PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Neeraj Gadgil

Entitled Error Resilient Video Coding Using Bitstream Syntax and Iterative Microscopy Image Segmentation

For the degree of ___Doctor of Philosophy___

Is approved by the final examining committee:

EDWARD J. DELP

PAUL SALAMA

MARK R. BELL

MARY L. COMER

MICHAEL D. ZOLTOWSKI

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification/Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University's "Policy on Integrity in Research" and the use of copyrighted material.

EDWARD J. DELP

Approved by Major Professor(s): _____

Approved by: V. Balakrishnan   06/06/2016

Head of the Department Graduate Program   Date

ERROR RESILIENT VIDEO CODING USING BITSTREAM SYNTAX AND

ITERATIVE MICROSCOPY IMAGE SEGMENTATION

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Neeraj Jayant Gadgil

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

August 2016

Purdue University

West Lafayette, Indiana

This thesis is dedicated to the memory of my late father, Jayant N. Gadgil,
who was always proud of my achievements.

ACKNOWLEDGMENTS

Video and Image Processing Laboratory (VIPER) is an ideal place to work on challenging signal processing problems with a group of talented engineers. I have been very fortunate to be a part of this vibrant research environment. I would like to thank Dr. Meilin Yang who has been a great mentor to me and a very supportive teammate for the video coding project. I really appreciate her guidance during the early stages of my doctoral research. I want to thank Mr. Khalid Tahboub for being a great teammate for the crowdsourcing and the video signature projects. I appreciate his advice in many technical/management issues and his patience in dealing with my temper. I would like to thank Dr. Albert Parra Pozo for being a great colleague and a teammate in solving many lab management issues. We have spent many hours in the lab debugging and fixing computer hardware issues, linux-related problems and learnt a lot in the process.

I wish to thank Mr. Soonam Lee, Mr. David Ho and Mr. Chichen Fu for making a great team for the microscopy imaging project. Chichen's efforts in developing the "Jelly Filling" plugin and its various versions are much appreciated. I would like to thank Ms. Di Chen and Mr. He Li for their help in the video coding project. I wish to thank my former VIPER colleagues Dr. Satyam Srivastava, Dr. Aravind Mikkilineni, Dr. Kevin Lorenz, Dr. Ye He, Dr. Chang (Joy) Xu, Dr. Nitin Khanna, Dr. Marc

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

AC	Arithmetic Coding
ARC	Adaptive Redundancy Control
AVC	Advanced Video Coding
BMA	Boundary Matching Algorithm
CABAC	Context-Adaptive Binary Arithmetic Coding
CBR	Constant Bit Rate
CQ	Constrained Quality
CTU	Coding Tree Unit
CU	Coding Unit
DASH	Dynamic Adaptive Streaming over HTTP
DCT	Discrete Cosine Transform
DIC	Differential Interference Contrast
DP	Data Partitioning
DPB	Decoded Picture Buffer
DST	Discrete Sine Transform
ECMDSQ	Energy-Constrained Multiple Description Scalar Quantizer
ER	Endoplasmic Reticulum
FMO	Flexible Macroblock Ordering
fps	frames per second
GFP	Green Fluorescent Protein
HD	High Definition
HEVC	High Efficiency Video Coding
HTTP	Hypertext Transfer Protocol
IDR	Instantaneous Decoding Refresh

IR	Infra-Red
JCT-VC	Joint Collaborative Team on Video Coding
JVT	Joint Video Team
LOT	Lapped Orthogonal Transform
MB	Macroblock
MCP	Motion Compensated Prediction
MCTF	Motion Compensated Temporal Filtering
MD-BRDS	Multiple Description-Balanced Rate Distortion Splitting
MDC	Multiple Description Coding
MDTC	Multiple Description Transform Coder
MDSQ	Multiple Description Scalar Quantizer
MDSVC	Multipe Description Scalable Video Coding
ME	Motion Estimation
MP	Matching Pursuits
MPEG	Moving Picture Experts Group
MPP	Marked Point Process
MRF	Markov Random Field
MSE	Mean Squared Error
MV	Motion Vector
NAL	Network Abstraction Layer
PALM	Photo-Activated Localization Microscopy
PCT	Pairwise Correlating Transform
PPS	Picture Parameter Set
PSF	Point Spread Function
PSNR	Peak Signal-To-Noise Ratio
PU	Prediction Unit
QoE	Quality of Experience
QP	Quantization Parameter
RJMCMC	Reversible Jump Markov Chain Monte Carlo

RTP	Real Time Protocol
SAO	Sample Adaptive Offset
SAD	Sum of Absolute Differences
SBF	Sliding Band Filter
SDC	Single Description Coding
SI	Switching I
SNR	Signal-To-Noise Ratio
SP	Switching P
STACS	Stochastic Active Contour Scheme
STED	Stimulated Emission Depletion
STORM	Stochastic Optical Reconstruction Microscopy
SVC	Scalable Video Coding
TU	Transform Unit
UDP	User Datagram Protocol
UV	Ultra-Violet
VBR	Variable Bit Rate
VERL	Video Event Representation Language
WHT	Walsh Hadamard Transform

ABSTRACT

Gadgil, Neeraj Jayant. Ph.D., Purdue University, August, 2016. Error Resilient Video Coding Using Bitstream Syntax And Iterative Microscopy Image Segmentation. Major Professor: Edward J. Delp.

There has been a dramatic increase in the amount of video traffic over the Internet in past several years. For applications like real-time video streaming and video conferencing, retransmission of lost packets is often not permitted. Popular video coding standards such as H.26x and VPx make use of spatial-temporal correlations for compression, typically making compressed bitstreams vulnerable to errors. We propose several adaptive spatial-temporal error concealment approaches for subsampling-based multiple description video coding. These adaptive methods are based on motion and mode information extracted from the H.26x video bitstreams. We also present an error resilience method using data duplication in VPx video bitstreams.

A recent challenge in image processing is the analysis of biomedical images acquired using optical microscopy. Due to the size and complexity of the images, automated segmentation methods are required to obtain quantitative, objective and reproducible measurements of biological entities. In this thesis, we present two techniques for microscopy image analysis. Our first method, "Jelly Filling" is intended to provide 3D segmentation of biological images that contain incompleteness in dye labeling. Intuitively, this method is based on filling disjoint regions of an image with jelly-like fluids to iteratively refine segments that represent separable biological entities. Our second method selectively uses a shape-based function optimization approach and a 2D marked point process simulation, to quantify nuclei by their locations and sizes. Experimental results exhibit that our proposed methods are effective in addressing the aforementioned challenges.

1. INTRODUCTION

1.1 Video Coding And Transmission

In recent years there has been a dramatic increase in the volume of video traffic over the Internet. With the development of digital communication standards such as 3G/4G/LTE and WiFi networks, the demand for video delivery is projected to increase even more. According to a recent network usage study by Cisco Inc., 64% of global Internet traffic was used for video delivery in the year 2014 and this usage is predicted to increase to 80% by the year 2019 [1]. In 2014, wired devices accounted for the majority of the IP traffic (54%), soon to be overtaken by the traffic from WiFi and mobile devices (66%) in 2019 [1]. This rapid increase in traffic over wireless channels and among heterogeneous clients has presented significant challenges for developing video coding techniques for wireless transmission.

As shown in Figure 1.1, a common video transmission system consists of a source encoder, a channel encoder, the transmission channel, a channel decoder and a source decoder.

Original Video Signal → Source Encoder → Channel Encoder → Noisy Channels → Channel Decoder → Source Decoder → Reconstructed Video Signal

Fig. 1.1.: A common video transmission system

One of the defining characteristics of a typical wireless channel is the variation of the channel strength over time and frequency [2]. This can cause packet loss during video transmission such that the encoded video information in a lost packet is not available at the receiver. In real-time applications such as video chat, live streaming,

retransmission of lost content is unacceptable because of the strict constraints on the deadline of displaying them. As a result, only a subset of total transmitted packets is available at the receiver, which must reconstruct the signal from the available information.

Traditionally, the goal of any source coding is to represent source symbols by the lowest data rate (bits/pixel or bits/second) for a given reconstruction quality [3]. This is achieved by removing statistical redundancies from the source signal [4, 5]. Typical video compression systems use statistical correlations within a video frame to reduce the spatial redundancy and among the neighboring video frames to reduce the temporal redundancy. In addition, orthogonal transformations are used to further decorrelate the signal [6].

To ensure inter-operability between different manufacturers and devices, many video coding standards have been developed over last couple of decades. Many well-established and popular standards such as MPEG-2 [7], H.264 [8] or high efficiency video coding (HEVC or H.265) [9], VP8 [10], VP9 [11] or VP10 [12] are mainly aimed at achieving better compression efficiencies by removing signal redundancies. This makes the encoded video bitstreams vulnerable to errors. All of the above standards only specify the bitstream syntax such that the decoder is always standardized, whereas the encoder can be designed as per application-specific needs. Yet, mainly due to the high complexity of the coding tools, most encoders are designed according to the design guidelines specified in a standard specification draft which is usually accompanied by its reference software. Typically, during the encoding process, error-free frames are used as a reference to obtain the prediction signal for the current frame. However, if the reference frame becomes corrupt due to transmission losses, error may propagate until the next instantaneous decoding refresh (IDR) (for H.26x) or key frame (for VPx). This phenomenon is known as the encoder-decoder *Mismatch* [13].

