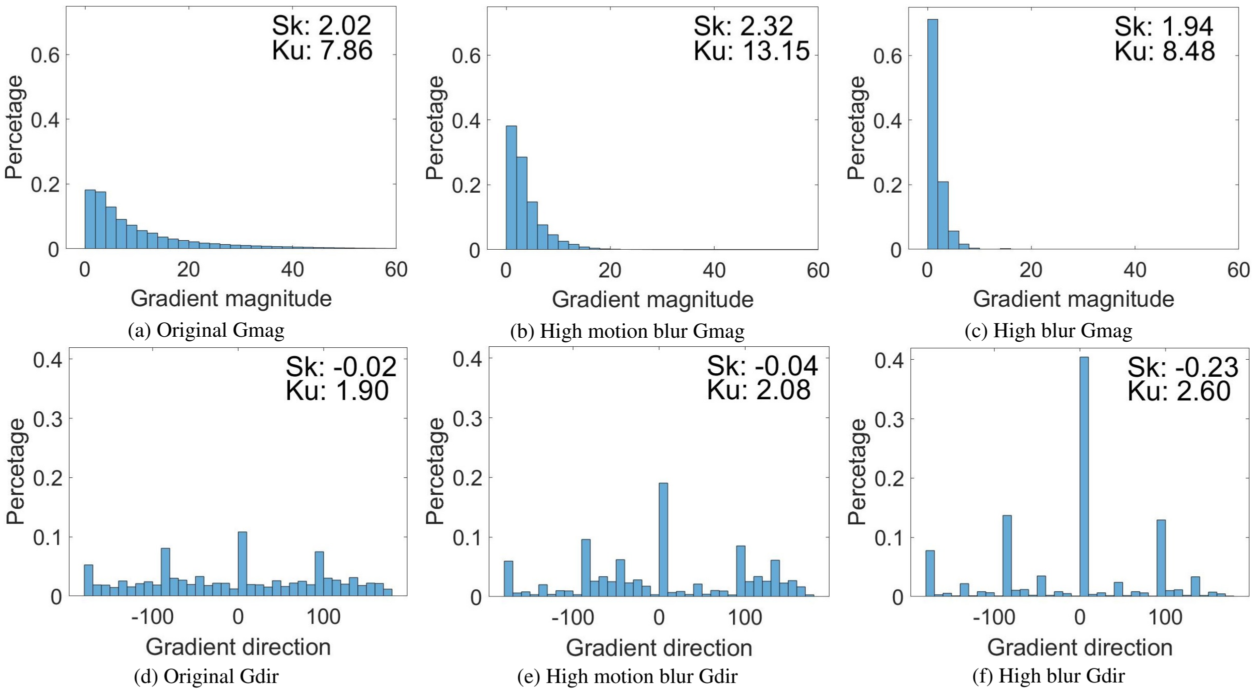
$$s = \frac{E(x-\mu)^3}{\sigma^3}, \tag{5.1}$$

where μ is the mean of x, σ is the standard deviation of x, and $E(t)$ represents the expected value of the quantity t.

A measure of the peakness or convexity relative to a normal distribution is known as Kurtosis. A high kurtosis value indicates that the distribution has heavy tails, or more outliers, and a low kurtosis value means the data set has light tails, or lack of outliers. The kurtosis of a random variable X is the fourth standardized moment, defined as:

$$k = \frac{E(x-\mu)^4}{\sigma^4}, \tag{5.2}$$

where μ is the mean of x, σ is the standard deviation of x, and $E(t)$ represents the expected value



(a) Original Gmag (b) High motion blur Gmag (c) High blur Gmag

(d) Original Gdir (e) High motion blur Gdir (f) High blur Gdir

Figure 5.3: Distribution comparison of original and blur images.

of the quantity t.

One sample of histograms for gradient of the original and the blur images is shown in Figure 5.3. The first row is the histograms of gradient magnitude (Gmag) for the original image, high blur version, and high motion blur version. The second row is the corresponding histograms of gradient direction (Gdir) calculated from images. The skewness (Sk) and kurtosis (Ku) for gradient's direction and magnitude are calculated and shown on the top-right corner of each figure. We can notice that the shape/distribution for the histograms of blur version is different from the one of the original image, and such difference is also reflected on the skewness and the kurtosis values. Thus, we calculate 4 more related global features based on the image gradient: (14) skewGmag: the skewness of gradient magnitude; (15) kurtGmag: the kurtosis of gradient magnitude; (16) skewGdir: the skewness of gradient direction; and (17) kurtGdir: the kurtosis of gradient direction.

5.1.2 Features for noise classification

Noise has always been associated with image acquisition equipments and it is a setback for object detection and further analysis or processing. Due to the randomness of noise, it can cause arbitrary changes of intensities locally, which will bring more inconsistency of intensities compared with normal or natural images. Thus, we utilize one feature, (18) image contrast, to depict this inconsistency. The image contrast is defined as:

$$C = \sqrt{\sum_{x=1}^{M} \sum_{y=1}^{N} (I(x,y) - \mu)^2 / MN}, \tag{5.3}$$

where $I(x,y)$ denotes the intensity value of the gray image at location (x,y), μ is the mean value of the entire gray image, and $M \times N$ stands for image size.

5.1.3 Binary classifiers for blur and noise

We propose to build binary classifiers to indicate whether or not there is blur or noise in an image. Considering the requirements of high accuracy and good robustness as well as the property of binary classification, we train two Support Vector Machines (SVM) as the blur classifier and the noise classifier. An SVM can construct an optimal hyperplane in high-dimensional space as a decision surface such that the margin of separation between the two classes in the data set is maximized [151]. The mechanism of finding the maximal margin can bring good tolerance for the classifier, which is a key point in our proposed framework. The training procedures of an SVM includes transforming input data to a high-dimensional feature space using a kernel, and solving a quadratic optimization problem to find an optimal hyperplane to classify the transformed features into two classes.

Finally, the blur classifier is established based on 8 gradient related features, i.e., feature No. (1)-(4) and (14)-(17), and 4 compact HOG features, i.e., feature No. (5)-(8). The noise classifier is constructed based on 1 image contract feature namely feature No. (18) and 4 compact HOG features namely feature No. (5)-(8).