When a packet is lost during the transmission, the decoder needs to reconstruct the signal from the available information and by concealing the lost video content. This

is known as video error concealment [14]. To further improve the end-to-end video delivery performance, the encoded bitstream is often made *error-resilient* by designing video coders such that the loss of a part of it can still be faithfully recovered from the received bitstream contents. Therefore, error concealment and resilience methods are indispensable especially for video delivery over unreliable channels such as wireless networks [15].

We first present an overview of video coding standards.

1.1.1 Overview Of Video Coding Standards

Recent video coding standards are designed to address the increasing diversity of services, the growing popularity of high definition (HD) video and the emergence of beyond HD formats such as $4k \times 2k$ and $8k \times 4k$ pixels in spatial resolution and 60-100 frames per second (fps) in temporal resolution [9, 11]. Achieving better compression efficiency while encoding digital video in high quality and with a reasonable computational complexity is generally the main idea behind the development of a new standard. In a typical video coding standard, only the bitstream structure and syntax is standardized. The decoding process after the semantic interpretation of various syntax elements is specified so that any decoder conforming to that particular standard, needs to produce the same output video for a given encoded bitstream. This allows freedom to design various video encoders suitable for specific applications [9].

There are mainly two groups of currently popular video coding standards. The first group consists of standards jointly developed by the ITU-T and the ISO/IEC organizations. H.264 [8] and HEVC [9] are the two latest and popular standards from this group. The other group of video coding standards is released by Google Inc. as an open-source *libvpx* software repository of the WebM [16] project. VP8 [10], VP9 [11] and the latest VP10 [12] belong to this group.

Figure 1.2 depicts the block diagram of a typical video encoder that uses block-based coding approach. The dashed green lines represent the original video data
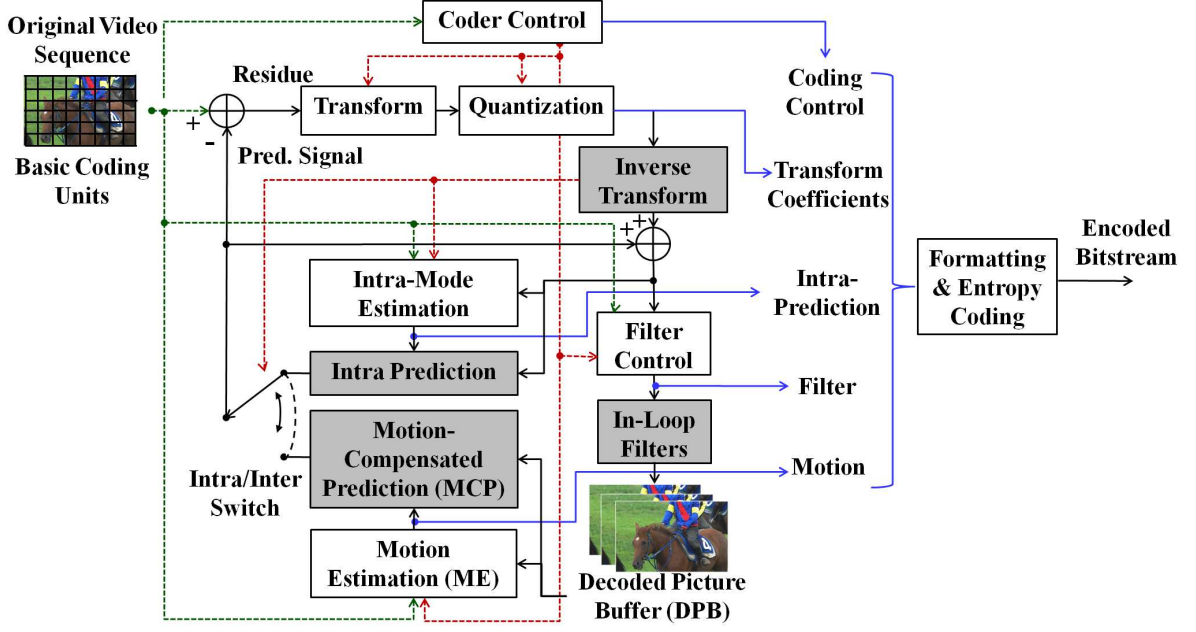
Fig. 1.2.: A typical video encoder

sent to various blocks of the encoder, the dashed red lines represent the signal from the central coder control and the blue lines denote the encoded information. Other exchanges between various blocks are shown using black lines. The encoder contains an *in-loop* decoder (blocks colored in gray) that mimics the operations of a decoder so that the prediction signal used to estimate a current frame is taken from the decoded (not the original) part of the video sequence.

A frame of the original video sequence is first divided into basic coding units or pixel-blocks. Each pixel-block is predicted using already encoded one or multiple pixel-blocks from either the same frame ("Intra") or different frames ("Inter"). Intra prediction uses a directional mode for estimating pixel values of the current block. Intra mode estimation selects for a pixel-block, a directional mode of the various options made available by a specific standard. Inter prediction employs motion compensated prediction (MCP) in which pixel values of the current pixel-block are estimated from that of the coded pixel-block(s) from other frame(s), typically by shifting them in the X and the Y direction. For an Inter-coded block, the amount of shift of the other

frame needed to form the prediction signal of a current pixel-block, is known as the motion vector (MV) and it is generally specified in sub-pixel units. The process of selecting an MV is known as motion search or motion estimation (ME). ME is generally computationally intensive and an encoder can employ an estimation technique suitable to a particular application, considering the trade-off between the accuracy of inter prediction and computational complexity. Thus, the prediction signal (Intra and/or Inter) is used to estimate the original signal of the current pixel-block.

The prediction error/residue then undergoes an orthogonal transformation such as discrete cosine transform (DCT), discrete sine transform (DST) or Walsh-Hadamard transform (WHT). This reduces the statistical correlation within the residue and also allows its spatial frequency-based analysis useful during quantization. The lossy compression occurs next, in the process of quantization of the transformed residue signal. A parameter that specifies the width of the quantizer bin is known as the quantization parameter (QP) and is often used to specify the level of compression for the input video. Many encoders employ rate control techniques with constant bit rate (CBR) and variable bit rate (VBR) as a part of the coder control. A few advanced encoders also have quality control mechanisms such as a constant quality (CQ) mode. In order to form the prediction signal as it would be at the decoder, the *in-loop* decoder uses inverse transformation of the quantized signal. This signal is then added to the previously predicted signal to form the pre-filtered signal. A *deblocking* filter is then applied to reduce the artifacts of the block-based signal processing. Some standards also specify syntax for additional filters e.g. the sample adaptive offset (SAO) filter used in HEVC. The output of this filter is the decoded video frame that is stored in the decoded picture buffer (DPB) that is used to obtain prediction signals for encoding the subsequent frames.

Thus, each pixel-block of the input video sequence is expressed in terms of mode/motion and filter data, transform coefficients and control signals. Control signals contain a few sequence-specific and picture-specific parameters such as frame type, timestamp, type of filters used etc. Above described data is then concatenated and entropy-coded

using arithmetic coding (AC). The encoded bitstream consists of entropy-coded symbols representing different syntax elements that are compliant with a specific standard decoder. Each functional block of Figure 1.2 has been studied with a great interest and many new techniques and inventions have been documented.

A standard has its own specification for the bitstream syntax and the decoding procedures. The details of these for H.264, HEVC and VP8 are specified in [10,17,18]. The reference softwares for these standards: JM (H.264/AVC) [19], HM (HEVC) [20] and VPx (*libvpx*) [16] are also available to provide guidelines for researchers and video systems engineers. The transport mechanisms such as real time protocol and user datagram protocol (RTP/UDP) [21], H.320 [22], MPEG-2 transport stream (MPEG-TS) [23] and dynamic adaptive streaming over hypertext transfer protocol (DASH) [24] are out of the scope of the video coding standards.

1.1.2 Video Error Resilience And Concealment

Many error concealment methods have been proposed recently. This consists of spatial pixel interpolation, frequency domain reconstruction and motion-compensated temporal concealment. A boundary matching algorithm (BMA) is used to recover the lost motion vectors to avoid error propagation [25]. It has been used as a reference concealment method in many studies. A two-stage error concealment that makes use of available motion vectors, an image continuity preserving method and a MAP estimation-based refinement is presented in [26]. A method based on a two-step spatial-temporal extrapolation is described in [27]. Due to high computational complexity of block-matching, BMA and other methods, efforts are made to develop a practical yet effective methods [28]. A comprehensive overview of various error concealment methods is described in [15, 29].

To improve the end-to-end video delivery performance, the encoded bitstream is often made more error-resilient by designing video coders that allow retaining some redundancy in the encoded bitstreams. This redundancy is for the use by the receiver

when some part of the bitstream is lost during transmission. H.26x and VPx standards offer tools for error resilience in their encoder profiles [11, 30].