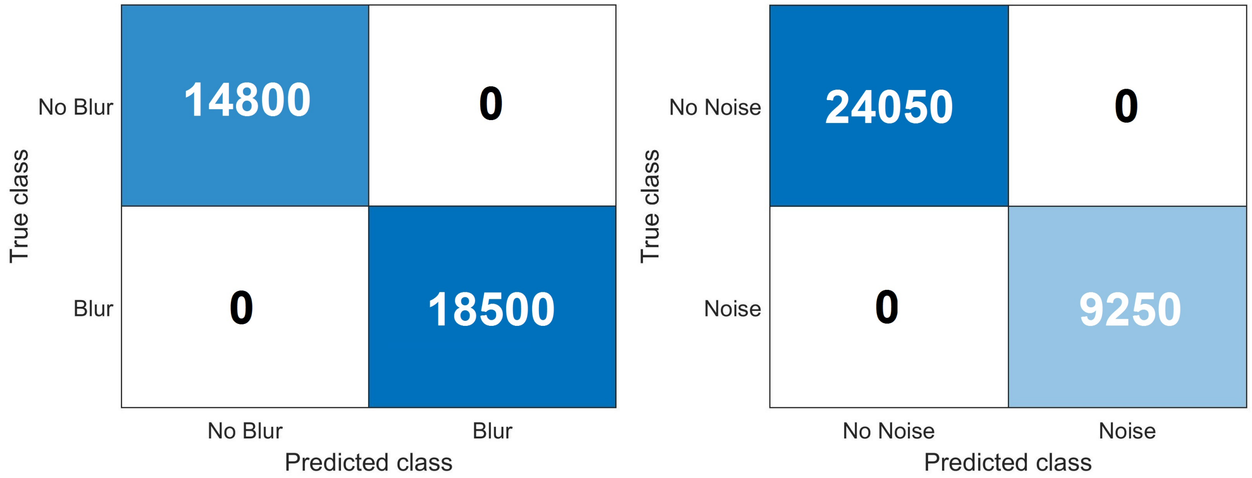
5.2 Performance evaluation

To evaluate the performance of the proposed quality model and classifiers of distortion types, we divide the entire data set into a training set and testing set, which are described in Table 4.3. The total number of images in our data set is 133344. The images from 8 raw videos and their distorted versions are used for training (75.03%), and the images from the remaining 2 raw videos and their distorted versions are used for testing. Through 5-fold cross validation during the training procedure, 30 base learners and a minimum leaf size of 8 are used to build the ensemble of trees for the proposed quality model. For two classifiers of distortion types, the box constraint parameter with 1 and the standardized predictor data are utilized to train two linear kernel-based SVMs.

5.2.1 Evaluation of blur and noise classifiers

We use the same training set and testing set to train and test two linear kernel-based SVMs for the proposed blur and noise classifiers. The confusion matrices of the two classifiers are shown in Figure 5.4. The total amount of test images is 33300, which includes original images, 2 versions of down-sampling in spatial resolution, 5 distorted levels of out-of-focus blur, motion blur, and imaging noise. Since there are two kinds of blur and one type of noise, the amount of blur images is twofold the amount of noise image. In such a diversified data set, both the noise classifier and blur classifier can precisely distinguish noise and blur distortions among other interference factors. The accuracies for both classifiers reach to 100%. The accurate classifications can help to determine whether or not denoise and/or deblurring algorithms could be applied to restore a distorted image.

All the features in the blur and noise classifiers could be obtained through light-weight computation, and it is worth noticing that, the majority of the features for the classifiers are reused from the features extracted for the proposed quality model. The entire computational complexity of the proposed quality model and two classifiers is also tested on the same configuration and machine used in Section **??**. The averaging time consumptions of the proposed quality model and two clas-



(a) Confusion matrix of blur classifier

(b) Confusion matrices of noise classifier

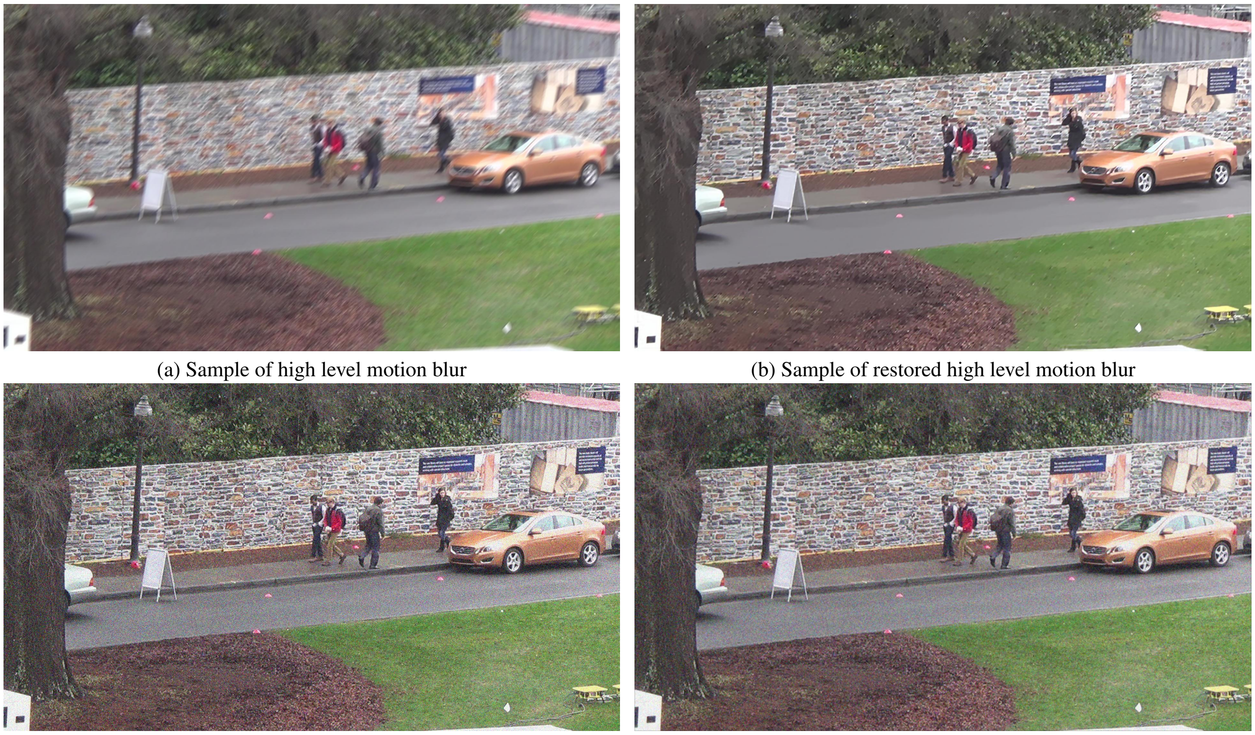
Figure 5.4: Classification results for two classifier of distortion types.

sifiers for the original 1080p, half, and quarter resolution are 0.473 (±0.028), 0.245 (±0.017), and 0.153 (±0.014) seconds, respectively. Comparison with the time consumptions in Table 4.7, there is only slight increasement over the proposed quality model and it is still less than the BRISQUE algorithm, which indicates that the proposed classifiers can be implemented on embedded cameras with low complexity.

5.2.2 Evaluation of image quality adjustment

After determining the distortion type, the quality adjustment framework can call appropriate image restore algorithms, i.e., denoise and deblurring, to adjust image quality. For the denoise, Wiener filter is deployed to suppress noise and preserve edge, texture, and other high frequency details based on noise level estimation using [145]. For the deblurring, we employ an efficient method via dark channel prior in [152], which is based on the observation that the dark channel, i.e., the minimum value in an image patch, of blurred is less sparse.

From each of the 10 scenes shown in Figure 4.1, one image frame is randomly selected, and the corresponding distorted images with five levels of distortion are restored. Samples of distorted images and restored images are shown in Figure 5.5, in which the original image frame is the 1_{th} frame of DMcam01 video. Figure 5.5 (a) and (b) shows one pair of distorted image and



(a) Sample of high level motion blur

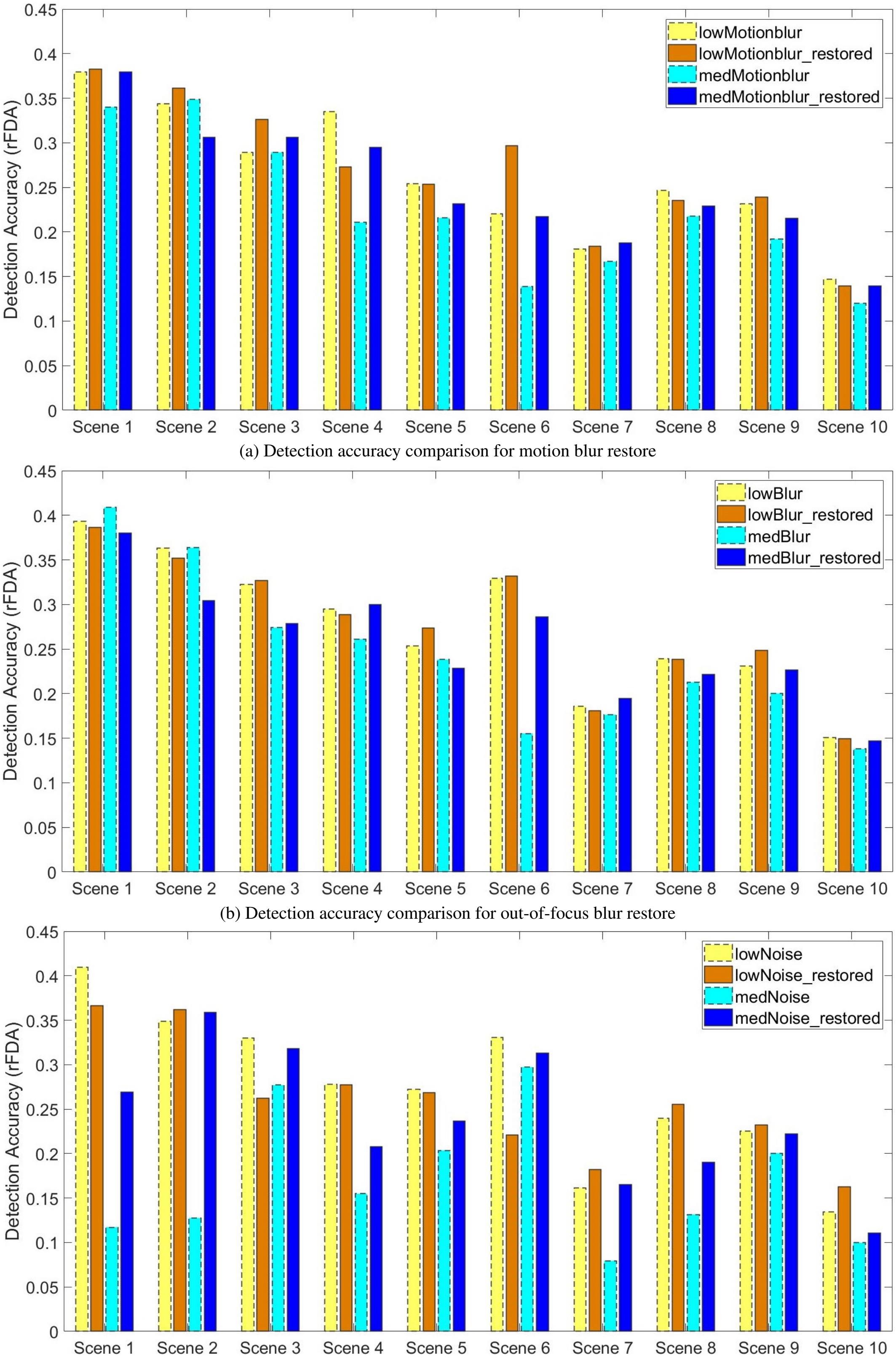
(b) Sample of restored high level motion blur

(c) Sample of high level noise

(d) Sample of restored high level noise

Figure 5.5: Sample of distorted images and restored images.

restored image for high level motion blur, we can find that there is significantly improvement for restored image. Figure 5.5 (c) and (d) shows one pair of distorted image and restored image for high level noise, we can notice that there is obvious improvement for restored image. Object detection performances on the distorted images and the restored images are compared. Due to space limit, results from only the low level and the medium level distortions are visualized in Figure 5.6. Results from high, higher, and extreme levels are consistent with the ones of low and medium levels. The results on distorted images are labeled with light color bars with dash line, and the ones on restored images are labeled with deep color bars with solid line. From Figure 5.6 (a), the restored motion blur images result in better object detection accuracy in most cases. Figure 5.6 (b) describes the detection performance on out-of-focus blur images, and overall the restored images performs better than the distorted ones except in rare cases. The reason for different gains in Figure 5.6 (a) and Figure 5.6 (b) is that the algorithm used performs better with motion blur than out-of-focus blur [152]. Figure 5.6 (c) presents the detection results on noisy



(a) Detection accuracy comparison for motion blur restore

(b) Detection accuracy comparison for out-of-focus blur restore

(c) Detection accuracy comparison for imaging noise restore

Figure 5.6: Detection accuracy comparisons for three distortion restore.

images and their restored counterparts, and the results indicate that restored images for the majority of scenes have higher detection accuracy than the distorted ones. From Figure 5.6, we can also find that the improvement on low level distortions is not as much as the ones on medium level distortions. This is because distortion at a low level does not degrade the performance of object detectors significantly. For most cases of the different scenes, the restored images produces a better detection accuracy performance. Specifically, the average improvements for restoring the five level distortions on detection accuracy are 72.26%, 18.93%, and 42.87% for motion blur, out-of-focus blur, and imaging noise, respectively. One reason for different ranges of gain is that the content characteristics of different scenes can also affect the object detection performance.

5.3 Conclusion

In this chapter, we have proposed an image quality adjustment framework with the objective to provide satisfactory object detection performance on embedded cameras. The core components of the framework are: a new image quality model that could predict the performance of object detection, a classifier for detecting out-of-focus and motion blur, and a classifier for detecting imaging noise. All the components are designed based on a data set that includes diverse scene characteristics and commonly used light-weight object detection algorithms. Utilizing easily extracted local and global features, we have designed a regression model for predicting quality based on the ensemble of trees and two classifiers for detecting blur and noise based on the SVM. Evaluation results have shown that the framework achieves accurate estimations of both image quality and image distortion types with low computational complexity. It has also been demonstrated that the framework could effectively enhance the performance of object detection on images captured by embedded cameras.

Chapter 6

Conclusion and future work

6.1 Research contributions

The impact of video quality on object detection and its applications are systematically investigated in the dissertation for compressed videos and local processing on embedded cameras.

For object detection on compressed videos, it has been found that current standardized video encoding schemes cause temporal domain fluctuation for encoded blocks in stable background areas and spatial texture degradation for encoded blocks in dynamic foreground areas of a raw video, both of which degrade the accuracy of object detection. Two measures, the sum-of-absolute frame difference (SFD) and the degradation of texture (TXD), are introduced to depict the temporal domain fluctuation and the spatial texture degradation in an encoded video, respectively. A model of object detection quality on compressed videos is established based on these two measures. Then we have proposed an efficient video encoding framework for boosting the accuracy of object detection for distributed sensing applications. The proposed encoding framework is designed to suppress unnecessary temporal fluctuation in stable background areas and preserve spatial texture in dynamic foreground areas based on the two measures, and it introduces new mode decision strategies for both intra and inter frames to improve the accuracy of object detection while maintaining an acceptable rate-distortion performance. Experimental results show that, compared with traditional encoding schemes, the proposed scheme improves the performance of object detection and results in lower bit rates and significantly reduced complexity with comparable quality in terms of PSNR and SSIM.