Flexible macroblock ordering (FMO) is a macroblock (MB) ordering syntax provided by the H.264 standard for an increased error resilience to the bitstream. Generally, a coded slice [8] contains a number of MBs following the raster scan order. When FMO is enabled, the encoder can define a specific allocation of MBs using MB-to-slice mapping (MBAmap) that has a variety of mapping options such as checkerboard, interleave and also custom mapping [19]. Another H.264 tool is the switching-P (SP) and switching-I (SI) slices. They are two special slices that enables efficient switching between video streams and random access for decoders [29, 31].

Since some syntax elements of the bitstream such as motion vectors are more important than others, H.264 allows data partitioning (DP) i.e. partitioning a slice into up to three different partitions for unequal error protection. Each partition can be encapsulated into a separate network abstraction layer (NAL) packet [8, 29]. "DP-A" contains the header information such as MB types, quantization parameters and motion vectors, which are the most important part of data in a slice. "DP-B" contains intra coded block patterns (CBPs) and transform coefficients of I blocks, which are the second most important part of data in a slice. "DP-C" contains inter CBPs and transform coefficients of P blocks which are the least important part. A detailed overview of DP used in the H.264 standard and more advanced DP schemes can be found in [30, 32].

As described in [10–12], VPx offer a few resilience tools for communication of conversational video with low latency over an unreliable network. When arbitrary frames are lost, it becomes necessary to support a coding mode where decoding can still continue regardless of inconsistencies in the received bitstream. A key assumption is that the drift between the encoder and the decoder is still manageable until a key frame is received. It is important that the arithmetic encoder must be able to decode symbols correctly in frames subsequent to the lost one, in spite of corrupt frame buffers leading to a mismatch. In the current VP9 implementation [16], a

flag "error_resilient_mode" is used to achieve error resilience while encoding a source sequence. This mode restricts encoder in the following ways. First is that the entropy coding context probabilities are reset to defaults at the beginning of each frame. Another restriction is on the MV reference selection, where the colocated MV from previously encoded reference frame cannot be included in the candidate list and sorting of the initial list of MV reference candidates based on search in the reference frame buffer is disabled. However, this cannot prevent drift between the encoder and decoder and a key frame is required for resetting the buffers. It also causes a drop in compression efficiency (in the order of 4-5%) and is not recommended when there is no packet loss [11].

Scalable video coding (SVC) [33] has been developed to address the demand of services to heterogeneous clients with various network conditions and device capabilities. In 2007, the JVT approved the SVC [33] extension of the H.264 standard [8]. Multiple description coding (MDC) [34, 35] is another popular error resilient coding technique.

In our work, we investigate MDC in detail.

1.1.3   Overview Of Multiple Description Coding

In MDC, a single signal source is partitioned into several equally important descriptions so that each description can be decoded independently at an acceptable decoding quality. The decoding quality is improved when more descriptions are received. The encoded descriptions are sent through the same or different channels. When packet loss occurs, the bitstream is still decodable and any subset of the descriptions can reconstruct the original signal with a reduced quality. Thus, when even one of the descriptions is received, the decoder is still able to decode the bitstream to provide an acceptable quality without retransmission. Figure 1.3 shows a basic MDC architecture taken from [36].
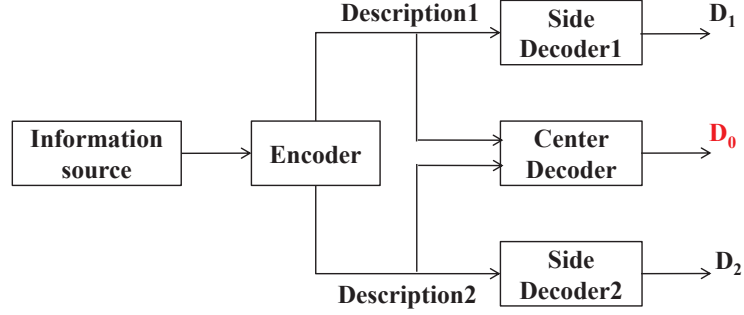
Fig. 1.3.: A basic MDC architecture.

The problem of multiple descriptions was first described at the September 1979 *IEEE Information Theory Workshop* by Gersho, Witsenhausen, Wolf, Wyner, Ziv and Ozarow [34,37]. Then, with a series of following articles published in *The Bell System Technical Journal*, the first theoretical framework was established. Witsenhausen, in February 1980, proposed the use of two independent channels and analyzed channel breakdown for reliable delivery [38]. Wolf, Wyner and Ziv presented a systematic treatment of the problem of multiple descriptions using Shannon's rate-distortion analysis [4] and provided theoretical lower bounds on the distortion [36]. Witsenhausen and Wyner improved the bounds in [39]. Ozarow extended the discussion to Gaussian memoryless sources and presented a complete solution [40]. Jayant applied the concept of multiple descriptions to a speech system with the sample-interpolation procedure that consisted of partitioning of odd-even speech samples [41,42]. In 1982, El Gamal and Cover provided a detailed analysis for achievable rates for multiple descriptions [43,44]. Berger and Zhang analyzed it further in [37,45]. The problem was also studied by Ahlswede in 1985 [46,47].

The pioneering MD video coder was presented in [48] by Vaishampayan et al. with the design of a multiple description scalar quantizer (MDSQ). In this approach, two substreams are created by using two indexes, corresponding to two different quantization levels. To design the quantizer, a method that uses two quantizers whose decision regions are shifted by half of the quantizer interval with respect to each other was

developed by Wang et al. [15]. To improve the coding efficiency, the MDSQ approach was extended to entropy-constrained multiple description scalar quantizers (ECMD-SQs), which uses variable length codes for the index streams [49]. The original MDSQ was developed for memoryless sources, with an asymptotic analysis been presented in [50]. For systems with memory, such as communication over a Rayleigh fading channel or a lossy packet network, a multiple description transform coder (MDTC) was proposed by Batllo et al. [51, 52].

Pairwise correlating transform (PCT) approach introduces dependencies between two descriptions [53–55]. Here, the main concept is to divide the transform coefficients into several groups so that the coefficients between different groups are correlated while the coefficients within the same group are uncorrelated. In [56], the design of lapped orthogonal transform (LOT) bases is proposed. In this method, a transform is selected based on the channel characteristics, the desired reconstruction performance, and the desired coding efficiency. In [57], a LOT-DCT basis approach is proposed to maximize the coding efficiency.

The design of an MD video coder has two main challenges: mismatch control and redundancy allocation. Addressing these challenges a general framework using MCP, a prominent feature of most established video coding standards, has been developed by Reibman in [58]. A detailed treatment to this approach is presented in [58, 59]. It also describes three specific methods using different prediction paths in this general video coder. They demonstrate that mismatch control can be advantageous when there is packet loss as against the complete loss of description [58]. When subjected to packet loss, this MD video coder performs significantly better than the traditional SD coders [59].

There have been efforts to classify MDC approaches to summarize the development. In [60], three classes have been defined based on the predictor type. Class A aims at achieving mismatch control [61–64]. Class B aims at achieving prediction efficiency [65, 66]. Class C represents a trade off between Class A and Class B [67]. Another classification of MDC methods is proposed in [68] to study various MDC

approaches based on the partitioning stage. The coders developed above incorporate spatial and temporal prediction mechanisms into the MD framework. Multiple candidate predictors make this system practically very complicated. The central decoder can use information from both streams to form the best predictor, but at a side decoder, information from the other channel is unavailable and this may cause mismatch. An estimation-theoretic approach to prediction and reconstruction has been presented in [69]. This approach is advantageous because it takes into account all the information available at each decoder for an optimal estimate, and mitigates the degradation due to quantization in the prediction feedback loop. The MDC method described in [58] yields a poor prediction when both descriptions are available. Therefore, an MDC scheme based on a matching pursuits (MP) video coding framework is presented in [70]. In [65], a MD video coder is proposed based on rate-distortion splitting in which the output of a standard video coder is split into two correlated streams. The problem of formation of unbalanced descriptions due to alternation of nonzero DCT coefficients is addressed using MD-balanced rate distortion splitting (MD-BRDS) [71].

Some MDC methods are developed to address source and channel coding jointly for error robustness. One such FEC-based MDC method is proposed in [72] in which the maximum distance separable (n, k) erasure channel codes are used to generate multiple substreams using a joint source-channel coding method. An example of the spatial subsampling-based methods can be found in [73]. One approach to implement the preprocessing data partitioning in the frequency domain is to add zeros to the transform coefficients in both dimensions after DCT, followed by MD generation using polyphase downsampling [74]. Another approach is to split the wavelet coefficients into maximally separated sets [75] and use simple error concealment methods to produce estimates of lost signal samples.

A multiple description scalable video coding (MDSVC) scheme based on motion-compensated temporal filtering (MCTF) [76] is described in [77] in which the advantages of SVC and MDC are combined. In [78], a method based on fully scalable

wavelet video coding is described where post encoding is done to adapt the number of descriptions, the redundancy level and the target data rate.