For object detection performed locally on embedded cameras, we have investigated the impact of imaging quality, such as imaging noise, motion blur, and out-of-focus blur, on the performance of distributed in-network video analysis. We have proposed a no-reference regression model based on a bagging ensemble of regression trees to predict the accuracy of object detection using observable features in an image. Based on the estimation of detection performance, we have proposed a quality adjustment framework to provide satisfactory object detection performance on embedded cameras. Key components of the framework include a blind regression model for predicting the performance of object detection and two classifiers for determining the type of distortion in an image. A video data set is constructed that considers different factors related to quality degradation in the imaging process. The performances of common low-complexity object detection algorithms are obtained for the data set. Based on the data set and utilizing features that can be easily extracted from an image, a regression model and two classifiers are trained and tested. The proposed framework achieves accurate estimations of both image quality and image distortion types with low computational complexity and it can effectively enhance the performance of object detection on embedded cameras.

6.2 Future work

Considering the constraint of energy on embedded cameras, the trade off between detection quality and energy consumed for local processing on individual cameras should be explored further. After object detection, object tracking and other high level analysis are employed. The quality of object tracking also should be investigated. Since the view of a single camera is finite and limited by scene structures, collaborative tracking scheme for wireless camera networks should be explored accordingly.

Quality-Rate-Energy optimization for local processing on individual cameras

Firstly, we can model the rate for local processing on wireless cameras based on extracted features. The amount of transmitted data is determined mainly by local processing algorithms, since different video processing algorithms generate different semantic information. Then, the semantic information is encoded by specific algorithms, and the encoded data is transmitted based on specific communication protocol. Then, we can optimize detection accuracy with rate and energy together for local processing on wireless cameras. In principle, this problem can be rephrased as – how to achieve the best object detection quality under the network bandwidth and battery power constraints at individual wireless camera sensors. The CPU can reduce its energy consumption substantially by running more slowly. Reducing the supply voltage in conjunction with the clock frequency eliminates the idle cycles and saves the energy significantly. Therefore, we can dynamically adjust the processing power for tracking for saving energy. For evaluation the performance of proposed Quality-Rate-Energy optimization scheme, we can try to perform simulation on certain software platforms, such as WiSE-Mnet++ [153], or conduct testbed experiments.

A quality-of-tracking model for collaborative object tracking using multiple cameras

The view of a single camera is finite and limited by scene structures. In order to monitor a wide area, such as tracking a person walking through the road network of a city, video streams from multiple cameras have to be used for collaborative object tracking.

Collaborative object tracking includes two scenarios: 1) one object appear in multiple cameras at the same time, i.e., multiple cameras in overlapping fields of view (FOVs); 2) one object appear sequentially in multiple cameras, i.e., multiple cameras in non-overlapping FOVs. For one object appearing in multiple cameras at the same time, which view of camera should be selected and when views of camera should be switched decide the collaborative object tracking quality. For one object appearing sequentially in multiple cameras, re-identification and relay tracking play a key part in the collaborative object tracking quality. In summary, tracking accuracy is always a key issue to be considered, while strategies of tracking in multiple cameras is also critical.

When the FOVs of different cameras overlap, not all cameras are equally needed for localizing a tracking target. It is possible to base the tracking on the observations of only a subset of cameras, where this subset is selected such that the associated drop in tracking quality is limited. When only these selected cameras do processing to track the target and data is transmitted only between the relevant cameras, a substantial saving of resources is achieved. The nonselected cameras can be left idle or can be used for other, additional network tasks. An example of an additional task in a camera network used for tracking is the discovery of new tracking targets.

Object tracking in nonoverlapping multiple cameras is more challenging because 1) the prediction of the spatio-temporal information of objects across camera views is much less reliable than in the same camera view; 2) the appearance of objects may undergo dramatic changes because of variations of many factors, such as camera settings, viewpoints and lighting conditions, in different camera views. The most typical way of multi-camera tracking is to track objects in a 3D coordinate system or on a single global ground plane or based on the homography between camera views after calibration. In order to track objects across disjoint camera views, appearance cues have to been integrated with spatio-temporal reasoning.

To study collaborative object tracking quality for multiple cameras, we can firstly search for video dataset, which should include two scenarios of collaborative object tracking. Then, we can choose several classical object tracking algorithms. Object tracking includes target representation scheme, search mechanism, and model update. Object representation is one of the major components in any visual tracking algorithm. Since objects have been detected before tracking, we can utilize detected objects' information to construct features that might affect the quality of tracking, such as: illumination conditions, changes of objects' appearance, changes in shape and size, targets motion, and occlusion.

A quality-aware collaborative object tracking scheme for wireless camera networks

According to the quality-of-tracking model for collaborative object tracking using multiple cameras, we can design a quality-aware collaborative object tracking scheme for wireless camera networks. This collaborative object tracking scheme can achieve better tracking accuracy and preserve more information of object for further analysis through calculating the quality of tracking in real-time and dynamical adjustment and allocation of computing resources in networks.

We cast collaborative tracking in wireless camera networks as *a resource allocation problem* where cameras are available network resources. The general target is to find the specific paradigm each active camera node should adopt and the related transmission rate to maximize the accuracy of the analysis task. The problem formulation explicitly considers bandwidth, energy, timeliness of response and routing constraints dictated by the individual nodes and network topology, as well as the costs of operating each camera node.

Bibliography

[1] Lingchao Kong, Ademola Ikusan, Rui Dai, Jingyi Zhu, and Dara Ros. A no-reference image quality model for object detection on embedded cameras. *International Journal of Multimedia Data Engineering and Management (IJMDEM)*, 10(1):22–39, 2019.

[2] Lingchao Kong, Ademola Ikusan, Rui Dai, and Jingyi Zhu. Blind image quality prediction for object detection. In *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 216–221. IEEE, 2019.

[3] Lingchao Kong and Rui Dai. Efficient video encoding for automatic video analysis in distributed wireless surveillance systems. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 14(3):72, 2018.

[4] Lingchao Kong, Jingyi Zhu, Rui Dai, and Mohammad Nazmus Sadat. Impact of distributed caching on video streaming quality in information centric networks. In *2017 IEEE International Symposium on Multimedia (ISM)*, pages 399–402. IEEE, 2017.

[5] Lingchao Kong and Rui Dai. Object-detection-based video compression for wireless surveillance systems. *IEEE MultiMedia*, 24(2):76–85, 2017.

[6] Lingchao Kong and Rui Dai. Temporal-fluctuation-reduced video encoding for object detection in wireless surveillance systems. In *Multimedia (ISM), 2016 IEEE International Symposium on*, pages 126–132. IEEE, 2016.

[7] Lingchao Kong, Rui Dai, and Yuchi Zhang. A new quality model for object detection using compressed videos. In *Image Processing (ICIP), 2016 IEEE International Conference on*, pages 3797–3801. IEEE, 2016.

[8] Mohammad Nazmus Sadat, Rui Dai, Lingchao Kong, and Jingyi Zhu. QoE-aware multi-source video streaming in content centric networks. *revised for journal publication*, April 2019.

[9] Lingchao Kong, Ademola Ikusan, Rui Dai, and Dara Ros. An image quality adjustment framework for object detection on embedded cameras. *submitted for journal publication*, June 2019.

[10] Yun Ye, Song Ci, Aggelos K Katsaggelos, Yanwei Liu, and Yi Qian. Wireless video surveillance: A survey. *IEEE Access*, 1:646–660, 2013.

[11] Weiming Hu, Tieniu Tan, Liang Wang, and Steve Maybank. A survey on visual surveillance of object motion and behaviors. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 34(3):334–352, 2004.

[12] Tao Ma, Michael Hempel, Dongming Peng, and Hamid Sharif. A survey of energy-efficient compression and communication techniques for multimedia in resource constrained systems. *Communications Surveys & Tutorials, IEEE*, 15(3):963–972, 2013.

[13] Lauro Snidaro, Ingrid Visentini, and Gian Luca Foresti. Fusing multiple video sensors for surveillance. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 8(1):7, 2012.

[14] Zhou Wang, Alan Conrad Bovik, Hamid Rahim Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *Image Processing, IEEE Transactions on*, 13(4):600–612, 2004.

[15] Zhou Wang and Alan C Bovik. Mean squared error: Love it or leave it? a new look at signal fidelity measures. *IEEE signal processing magazine*, 26(1):98–117, 2009.

[16] Mehul P Sampat, Zhou Wang, Shalini Gupta, Alan Conrad Bovik, and Mia K Markey. Complex wavelet structural similarity: A new image similarity index. *IEEE transactions on image processing*, 18(11):2385–2401, 2009.

[17] Lin Zhang, Lei Zhang, Xuanqin Mou, David Zhang, et al. Fsim: a feature similarity index for image quality assessment. *IEEE transactions on Image Processing*, 20(8):2378–2386, 2011.

[18] Anmin Liu, Weisi Lin, and Manish Narwaria. Image quality assessment based on gradient similarity. *IEEE Transactions on Image Processing*, 21(4):1500–1512, 2012.

[19] Wufeng Xue, Lei Zhang, Xuanqin Mou, and Alan C Bovik. Gradient magnitude similarity deviation: A highly efficient perceptual image quality index. *IEEE Transactions on Image Processing*, 23(2):684–695, 2014.

[20] Zhou Wang and Qiang Li. Information content weighting for perceptual image quality assessment. *IEEE Transactions on Image Processing*, 20(5):1185–1198, 2011.

[21] Wei Zhang, Ali Borji, Zhou Wang, Patrick Le Callet, and Hantao Liu. The application of visual saliency models in objective image quality assessment: A statistical evaluation. *IEEE transactions on neural networks and learning systems*, 27(6):1266–1278, 2016.

[22] Damon M Chandler. Seven challenges in image quality assessment: past, present, and future research. *ISRN Signal Processing*, 2013, 2013.

[23] Zhou Wang and Alan C Bovik. Reduced-and no-reference image quality assessment. *IEEE Signal Processing Magazine*, 28(6):29–40, 2011.

[24] Shiqi Wang, Ke Gu, Xinfeng Zhang, Weisi Lin, Siwei Ma, and Wen Gao. Reduced-reference quality assessment of screen content images. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(1):1–14, 2018.

[25] Michele A Saad, Alan C Bovik, and Christophe Charrier. Blind image quality assessment: A natural scene statistics approach in the dct domain. *IEEE transactions on Image Processing*, 21(8):3339–3352, 2012.

[26] Yuming Fang, Kede Ma, Zhou Wang, Weisi Lin, Zhijun Fang, and Guangtao Zhai. No-reference quality assessment of contrast-distorted images based on natural scene statistics. *IEEE Signal Processing Letters*, 22(7):838–842, 2015.

[27] Weilong Hou, Xinbo Gao, Dacheng Tao, and Xuelong Li. Blind image quality assessment via deep learning. *IEEE transactions on neural networks and learning systems*, 26(6):1275–1286, 2015.

[28] Kede Ma, Wentao Liu, Kai Zhang, Zhengfang Duanmu, Zhou Wang, and Wangmeng Zuo. End-to-end blind image quality assessment using deep neural networks. *IEEE Transactions on Image Processing*, 27(3):1202–1213, 2018.

[29] Alan J Lipton, Hironobu Fujiyoshi, and Raju S Patil. Moving target classification and tracking from real-time video. In *Applications of Computer Vision, 1998. WACV'98. Proceedings., Fourth IEEE Workshop on*, pages 8–14. IEEE, 1998.

[30] Chris Stauffer and W Eric L Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2, pages 246–252. IEEE, 1999.

[31] John L Barron, David J Fleet, and Steven S Beauchemin. Performance of optical flow techniques. *International journal of computer vision*, 12(1):43–77, 1994.

[32] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005.

[33] Yaakov Bar-Shalom. *Tracking and data association*. Academic Press Professional, Inc., 1987.

[34] Lawrence R Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[35] Michael Isard and Andrew Blake. Condensationconditional density propagation for visual tracking. *International journal of computer vision*, 29(1):5–28, 1998.

[36] Changjiang Yang, Ramani Duraiswami, and Larry Davis. Fast multiple object tracking via a hierarchical particle filter. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 1, pages 212–219. IEEE, 2005.

[37] Jingjing Xiao and Mourad Oussalah. Collaborative tracking for multiple objects in the presence of inter-occlusions. *IEEE Transactions on Circuits and Systems for Video Technology*, 26(2):304–318, 2016.

[38] Daw-Tung Lin and Kai-Yung Huang. Collaborative pedestrian tracking and data fusion with multiple cameras. *IEEE Transactions on Information Forensics and Security*, 6(4):1432–1444, 2011.

[39] Wasit Limprasert, Andrew Wallace, and Greg Michaelson. Real-time people tracking in a camera network. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 3(2):263–271, 2013.

[40] Wei Zhang, Bingpeng Ma, Kan Liu, and Rui Huang. Video-based pedestrian re-identification by adaptive spatio-temporal appearance model. *IEEE transactions on image processing*, 26(4):2042–2054, 2017.

[41] Aaron F Bobick and Andrew D Wilson. A state-based approach to the representation and recognition of gesture. *IEEE Transactions on pattern analysis and machine intelligence*, 19(12):1325–1337, 1997.

[42] Andrew D Wilson, AE Bobick, and Justine Cassell. Temporal classification of natural gesture and application to video coding. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 948–954. IEEE, 1997.

[43] Matthew Brand and Vera Kettnaker. Discovery and segmentation of activities in video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):844–851, 2000.

[44] Gabriella Csurka, Christopher Dance, Lixin Fan, Jutta Willamowski, and Cédric Bray. Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, volume 1, pages 1–2. Prague, 2004.

[45] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

[46] Christophe Bobda, Senem Velipasalar, et al. *Distributed Embedded Smart Cameras*. Springer, 2014.

[47] Bulent Tavli, Kemal Bicakci, Ruken Zilan, and Jose M Barcelo-Ordinas. A survey of visual sensor network platforms. *Multimedia Tools and Applications*, 60(3):689–726, 2012.

[48] J Boice, X Lu, C Margi, G Stanek, G Zhang, R Manduchi, and K Obraczka. Meerkats: A power-aware, self-managing wireless camera network for wide area monitoring. In *Proc. Workshop on Distributed Smart Cameras*, pages 393–422, 2006.

[49] Mohammad Rahimi, Rick Baer, Obimdinachi I Iroezi, Juan C Garcia, Jay Warrior, Deborah Estrin, and Mani Srivastava. Cyclops: in situ image sensing and interpretation in wireless sensor networks. In *Proceedings of the 3rd international conference on Embedded networked sensor systems*, pages 192–204. ACM, 2005.