Many recent approaches take into account adaptive redundancy control (ARC) to optimize the performance of MDC with time-varying channels. In [79], a redundancy allocation for a three-loop slice group MDC is presented. Another three-loop framework is proposed in [80] that uses the slice group-coding tool proposed in H.264 [8]. In [81] an ARC scheme for a prediction-compensated polyphase MDC is described. In [82] presents another ARC scheme for MDC at MB-level. An MDC method based on the concept of redundant slice from H.264 [8] is presented in [83].

Many methods mentioned above are incompatible with the ITU-T H.26X [8, 9] video standards. The use of MDC at pre- and/or post-processing stages allows producing standard-compliant bitstreams [84]. The general idea is to split the video source into two subsequences, which are encoded independently. At the decoder side, when the two descriptions are received, the decoded subsequences are post-processed to recover the full quality video. When only one description is received, the received description is used to reconstruct the video at a coarser quality using error concealment. In [85], an oversampling method is proposed to add redundancy to an image. Then, with a partitioning scheme of the oversampled image, multiple sub-images of equal pixel dimensions are created. This has been extended to video applications in [74], which uses zero padding in the two-dimensional DCT domain. In [86], a method to generate an arbitrary number of descriptions based on zero padding in the DCT domain is described. A simple way to generate two descriptions is to use horizontal or vertical downsampling. To generate more than two descriptions, the partition is done in such a way that redundancy is uniformly distributed along the image columns and rows [62, 64, 87]. Other subsampling-based MDC in the spatial, temporal or frequency domain can be found in [61, 64, 75, 77, 78, 88–92].

Recently, there has been an increasing interest in MDC methods with more than two descriptions for applications in scalable, multicast and P2P environments. Zhu and Liu [68] propose a multi-description video coding based on hierarchical B-pictures

where the temporal level based key pictures are selected in a staggered way among different descriptions. In [93], an MDC architecture with polyphase permuting and splitting of residual blocks is presented. It uses joint temporal and spatial adaptive concealment method based on pixel gradients. A flexible redundancy allocation framework, based on an end-to-end distortion model for three loop, two description MDC is presented in [94]. Hsiao and Tsai [95] present a four-description MDC which takes advantage of residual-pixel correlation in the spatial domain and coefficient correlation in the frequency domain. Several adaptive spatial-temporal concealment methods for subsampling-based MDC architectures are presented in [91, 92, 96–98].

MDC is an active area for inventions and many new implementations have been documented. Some key proposals are [99–113]. A more comprehensive overview of MDC is presented in [34, 35].

Next, we discuss an interesting topic in image processing: analysis of biomedical images. We first review the basic concepts of microscopy imaging with its various modalities.

1.2 Microscopy Imaging

Optical microscopy is considered as an important tool for biomedical research [114, 115]. In recent years, fluorescent microscopy, a form of optical microscopy, has permeated all of cell and molecular biology. It is in a state of rapid evolution, with new techniques, probes and equipment appearing almost daily [116]. Imaging live biological samples (*in-vivo*) has provided key insights towards the functional studies which help characterize various physiological processes [117]. This form of imaging, also known as intravital microscopy, is able to image dynamic processes such as intracellular transport, cell migration and motility, and other cellular interactions and metabolic activities, all of which are important for getting advanced knowledge about clinical diagnosis and treatments [118]. Intravital microscopy has a long history of understanding functional behavior of visceral organs such as the liver [119, 120] and

the kidney [121–123]. Intravital multiphoton microscopy, an advanced optical imaging modality, has been recently used to obtain unique insights into the *in-vivo* cell biology of the brain [124, 125], the immune system [126–128] and cancer tissues [129–131].

Acknowledging the impact of advances in this area, the 2008 Nobel prize in Chemistry was awarded "for the discovery and development of the green fluorescent protein, GFP" that enables scientists to track, amongst other things, how cancer tumors form new blood vessels, how Alzheimer's disease kills brain neurons and how HIV infected cells produce new viruses [132]. The 2014 Nobel prize in Chemistry was awarded "for the development of super-resolved fluorescence microscopy" that allowed a resolution far beyond Abbe's famous limit [133].

We first discuss the background and basic principles of optical microscopy.

1.2.1   Optical Microscopy Background

Optical microscopy or light microscopy uses visible light and a system of lenses to project a magnified image of an object onto the retina of the eye or an imaging device [134]. A typical compound microscope consists of two important components the objective lens and the condenser lens. The objective lens that collects light diffracted by the object and forms a magnified real image near the eyepiece. The condenser lens focuses light from the illuminator onto a small area of the object [134]. During microscope design, it is important to use *Kohler illumination* that gives bright, uniform illumination of the object and simultaneously positions the sets of image and diffraction planes at their proper locations [134, 135]. In an advanced microscope, both these components consists of smaller sub-components and they perform close to their theoretical limits. The early development of the theory of diffraction and image formation is attributed to Abbe who also set the famous diffraction limit (known as *Abbe diffraction limit*) that specifies the maximum spatial resolution that can be achieved while imaging an object [135].

The human visual system requires contrast to perceive details of objects [136]. The simplest and very effective contrasting method is dark-field [137] that uses the scattering of light on small particles that differ from their environment in refractive index, the phenomenon known as *Tyndall effect* [136]. Based on the illumination technique, a few major techniques are phase contrast [138] proposed by Zernike, differential interference contrast (DIC) [139] developed by Nomarski, reflection [140] and modulation contrast [141] microscopy. The most popular contrasting technique is Fluorescence microscopy [116, 136]. It is an advanced imaging modality based on the principle of fluorescence exhibited by *fluorophores* which absorb light in a specific wavelength range and emit it with lower energy shifting the emitted light to a longer wavelength.

More recent super-resolution microscopy techniques such as stimulated emission depletion (STED) microscopy [142], photo-activated localization microscopy (PALM) [143] and stochastic optical reconstruction microscopy (STORM) [144] have successfully broken the diffraction limit to provide higher resolution. A detailed discussion of imaging modalities is presented in [134–136].

Below we review fluorescence microscopy in more detail.

1.2.2 Fluorescence Microscopy

The phenomenon of fluorescence was first documented by Herschel as *dispersive reflection* in 1825 when he observed blue light emitted from the surface of a solution of quinine [135]. Later in 1852, Stokes coined the term *fluorescence* when he studied the distance between the maximum of excitation and emission wavelength. This distance is known as *Stokes shift* [116, 135].

The outermost electron orbitals in a *fluorophore* determine the wavelengths of absorption and emission, and also its efficiency as a fluorescent compound [116]. When a *ground state fluorophore* absorbs energy from photons, the electronic, vibrational and rotational states of the molecule can alter. The absorbed energy makes an elec-

tron jump into a different orbital farther from the nucleus. This state is known as the *excited state*. The transition from *ground*-to-*excited state* typically takes a few femtoseconds. The absorbed energy is then released in the form of photon emission and via vibrational relaxation, the molecule returns to its ground state [116]. The details of excitation and emission process were studied using a form of diagram proposed by Jablonski in the 1930s [145]. This diagram is known as *Jablonski diagram* that is a schematic drawing based on electronic states corresponding to various energy levels.
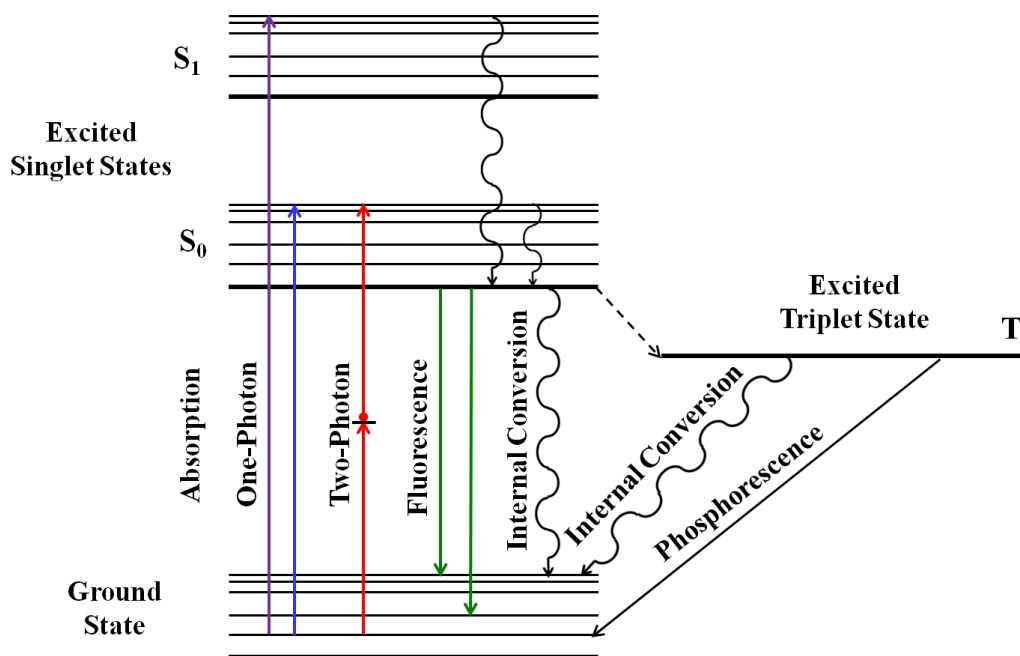
Fig. 1.4.: *Jablonski diagram* for single and two photon excitation

Figure 1.4 represents the *Jablonski diagram* showing attainable energy levels in a fluorescent molecule. The molecule can absorb one photon of the ultra-violet (UV)-blue wavelength or two photons of the red wavelength of the visible spectrum. The absorption of a UV photon can cause electron excitation to a higher energy *singlet state* (S_1), whereas a blue photon can cause excitation to S_0, a lower energy *singlet state* [116, 134]. Also, two red photons can cause electron excitation to S_0 from the *ground state* [146, 147]. The collapse to the *ground state* from the either *singlet state* can occur through one of the following three ways:

- *Fluorescence emission:* The molecule emits a photon. The process of absorption and emission occurs almost simultaneously (in the interval of $10^{-9} - 10^{-12}$ seconds).

- *Internal conversion:* The molecule releases vibrational energy in the form of heat without photon emission to enter either a lower energy *singlet state* or the *ground state*.

- *Triplet state:* The excited electron enters the *triplet state* that can make the molecule chemically active often leading to *photobleaching*, the permanent loss of fluorescence. The electron in the *triplet state* can return to the *ground state* through internal conversion or by the phenomenon of *phosphorescence*. Unlike *fluorescence*, *phosphorescence* the emission is not instantaneous and typically lasts fraction of a second to minutes.

As shown in Figure 1.4, the energy of the emitted photon is less than that of the absorbed photon. The energy (E) of a photon is inversely proportional to its wavelength(λ): $E = hc/\lambda$, where h is the *Planck's constant* and c is the speed of light. Therefore, for single-photon excitation, the wavelength of the emitted photon is longer than that of the absorbed photon. In case of two-photon (or multi-photon) excitation, the wavelength of the emitted photon is typically shorter than that of the absorbed photons.

Fluorophores that are used for staining biological specimen are known as *fluorescent dyes* (dyes hereafter). The absorption and emission spectra of a dye are distinct and considered as an important property while selecting the dyes to stain a biological specimen. When two or more molecules in the same specimen are required to be labeled with dyes, it is essential to study their absorption and emission spectra [134]. In practice, *Stokes shift* is obtained as the difference between the emission and absorption maxima. For two-photon and multi-photon excitation, this difference is known as *anti-Stokes shift* [147, 148], since the emission wavelength is shorter than the absorption wavelength. Depending on the properties of a *fluorophore*, this shift

can range from a few to several hundred nanometers [134]. It is important to note that for a particular dye, the two-photon absorption spectrum scaled to half the wavelength is typically not equivalent to its single-photon absorption spectrum [147]. Dyes exhibiting a large *Stokes shift* or *anti-Stokes shift* are advantageous for fluorescence microscopy because the bands of absorption and emission are easier to isolate using *interference filters.* The probability that a dye molecule absorbs a photon is known as its *molar extinction coefficient* (ϵ) which is typically expressed in per mole per centimeter ($M^{-1}cm^{-1}$) [116]. Another important property of a dye is its *quantum efficiency* of fluorescence emission. *Quantum efficiency* is the ratio of the number of emitted photons causing fluorescence, to the number of absorbed photons. Other main characteristics of dyes are resistance to *photobleaching*, solubility and chemical stability [134]. Newly developed dyes such as *Alexa* and *Cyanine* dyes are popular because of their high *quantum efficiency* and high resistance to *photobleaching. Green fluorescent protein (GFP)* obtained from the jellyfish *Aequorea victoria* with its mutated forms blue, cyan and yellow fluorescent proteins and a recently developed group *red fluorescent proteins (DsRed)* are some other examples of commonly used dyes. A detailed discussion of fluorescent dyes and their applications is provided in [149].

When a biological specimen is stained with a fluorescent dye, it can be observed using an optical microscope. Figure 1.5 shows optical schematic diagrams of three types of microscopes: conventional (widefield), confocal and two-photon respectively [135, 147, 150]. A widefield microscope (Figure 1.5 (a)) is a conventional imaging device that can illuminate the specimen using *Kohler illumination* and magnify the optical signal emitted by a relatively wide region of the specimen. This has an obvious disadvantage of collecting a considerable amount of out-of-focus signal. To solve this, Minsky, in the 1950s, proposed the design of a "confocal" microscope [151]: Figure 1.5 (b). Unlike viewing the whole sample in case of widefield microscopy, the specimen is scanned point-by-point by making the excitation light and the detector "in-focus." The use of pinhole is to block all out-of-focus signal resulting in increased signal contrast [135]. Confocal microscopes can provide 3D resolution using a set

(a) Widefield

(b) Confocal
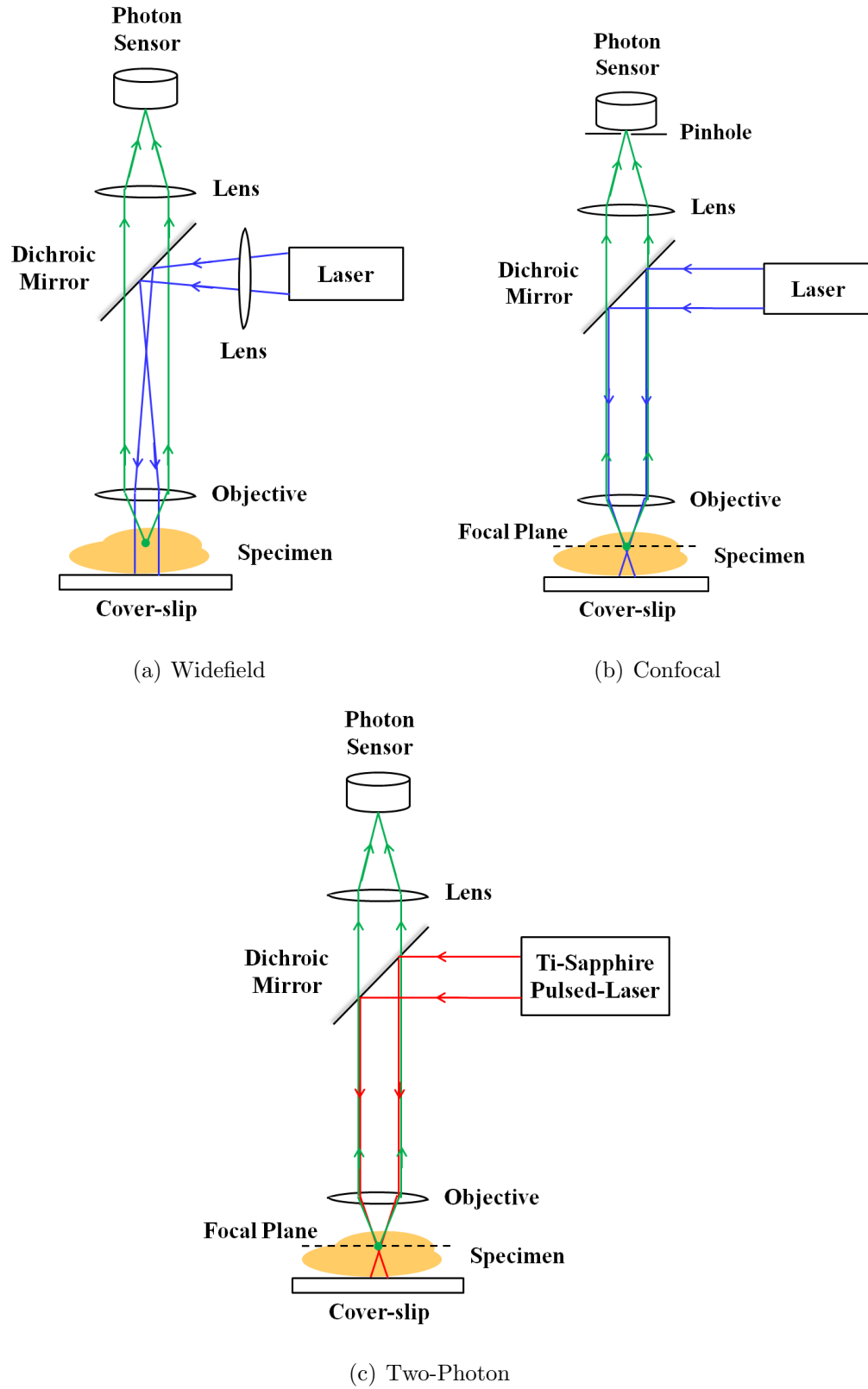
(c) Two-Photon

Fig. 1.5.: Fluorescence microscope schematics

of conjugate apertures functioning as spatial filters [147]. As shown in Figure 1.5 (c), two-photon excitation microscope developed in the 1990s by Denk et al. [152] uses a high-power pulsed laser to illuminate the specimen with longer (than confocal microscope) wavelength photons. The excitation is restricted to a small focal volume, eliminating the need to use pinhole [150]. Although confocal and two-photon microscopy techniques are similar, two-photon has a number of advantages:

- *Photobleaching* and *Photodamage*: In confocal microscopy, the entire specimen is typically subjected to high-energy UV photons. While imaging a specimen *in-vivo*, this photon excitation can cause rapid *photobleaching* of the *fluorophore* and *photodamage* (damage to the specimen) outside the focal region [153,154]. In two-photon microscopy, only a part of the specimen at a particular focal plane is excited and the damage is limited to that region.