[50] Stephan Hengstler, Daniel Prashanth, Sufen Fong, and Hamid Aghajan. Mesheye: a hybrid-resolution smart camera mote for applications in distributed intelligent surveillance. In *Proceedings of the 6th international conference on Information processing in sensor networks*, pages 360–369. ACM, 2007.

[51] Aliaksei Kerhet, Michele Magno, Francesco Leonardi, Andrea Boni, and Luca Benini. A low-power wireless video sensor node for distributed object detection. *journal of real-time image processing*, 2(4):331–342, 2007.

[52] Anthony Rowe, Dhiraj Goel, and Raj Rajkumar. Firefly mosaic: A vision-enabled wireless sensor networking system. In *Real-time systems symposium, 2007. RTSS 2007. 28th IEEE international*, pages 459–468. IEEE, 2007.

[53] Thiago Teixeira, Eugenio Culurciello, Joon Hyuk Park, Dimitrios Lymberopoulos, Andrew Barton-Sweeney, and Andreas Savvides. Address-event imagers for sensor networks: evaluation and modeling. In *Proceedings of the 5th international conference on Information processing in sensor networks*, pages 458–466. ACM, 2006.

[54] Phoebus Chen, Kirak Hong, Nikhil Naikal, S Shankar Sastry, Doug Tygar, Posu Yan, Allen Y Yang, Lung-Chung Chang, Leon Lin, Simon Wang, et al. A low-bandwidth camera sensor platform with applications in smart camera networks. *ACM Transactions on Sensor Networks (TOSN)*, 9(2):21, 2013.

[55] Youlu Wang, Senem Velipasalar, and Mauricio Casares. Cooperative object tracking and composite event detection with wireless embedded smart cameras. *IEEE Transactions on Image Processing*, 19(10):2614–2633, 2010.

[56] Mauricio Casares and Senem Velipasalar. Adaptive methodologies for energy-efficient object detection and tracking with battery-powered embedded smart cameras. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(10):1438–1452, 2011.

[57] UTC Fire Safety and Security. Available: `http://www.ccs.utc.com/ccs/en/worldwide/fire-security/`.

[58] Ying-li Tian, Lisa Brown, Arun Hampapur, Max Lu, Andrew Senior, and Chiao-fe Shu. Ibm smart surveillance system (s3): event based video surveillance system with an open and extensible framework. *Machine Vision and Applications*, 19(5):315–327, 2008.

[59] FLIR Intelligent Transportation Systems. Available: `http://www.flir.com/traffic/content/?id=66601`.

[60] DAHUA security. Available: `http://www.dahuasecurity.com/en/us/index.php`.

[61] Khalid Tahboub, Amy R Reibman, and Edward J Delp. Accuracy prediction for pedestrian detection. In *Image Processing (ICIP), 2017 IEEE International Conference on*, pages 4192–4196. IEEE, 2017.

[62] Chengzhang Zhong and Amy R Reibman. Prediction system for activity recognition with compressed video. *Electronic Imaging*, 2018(2):1–6, 2018.

[63] Hyomin Choi and Ivan V Bajic. High efficiency compression for object detection. *arXiv preprint arXiv:1710.11151*, 2017.

[64] Hyomin Choi and Ivan V Bajic. Deep feature compression for collaborative object detection. *arXiv preprint arXiv:1802.03931*, 2018.

[65] Xiang Chen, Jenq-Neng Hwang, Kuan-Hui Lee, and Ricardo L de Queiroz. Quality-of-content (QoC)-driven rate allocation for video analysis in mobile surveillance networks. In *Multimedia Signal Processing (MMSP), 2015 IEEE 17th International Workshop on*, pages 1–6. IEEE, 2015.

[66] Xiang Chen, Jenq-Neng Hwang, De Meng, Kuan-Hui Lee, Ricardo L de Queiroz, and Fu-Ming Yeh. A quality-of-content-based joint source and channel coding for human detections in a mobile surveillance cloud. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(1):19–31, 2017.

[67] Mikołaj Leszczuk. Optimising task-based video quality. *Multimedia Tools and Applications*, 68(1):41–58, 2014.

[68] Changchun Long, Yang Cao, Tao Jiang, and Qian Zhang. Edge computing framework for cooperative video processing in multimedia iot systems. *IEEE Transactions on Multimedia*, 20(5):1126–1139, 2018.

[69] Juan C SanMiguel and Andrea Cavallaro. Efficient estimation of target detection quality. In *Image Processing (ICIP), 2017 IEEE International Conference on*, pages 915–919. IEEE, 2017.

[70] Ayman Abaza, Mary Ann Harrison, and Thirimachos Bourlai. Quality metrics for practical face recognition. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 3103–3107. IEEE, 2012.

[71] Camilo G Rodríguez Pulecio, Hernán D Benítez-Restrepo, and Alan C Bovik. Image quality assessment to enhance infrared face recognition. In *Image Processing (ICIP), 2017 IEEE International Conference on*, pages 805–809. IEEE, 2017.

[72] Suriya Gunasekar, Joydeep Ghosh, and Alan C Bovik. Face detection on distorted images augmented by perceptual quality-aware features. *IEEE transactions on information forensics and security*, 9(12):2119–2131, 2014.

[73] Apurva Bedagkar-Gala and Shishir K Shah. Joint modeling of algorithm behavior and image quality for algorithm performance prediction. In *Proc. British Mach. Vis. Conf., Aberystwyth, UK*, 2010.

[74] John M Irvine and Eric Nelson. Image quality and performance modeling for automated target detection. In *Automatic Target Recognition XIX*, volume 7335, page 73350L. International Society for Optics and Photonics, 2009.

[75] John M Irvine and Richard J Wood. Real-time video image quality estimation supports enhanced tracker performance. In *SPIE Defense, Security, and Sensing*, pages 87130Z–87130Z. International Society for Optics and Photonics, 2013.

[76] John M Irvine, Richard J Wood, David Reed, and Janet Lepanto. Video image quality analysis for enhancing tracker performance. In *Applied Imagery Pattern Recognition Workshop (AIPR): Sensing for Control and Augmentation, 2013 IEEE*, pages 1–9. IEEE, 2013.

[77] Richard J Wood, David Reed, Brian Collins, and John M Irvine. Enhancing event detection in video using robust background and quality modeling. In *Video Surveillance and Transportation Imaging Applications 2014*, volume 9026, page 902609. International Society for Optics and Photonics, 2014.

[78] John M Irvine and Richard J Wood. Context and quality estimation in video for enhanced event detection. In *Airborne Intelligence, Surveillance, Reconnaissance (ISR) Systems and Applications XII*, volume 9460, page 94600L. International Society for Optics and Photonics, 2015.

[79] Koteswar Rao Jerripothula, Jianfei Cai, and Junsong Yuan. Qcce: Quality constrained co-saliency estimation for common object detection. In *Visual Communications and Image Processing (VCIP), 2015*, pages 1–4. IEEE, 2015.

[80] Syed Saud Naqvi, Will N Browne, and Christopher Hollitt. Feature quality-based dynamic feature selection for improving salient object detection. *IEEE Transactions on Image Processing*, 25(9):4298–4313, 2016.