- Signal loss: Confocal microscopy uses the pinhole that rejects light from most part of specimen that is out-of-focus [147]. But two-photon microscopy does not require pinhole aperture and hence can minimize signal loss. Due to localized excitation of the specimen, all of the fluorescence emanating from it may be collected and mapped to this single focal point in 3D space [154].

- Scattering: The wide separation between the absorption and emission spectra (*anti-Stokes shift*) in two-photon microscopy helps filter out the excitation photons and Raman scattering effect, while collecting photons emitted by the specimen [147].

- Imaging depth: The amount of scattering of light as it passes through a specimen decreases as the wavelength of the light increases [147]. Confocal uses UV (shorter wavelength) light, thus cannot practically penetrate beyond certain depth. Two-photon microscopy uses near-infrared (IR) (longer wavelength) excitation that is suitable for penetrating deeper into a biological tissue. However, it should be noted that the maximum imaging depth also depends on the

properties of the tissue sample, the *quantum efficiency* of the *fluorophore* and the properties of the microscopy optics [154].

Now we discuss the properties of images acquired using fluorescence microscopy and current challenges in analyzing them.

1.2.3 Challenges In Microscopy Image Analysis

Due to the recent advances in fluorescence microscopy as discussed earlier, it is possible to obtain images of thick biological tissues. Two photon microscopy allows collection of hundreds of focal plane images providing the capability to characterize 3D structures at subcellular resolution [114, 115]. The size and complexity of such 3D image data makes manual image analysis and visualization almost impracticable. Automated methods of image segmentation are required to obtain quantitative, objective and reproducible analysis [155]. Fluorescent microscopy when combined with automated digital image analysis such as image segmentation, becomes a powerful tool for biomedical research [154].

Image segmentation is the process of assigning a label to every pixel in an image such the pixels with the same label have similar visual characteristics. Visual characteristics include color, intensity, texture, shape or some other computed value. All pixels in a particular region are similar with respect to one or multiple visual characteristics, while pixels belonging to different regions are significantly different than one another [154, 156]. Biomedical image segmentation typically require labeling different biological entities such as tubules, vasculature, lumen, cell-nuclei in 2D or 3D, often based on the above characteristics, the understanding of the imaging modality and some prior knowledge of the biological structure.

Automatic analysis of microscopy images, however has many challenges. The images are inherently anisotropic, with aberrations and distortions that vary in different axes. For images taken successively in the z-direction by shifting the focal plane deeper in the sample, image contrast decreases with depth. This reduced con-

trast exacerbates a common problem of fluorescence of having characteristically low signal levels consisting of as little as a single photon [157]. Attempts to increase the image contrast during acquisition come at a cost of limiting spatial-temporal resolution and higher chances of *photodamage*, *photobleaching* and signal noise [154]. Moreover, the biological objects to be segmented have poorly defined edges and the object boundaries do not contain continuous edges. Due to the inherent trade-off between the spatial and temporal resolution, images representing dynamic biological processes must be captured at low spatial resolution. This makes microscopy image segmentation and rendering very sensitive to small changes in regularity assumptions and mathematical parameters used in analysis, often leading to inconsistent and unpredictable results [114].

Physical and optical properties of a microscope can result into planar and angular distortions some of which can be modeled using the point-spread function (PSF). The PSF is defined as the 3D convolution of a point source object by the microscope objective. It specifies how that microscope images a point source and how the lens spreads the optical signal along the three axes [154]. A bad PSF results in decreased image contrast, poorer edge details and worse signal-to-noise ratio (SNR) as compared with a near-ideal PSF. The authors in [158] describe ways to reverse the effects of the PSF by deconvolution, whereas others [159] argue that deconvolution is not necessarily helpful in improving the image quality considering the goals of a biological study.

In multiphoton microscopy data from multiple spectral regions is recorded. This acquisition process uses band-pass filters, each of that aggregates a range of wavelength to one spectral channel. This channel isolation is often imperfect causing crosstalk between different spectral channels [115]. An example of the effect of crosstalk is the data from one particular channel appears in the data from other channels with reduced intensity. Crosstalk effect is aggravated when a channel becomes saturated with more number of photon emissions than the acquisition device can capture. In *in-vivo* imaging, motion artifacts are introduced as a result of respi-

ration and heartbeat of the live specimen. Image registration is required to address this problem. This introduces distortions such as translations and warping into the images, degrading image quality [154].

1.2.4 Overview Of Image Segmentation Methods

There have been many recent techniques developed to segment and analyze biological images. Edge detection methods proposed by Canny [160], Harris [161] are often used for segmenting boundaries of biological quantities. Primary image processing methods such as thresholding [162], morphological operations [163, 164], 2D/3D filters with various kernels [156] are used as preprocessing to remove noise, distortions and binarize images before doing segmentation.

Active contours (also known as *snakes*) is a widely used segmentation approach. In principle, an active contour is a curve that evolves within an image from some initial position toward the boundary of the biological object [165, 166]. The initial position of the *snake* is usually specified by the user or is otherwise provided by an auxiliary rough detection algorithm. The evolution of the *snake* is formulated as a minimization problem. The associated cost function is usually referred to as the *snake* energy. Edge-based active contours [165, 167] compute image gradients map to identify objects. *Snakes* has been investigated using region-based approaches, seeking an energy equilibrium between the foreground and the background [168]. The region-based methods can typically produce better segmentation results than the edge-based methods. This is because the region-based methods are relatively independent of initial contour location and more robust against image noise. Yet, these region-based methods fail to segment when images have inhomogeneous intensities [169] Another popular approach is the stochastic active contour scheme (STACS) that uses textures, edge, and region-based information [170]. A topology preserving variant of STACS that combines topology with level set formulation is developed in [171] and is shown to outperform the widely used seeded watershed technique [172]. A vector

field convolution based active contour model is proposed in [173]. In [174], an open active contour model for analyzing actin filament is presented. A 3D active surfaces is proposed in [175] that considers images as a 3D volume using modified energy functions.

Although lots of work have been done in active contour and its variations, it still has many limitations [166]. First and foremost, active contour is very sensitive to the initial curves. If the initial curves are too far from the region of interest, it takes considerable time to capture the region while often failing in segmenting the desired quantities. Manual initialization is time consuming often questioning the very motivation of having an automatic segmentation method. Some approaches for automatic initialization of the contours have been proposed in [176] and references therein. Yet it remains an open research topic without a widely-accepted solution. *Snakes* have issues with robustness when used on images acquired in typical microscopy imaging conditions that unavoidably introduce noise, bias field and low contrast in the images. Another major problem is the lack of multi-object discrimination which can be an essential requirement for many biomedical applications e.g. cell tracking [166].

Watershed technique [177] is another popular segmentation approach based on mathematical morphology. Intuitively, the image is considered as topographic relief such that the height of each point is directly related to its gray level. The rainfall gradually begins to fill the low-terrain forming "lakes" (or "catchment basins"). The watershed lines are defined as the lines that separate the lakes. Generally, the watershed transform is computed on the gradient of the original image, so that the catchment basin boundaries are located at high gradient points. Many variations of this approach have been proposed over the years. A method that uses prior probabilistic information with the watershed transform is presented in [178]. In [179], a two step watershed method is presented in which, three types of cell structures: nuclei, cell walls and cell-cell contacts are segmented in order to distinguish different actin-binding proteins from the images of Epithelial cells. However, watershed used in biological images typically suffers from over-segmentation that results into thousands

of small basins [178]. This is addressed using a marker image [180], that reduces the number of minima in an image. Like active contours, watershed methods are also highly sensitive to image noise. The use of anisotropic filters [181] has been proposed to address this. Yet, watershed typically produces bad segmentation results in low contrast and poor edge-areas that are typical artifacts in microscopy images.

An active mask framework that uses a multiscale and multiresolution approach as well as region-based and voting-based functions is proposed [182]. Another region-based method called discrete region competition is proposed in [183]. Other methods include the sliding band filter (SBF)-based joint segmentation approach presented in [184], that is useful in detecting overall convex shapes. A method to segment vasculature in 3D, that uses noise modeling, planer geometry and adaptive region growing is presented in [185]. A novel approach for coupling image restoration-segmentation [186] has been proved effective in segmenting 3D biological structures e.g. the endoplasmic reticulum (ER) and the Drosophila wing disc. A recent popular edge and ridge-based method that uses steerable filters for feature detection is proposed in [187]. Recently, several methods based on convolutional and deep neural networks are proposed for biomedical image segmentation [188–191]

ImageJ [192] is a popular open-source, Java-based image analysis toolkit that is developed by Rasband and currently maintained by the National Institutes of Health (NIH). Fiji [193] is a distribution of ImageJ that has more specific tools for biomedical image analysis. Another platform called Icy is a collaborative bioimage informatics framework that combines a website for sharing tools and material, and software with high-end visual programming capabilities [194]. There have been several 3D image rendering softwares e.g. Voxx [195].