[81] Nadeesh Fernando, Manjusri Wickramasinghe, Kasun De Zoysa, and Charitha Elvitigala. A quality metric for object detection and focus for low-cost uavs. In *Advances in ICT for Emerging Regions (ICTer), 2016 Sixteenth International Conference on*, pages 237–244. IEEE, 2016.

[82] Margaret H Pinson and Stephen Wolf. A new standardized method for objectively measuring video quality. *Broadcasting, IEEE Transactions on*, 50(3):312–322, 2004.

[83] Kalpana Seshadrinathan and Alan Conrad Bovik. Motion tuned spatio-temporal quality assessment of natural videos. *Image Processing, IEEE Transactions on*, 19(2):335–350, 2010.

[84] Ravi Soundararajan and Alan C Bovik. Video quality assessment by reduced reference spatio-temporal entropic differencing. *Circuits and Systems for Video Technology, IEEE Transactions on*, 23(4):684–694, 2013.

[85] Jingtao Xu, Peng Ye, Yong Liu, and David Doermann. No-reference video quality assessment via feature learning. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 491–495. IEEE, 2014.

[86] Ke Gu, Shiqi Wang, Huan Yang, Weisi Lin, Guangtao Zhai, Xiaokang Yang, and Wenjun Zhang. Saliency-guided quality assessment of screen content images. *IEEE Transactions on Multimedia*, 18(6):1098–1110, 2016.

[87] R. Guerrero-Gomez-Olmedo, R. J. Lopez-Sastre, S. Maldonado-Bascon, and A. Fernandez-Caballero. Vehicle tracking by simultaneous detection and viewpoint estimation. In *IWINAC 2013, Part II, LNCS 7931*, pages 306–316, 2013.

[88] David Thirde, Longzhen Li, and F Ferryman. Overview of the PETS2006 challenge. In *Proc. 9th IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS 2006)*, pages 47–50. Citeseer, 2006.

[89] Nils T Siebel and SJ Maybank. Real-time tracking of pedestrians and vehicles. In *IEEE Workshop on PETS*, volume 33, 2001.

[90] VideoLAN organization. x264, the best H.264/AVC encoder, 2005.

[91] Shih-Chia Huang and Bo-Hao Chen. Automatic moving object extraction through a real-world variable-bandwidth network for traffic monitoring systems. *Industrial Electronics, IEEE Transactions on*, 61(4):2099–2112, 2014.

[92] Andrews Sobral and Antoine Vacavant. A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. *Computer Vision and Image Understanding*, 122:4–21, 2014.

[93] Zoran Zivkovic and Ferdinand van der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern recognition letters*, 27(7):773–780, 2006.

[94] Andrew B Godbehere, Akihiro Matsukawa, and Ken Goldberg. Visual tracking of human visitors under variable-lighting conditions for a responsive audio art installation. In *American Control Conference (ACC), 2012*, pages 4305–4312. IEEE, 2012.

[95] Yannick Benezeth, Pierre-Marc Jodoin, Bruno Emile, Hélene Laurent, and Christophe Rosenberger. Comparative study of background subtraction algorithms. *Journal of Electronic Imaging*, 19(3):033003–033003, 2010.

[96] Axel Baumann, Marco Boltz, Julia Ebling, Matthias Koenig, HartmutS Loos, Marcel Merkel, Wolfgang Niem, JanKarl Warzelhan, and Jie Yu. A review and comparison of measures for automatic video surveillance systems. *EURASIP Journal on Image and Video Processing*, 2008(1):824726, 2008.

[97] R. V. Hogg and J. Ledolter. *Engineering Statistics*. MacMillan New York, 1987.

[98] MathWorks Inc. Local range of image-MATLAB rangefilt, 2006.

[99] Emmanouil Kafetzakis, Christos Xilouris, Michail Alexandros Kourtis, Marcos Nieto, Iveel Jargalsaikhan, and Suzanne Little. The impact of video transcoding parameters on event detection for surveillance systems. In *Multimedia (ISM), 2013 IEEE International Symposium on*, pages 333–338. IEEE, 2013.

[100] Pavel Korshunov and Wei Tsang Ooi. Video quality for face detection, recognition, and tracking. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 7(3):14, 2011.

[101] Thomas Wiegand, Gary J Sullivan, Gisle Bjontegaard, and Ajay Luthra. Overview of the h. 264/avc video coding standard. *IEEE Transactions on circuits and systems for video technology*, 13(7):560–576, 2003.

[102] Andrew D Bagdanov, Marco Bertini, Alberto Del Bimbo, and Lorenzo Seidenari. Adaptive video compression for video surveillance applications. In *Multimedia (ISM), 2011 IEEE International Symposium on*, pages 190–197. IEEE, 2011.

[103] RMTP Rajakaruna, WAC Fernando, and J Calic. Application-aware video coding architecture using camera and object motion-models. In *Industrial and Information Systems (ICIIS), 2011 6th IEEE International Conference on*, pages 76–81. IEEE, 2011.

[104] Amaya Jiménez-Moreno, Eduardo Martinez-Enriquez, Vipin Kumar, and Fernando Díaz-de María. Standard-compliant low-pass temporal filter to reduce the perceived flicker artifact. *Multimedia, IEEE Transactions on*, 16(7):1863–1873, 2014.

[105] Hua Yang, Jill M Boyce, and Alan Stein. Effective flicker removal from periodic intra frames and accurate flicker measurement. In *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, pages 2868–2871. IEEE, 2008.

[106] Seong Soo Chun, Jung-Rim Kim, and Sanghoon Sull. Intra prediction mode selection for flicker reduction in H. 264/AVC. *Consumer Electronics, IEEE Transactions on*, 52(4):1303–1310, 2006.

[107] Peng Wang, Yongfei Zhang, Hai-Miao Hu, and Bo Li. Region-classification-based rate control for flicker suppression of i-frames in HEVC. In *Image Processing (ICIP), 2013 20th IEEE International Conference on*, pages 1986–1990. IEEE, 2013.

[108] Hai-Miao Hu, Bo Li, Weiyao Lin, Wei Li, and Ming-Ting Sun. Region-based rate control for h. 264/avc for low bit-rate applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 22(11):1564–1576, 2012.

[109] Fan Zhang and David R Bull. A parametric framework for video compression using region-based texture models. *IEEE Journal of Selected Topics in Signal Processing*, 5(7):1378–1392, 2011.

[110] Jianshu Chao, Robert Huitl, Eckehard Steinbach, and Damien Schroeder. A novel rate control framework for sift/surf feature preservation in h. 264/avc video compression. *IEEE Transactions on Circuits and Systems for Video Technology*, 25(6):958–972, 2015.

[111] Xiang Zhang, Siwei Ma, Shiqi Wang, Xinfeng Zhang, Huifang Sun, and Wen Gao. A joint compression scheme of video feature descriptors and visual content. *IEEE Transactions on Image Processing*, 26(2):633–647, 2017.

[112] Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG. Working draft number 2, revision 0 (wd-2). JVT-B118, 2001.

[113] Yuming Fang, Zhenzhong Chen, Weisi Lin, and Chia-Wen Lin. Saliency detection in the compressed domain for adaptive image retargeting. *IEEE Transactions on Image Processing*, 21(9):3888–3901, 2012.

[114] Yuming Fang, Weisi Lin, Zhenzhong Chen, Chia-Ming Tsai, and Chia-Wen Lin. A video saliency detection model in compressed domain. *IEEE transactions on circuits and systems for video technology*, 24(1):27–38, 2014.