1.2.5 Our Image Data And Notation

The images presented in this work are of biological tissues mainly belonging to the kidney and the liver, collected *in-vivo* using intravital multiphoton microscopy for

various structural and functional biological studies $[115, 117, 196, 197]$. As shown in
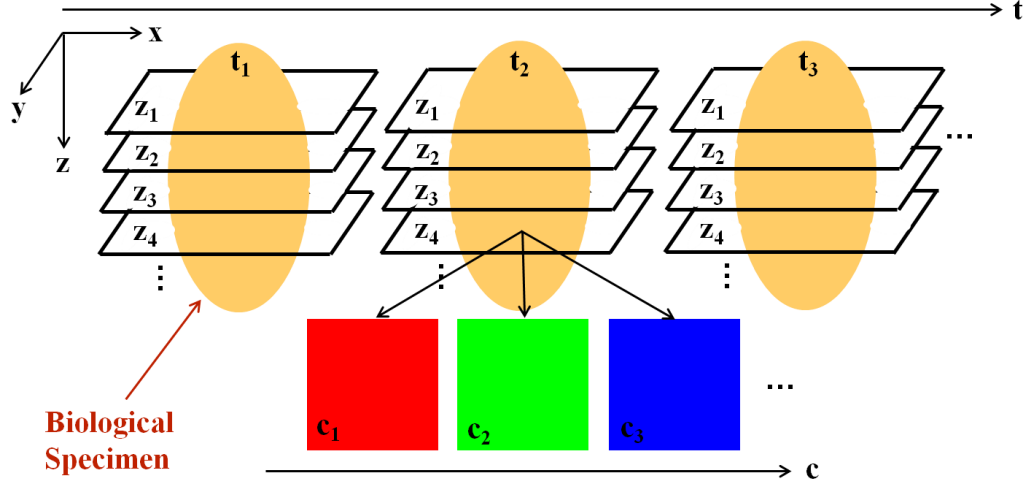
Fig. 1.6.: Our image data notation

Figure 1.6, our data has three spatial dimensions that represent a biological specimen in space, the time dimension which indicates that the data is collected over several time instances and the spectral dimension that represents the fluorescence of multiple dyes injected into the specimen. Let I_{z_p,t_q,c_r} represent a grayscale image of $X \times Y$ pixels collected from the p'th focal slice in the z-direction at the q'th time sample, representing the r'th spectral (or color) channel, where $p \in \{1, 2, ..., P\}$, $q \in \{1, 2, ..., Q\}$ and $r \in \{1, 2, ..., R\}$. P is the number of focal slices, Q is the number of time samples and R is the number of color channels of data collected from one biological specimen.

Note that our data and thus the notation consists of three types of dimensions: space, time and color. We call each of them a *dimension-type*. It is possible for a particular specimen, only one point from a *dimension-type* is collected. In such case our notation will drop that specific *dimension-type* from the original notation that consists of the three *dimension-types*. For example, in many structural studies the data is collected only at one time sample. For such data, we use the notation I_{z_p,c_r}. In some cases only one dye is injected into a specimen producing data only in one color

channel. For that data, the notation becomes I_{z_p,t_q}. We will use the above described notation and conventions throughout the thesis.

Some examples of images used in our work are shown in Figure 1.7. Figure 1.7 (a)

(a)   (b)

(c)   (d)
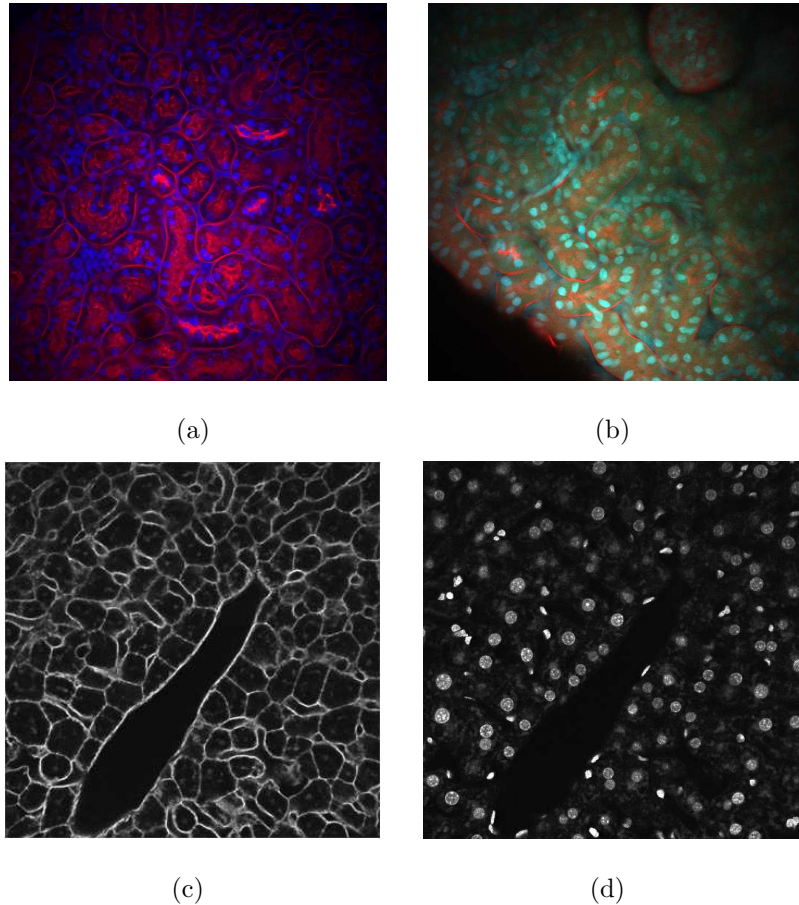
Fig. 1.7.: Examples of microscopy images used in our work

and (b) are from rat kidney samples. In this image data, the red channel represents proximal tubules with their lumen (brush border) and the blue channel represents cell nuclei in the kidney. Figure 1.7 (c) and (d) are separated red and blue channels respectively, of the data collected from a rat liver. The red channel represents vasculature, portal vein and hepatocytes, whereas the blue channel represents the cell nuclei. Our specific image analysis goals and methods will be discussed in Chapter 4 and Chapter 5.

We next discuss video surveillance and crowdsourcing for public safety applications.

1.3 Crowdsourcing For Public Safety

Video surveillance systems are widely deployed for public safety [198, 199]. They are used as an effective tool for crime prevention and an *after the fact* forensic tool. Agencies such as airport security, border control use real-time monitoring systems that have active warning capabilities. Most systems have multiple surveillance cameras. The huge amount of data being collected in real-time makes it practically impossible to do manual analysis and crime detection. Automatic methods are often used to reduce the manual efforts.

Automatic Detection For Surveillance Applications: There are many recent approaches to real-time monitoring and alerting for video surveillance systems [200, 201]. Some main technical areas are object detection, tracking and categorization, human action and behavior analysis [202–207]. However they have many challenges such as dealing with occlusions, shadows, illumination changes and the requirements to track objects across multiple cameras. Detection methods are not perfect and are susceptible to miss detections and false alarms. Automatic object tracking can be a problem due to occlusions [199, 208]. The work on object recognition in the past 30 years has demonstrated that object recognition or categorization even from a single image is a highly complex task [200, 209]. A comprehensive overview of machine recognition of human activities is presented in [205] that discusses several non-parametric, volumetric and parametric modeling methods. Some of the main challenges in automatic detection arise from noisy real-world conditions, the difficulties in finding invariance in human actions and the sparsity of standardized testbeds. The advances in this field lack in robustness to real-world conditions and real-time performance. Whereas, establishing feature correspondence between consecutive video frames is considered as a bottleneck in tracking-based human motion analysis [203].

A probabilistic technique along with other computer vision methods is used to model scene dynamics in [210]. However, it is sensitive to severe occlusions and it can be impacted by illumination changes. Improving the performance of such intelligent systems remains an active research area within the image processing and the computer vision community. Human motion analysis has been studied in great depth over the years. A popular approach to tracking multiple occluding people by localization of multiple planes is described in [211]. Here, a planar homographic occupancy constraint that fuses foreground likelihood information from multiple views is used to resolve occlusions and localize humans. This method faces issues such as missed detections in the case of similar appearance of background to a person and occlusions from the background. Also, two or more occupancy tracks of humans can merge to cause "identity switching." The limitations of automatic systems are also highlighted in the failure cases from [203, 211, 212].