[115] Ee-Leng Tan and Woon-Seng Gan. Perceptual image coding with discrete cosine transform. In *Perceptual Image Coding with Discrete Cosine Transform*, pages 21–41. Springer, 2015.

[116] Eren Soyak, Sotirios Tsaftaris, Aggelos K Katsaggelos, et al. Low-complexity tracking-aware H. 264 video compression for transportation surveillance. *Circuits and Systems for Video Technology, IEEE Transactions on*, 21(10):1378–1389, 2011.

[117] Luis Patino, Tahir Nawaz, Tom Cane, and James Ferryman. Pets 2017: Dataset and challenge. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017.

[118] ITU-T RECOMMENDATION. P.910. *Subjective video quality assessment methods for multimedia applications*, pages 910–200804, 2008.

[119] Thomas Kuo, Zefeng Ni, Carter De Leo, and BS Manjunath. Design and implementation of a wide area, large-scale camera network. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 25–32. IEEE, 2010.

[120] Peter Corke, Tim Wark, Raja Jurdak, Wen Hu, Philip Valencia, and Darren Moore. Environmental wireless sensor networks. *Proceedings of the IEEE*, 98(11):1903–1917, 2010.

[121] Wan Du, Zhenjiang Li, Jansen Christian Liando, and Mo Li. From rateless to distanceless: Enabling sparse sensor network deployment in large areas. *IEEE/ACM Transactions on Networking*, 24(4):2498–2511, 2016.

[122] Danileno Rosário, José Arnaldo Filho, Denis Rosário, Aldri Santosy, and Mário Gerla. A relay placement mechanism based on uav mobility for satisfactory video transmissions. In *Ad Hoc Networking Workshop (Med-Hoc-Net), 2017 16th Annual Mediterranean*, pages 1–8. IEEE, 2017.

[123] Yen-Fu Ou, Zhan Ma, Tao Liu, and Yao Wang. Perceptual quality assessment of video considering both frame rate and quantization artifacts. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(3):286–298, 2011.

[124] Zhan Ma, Meng Xu, Yen-Fu Ou, and Yao Wang. Modeling of rate and perceptual quality of compressed video as functions of frame rate and quantization stepsize and its applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 22(5):671–682, 2012.

[125] Sebastian Brutzer, Benjamin Höferlin, and Gunther Heidemann. Evaluation of background subtraction techniques for video surveillance. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1937–1944. IEEE, 2011.

[126] Nil Goyette, Pierre-Marc Jodoin, Fatih Porikli, Janusz Konrad, and Prakash Ishwar. Changedetection. net: A new change detection benchmark dataset. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 1–8. IEEE, 2012.

[127] Anton Milan, Laura Leal-Taixé, Ian Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016.

[128] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *European Conference on Computer Vision workshop on Benchmarking Multi-Target Tracking*, 2016.

[129] Changxing Ding and Dacheng Tao. Trunk-branch ensemble convolutional neural networks for video-based face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):1002–1014, 2017.

[130] Kede Ma, Zhengfang Duanmu, Qingbo Wu, Zhou Wang, Hongwei Yong, Hongliang Li, and Lei Zhang. Waterloo exploration database: New challenges for image quality assessment models. *IEEE Transactions on Image Processing*, 26(2):1004–1016, 2016.

[131] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645, 2010.

[132] Woonhyun Nam, Piotr Dollár, and Joon Hee Han. Local decorrelation for improved pedestrian detection. In *Advances in Neural Information Processing Systems*, pages 424–432, 2014.

[133] Rangachar Kasturi, Dmitry Goldgof, Padmanabhan Soundararajan, Vasant Manohar, John Garofolo, Rachel Bowers, Matthew Boonstra, Valentina Korzhova, and Jing Zhang. Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):319–336, 2009.

[134] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[135] Piotr Dollar, Christian Wojek, Bernt Schiele, and Pietro Perona. Pedestrian detection: An evaluation of the state of the art. *IEEE transactions on pattern analysis and machine intelligence*, 34(4):743–761, 2011.

[136] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1):62–66, 1979.

[137] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[138] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing*, 21(12):4695–4708, 2012.

[139] Wenda Zhao, Huimin Lu, and Dong Wang. Multisensor image fusion and enhancement in spectral total variation domain. *IEEE Transactions on Multimedia*, 20(4):866–879, 2017.

[140] Mading Li, Jiaying Liu, Wenhan Yang, Xiaoyan Sun, and Zongming Guo. Structure-revealing low-light image enhancement via robust retinex model. *IEEE Transactions on Image Processing*, 27(6):2828–2841, 2018.

[141] Shiqi Wang, Ke Gu, Siwei Ma, Weisi Lin, Xianming Liu, and Wen Gao. Guided image contrast enhancement based on retrieved images in cloud. *IEEE Transactions on Multimedia*, 18(2):219–232, 2015.

[142] Ke Gu, Dacheng Tao, Jun-Fei Qiao, and Weisi Lin. Learning a no-reference quality assessment model of enhanced images with big data. *IEEE transactions on neural networks and learning systems*, 29(4):1301–1313, 2017.

[143] Guo-Shiang Lin, Yi-Ting Chang, and Wen-Nung Lie. A framework of enhancing image steganography with picture quality optimization and anti-steganalysis based on simulated annealing algorithm. *IEEE Transactions on Multimedia*, 12(5):345–357, 2010.

[144] KS Raghunandan, Palaiahnakote Shivakumara, Hamid A Jalab, Rabha W Ibrahim, G Hemantha Kumar, Umapada Pal, and Tong Lu. Riesz fractional based model for enhancing license plate detection and recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(9):2276–2288, 2017.

[145] Stanislav Pyatykh, Jürgen Hesser, and Lei Zheng. Image noise level estimation by principal component analysis. *IEEE transactions on image processing*, 22(2):687–699, 2013.

[146] Ruomei Yan and Ling Shao. Blind image blur estimation via deep learning. *IEEE Transactions on Image Processing*, 25(4):1910–1921, 2016.

[147] Jia-Li Yin, Bo-Hao Chen, and Ying Li. Highly accurate image reconstruction for multimodal noise suppression using semisupervised learning on big data. *IEEE Transactions on Multimedia*, 20(11):3045–3056, 2018.

[148] Sung In Cho and Suk-Ju Kang. Gradient prior-aided cnn denoiser with separable convolution-based optimization of feature dimension. *IEEE Transactions on Multimedia*, 21(2):484–493, 2019.

[149] Yuanchao Bai, Gene Cheung, Xianming Liu, and Wen Gao. Graph-based blind image deblurring from a single photograph. *IEEE Transactions on Image Processing*, 28(3):1404–1418, 2019.

[150] Yi Zhang and Keigo Hirakawa. Blind deblurring and denoising of images corrupted by unidirectional object motion blur and sensor noise. *IEEE Transactions on Image Processing*, 25(9):4129–4144, 2016.

[151] Nello Cristianini, John Shawe-Taylor, et al. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.

[152] Jinshan Pan, Deqing Sun, Hanspeter Pfister, and Ming-Hsuan Yang. Deblurring images via dark channel prior. *IEEE transactions on pattern analysis and machine intelligence*, 40(10):2315–2328, 2017.

[153] Juan C SanMiguel and Andrea Cavallaro. Networked computer vision: the importance of a holistic simulator. *Computer*, 50(7):35–43, 2017.