Crowdsourcing: As defined by J. Howe in 2006 [213], "crowdsourcing" also referred to as the collective intelligence, the wisdom of the "crowd" or human computation, is often considered as an effective solution to problems that involve cognitive tasks. According to Surowiecki, the crowd is often holding a nearly complete picture of the world in its collective brains [214]. A crowd can perform the same or sometimes even better than an expert when the crowd typically satisfies the basic conditions: diversity, independence, decentralization and aggregation [214]. There have been efforts to use this collective intelligence to do tasks that machines find very difficult. The Internet provides a perfect platform to reach out to the crowd or contributors and collect their inputs. There has been an increasing interest in providing web-based crowdsourcing platforms such as Amazon's Mechanical Turk (*MTurk*), *Turkit*, *Freelancer*, *CloudCroud*, *uTest*, *Mob4hire*, *Topcoder* and *CrowdFlower* [215]. The members of the crowd work on a variety of tasks such as language translation, copying text from images and writing transcripts from audio messages. Such platforms are useful for many research communities including image/video processing for subjective experiments. In [216], a web-based platform is developed for quality of experience (QoE)

assessment and is integrated with *MTurk*. A subjective video quality evaluation system to assess Internet video quality using crowdsourcing was presented in [217]. According to [218], the proposed subjective QoE assessment method for YouTube videos based on crowdsourcing is highly effective and reliable.

Describing the important contents and actions in a video sequence is known as video annotation [219]. This is an area in which crowdsourcing is very useful [220]. The video event representation language (VERL) is a formal language for representing events for designing an ontology for an application domain and for annotating data with the ontology's categories [221]. An example is sports annotation as described in [222]. Another area of interest is object detection and tracking. Crowdsourcing-based annotation is very useful not only for obtaining the annotations, but for improving the performance of automatic detection methods. In [223], *MTurk* is used to provide annotations to train object detectors where the system automatically refines its models by actively requesting annotations of images from the crowd. In [224], a system in which machine learning and crowdsourcing enhance each other is proposed. In this system, a semi-automatic image annotation approach is presented that uses crowdsourcing to help robots register novel objects with their semantic meaning. A web-based social analysis tool is proposed in [225] in which several key strategies are presented that improve the quality and diversity of worker-generated explanations.

Crowdsourcing For Surveillance Using Annotation Tools: A recent approach to crowdsourcing surveillance videos is described in [226] where *CrowdFlow*: an *MTurk*-based toolkit for integrating machine learning with crowdsourcing is presented. *Crowded* is another such web-based platform developed by the defense science & technology laboratory (DSTL) in which, images of a particular location are collected from a variety of media sources to provide an operator with real-time situational awareness [227]. A similar *MTurk*-based web interface, known as *VATIC* was developed to monetize, high quality crowdsourced video labeling [228]. While these platforms combine ideas of crowdsourcing and video annotations, they are not

designed specifically to help law enforcement authorities with surveillance video analysis and alerting systems.

Developing and deploying crowdsourced annotation tools for law enforcement involve many issues. For example, as described in [229], protecting the video contents and allowing freedom of speech to the annotators while avoiding chaos and protests are some important considerations. Therefore, there are significant issues in using commercial crowdsourcing tools such as *MTurk* for the security applications. There is need to develop a web-based video annotation system for crowdsourcing surveillance videos in a controlled environment. Such system can help the law enforcement authorities recognize potential threats and investigate criminal activities.

1.4   Contributions Of The Thesis

In this thesis, we developed new methods for error resilient video coding, microscopy image segmentation and an implementation of a video annotation platform. The main contributions of the thesis are:

- Adaptive Error Concealment for Multiple Description Video Coding
  We propose two adaptive error concealment methods for a temporal-spatial four description multiple description video coding architecture. Our adaptive methods are motion vector analysis and error estimation using the H.264-coded MDC bitstreams. We propose another adaptive concealment method for a spatial-subsampling based MDC architecture. This method uses motion information and prediction mode extracted from HEVC-coded MDC bitstreams. Experimental results show that our proposed methods are effective under packet loss conditions during video transmission.

- Error Resilient Video Coding using Duplicated Prediction Information for VPx Bitstreams
  We describe an error resilient coding method for VPx-coded bitstreams using

duplication of prediction information. Experiments indicate that our method provides a graceful quality degradation under packet loss conditions.

- Jelly Filling Segmentation of Biological Images Containing Incomplete Labeling
We propose an iterative 3D segmentation method mainly for fluorescence microscopy images containing the incomplete labeling artifact. Intuitively, our method is based on filling the disjoint background regions of an image with "jelly-like" fluid such that the interactions between the "jellys" and the segmented foreground can be used to separate different biological entities in 3D. Experiments with our images exhibit the effectiveness of our proposed method as against some existing methods.

- Nuclei Segmentation of Microscopy Images using Midpoint Analysis and Marked Point Process
We present a cell-nuclei segmentation method based on midpoint analysis and a random process simulation. Midpoint analysis is used to classify the segmented regions into single-/multi-centered objects based on their shape properties. A 2D spatial point process simulation is then used to quantify cell-nuclei by their location and size.

- A Video Annotation Tool for Crowdsourcing Surveillance Videos
We describe our implementation of a web-based video annotation tool built for the use of the law enforcement authorities for rapid analysis of surveillance videos. The tool makes use of crowdsourcing in a controlled manner to distribute annotation tasks to a set of trained "crowds" and aggregates the results for the law enforcement authorities.

1.5 Publications Resulting From Our Work

Book Chapters

- **N. Gadgil**, M. Yang, M. L. Comer, and E. J. Delp, "Multiple description coding," *Academic Press Library in Signal Processing: Image and Video Compression and Multimedia*, S. Theodoridis and R. Chellappa, Eds. Oxford, UK: Elsevier, 2014, vol. 5, no. 8, pp. 251-294.

Journal Articles

- **N. Gadgil**, P. Salama, K. Dunn, and E. J. Delp, "Jelly filling image segmentation of biological structures," *To be submitted to the IEEE Transactions on Medical Imaging.*

- **N. Gadgil** and P. Salama, K. Dunn, and E. J. Delp, Nuclei segmentation of microscopy images using marked point process, *To be submitted to the SPIE Journal of Medical Imaging.*

- M. Yang, **N. Gadgil**, M. L. Comer, and E. J. Delp, "Adaptive error concealment for multiple description video coding," *Signal Processing: Image Communication*, 2016.

- J. Duda, P. Korus, **N. Gadgil**, K. Tahboub, and E. J. Delp, "Image-like 2D barcodes using generalizations of the Kuznetsov-Tsybakov problem," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 4, pp. 691-703, April 2016.

Conference Papers

- C. Fu, **N. Gadgil**, K. Tahboub, P. Salama, K. Dunn and E. J. Delp, "Four dimensional image registration for intravital microscopy," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, July 2016, Las Vegas, NV.

- **N. Gadgil**, P. Salama, K. Dunn and E. J. Delp, "Jelly filling segmentation of fluorescence microscopy images containing incomplete labeling," *Proceedings of the IEEE International Symposium on Biomedical Imaging*, April 2016, Prague, Czech Republic.

- **N. Gadgil**, P. Salama, K. Dunn and E. J. Delp, "Nuclei segmentation of fluorescence microscopy images based on midpoint analysis and marked point process," *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 37-40, March 2016, Santa Fe, NM.

- **N. Gadgil** and E. J. Delp, "VPx error resilient video coding using duplicated prediction information", *Proceedings of the IS&T Electronic Imaging: Conference on Visual Information Processing and Communication VII*, February 2016, San Francisco, CA.

- K. Tahboub, **N. Gadgil**, and E. J. Delp, "Content-based video retrieval on mobile devices: How much content is enough?" *Proceedings of the IEEE International Conference on Image Processing*, pp. 1603-1607, September 2015, Quebec City, Canada.

- **N. Gadgil**, H. Li, and E. J. Delp, "Spatial subsampling-based multiple description video coding with adaptive temporal-spatial error concealment," *Proceedings of the Picture Coding Symposium*, pp. 90-94, May 2015, Cairns, Australia.

- J. Duda, K. Tahboub, **N. Gadgil**, and E. J. Delp, "The use of asymmetric numeral systems as an accurate replacement for Huffman coding," *Proceedings of the Picture Coding Symposium*, pp. 65-69, May 2015, Cairns, Australia.

- K. Tahboub, **N. Gadgil**, J. Ribera, B. Delgado, and E. J. Delp, "An intelligent crowdsourcing system for forensic analysis of surveillance video," *Proceedings of the IS&T/SPIE Electronic Imaging: Video Surveillance and Transportation Imaging Applications*, vol. 9407, pp. 94070I: 1-9, February 2015, San Francisco, CA.

- J. Duda, **N. Gadgil**, K. Tahboub, and E. J. Delp, "Generalizations of the Kuznetsov-Tsybakov problem for generating image-like 2D barcodes," *Proceedings of the IEEE International Conference on Image Processing*, pp. 4221-4225, October 2015, Paris, France.

- **N. Gadgil**, K. Tahboub, D. Kirsh, and E. J. Delp, "A web-based video annotation system for crowdsourcing surveillance videos," *Proceedings of the IS&T/SPIE Electronic Imaging: Imaging and Multimedia Analytics in a Web and Mobile World*, vol. 9027, pp. 90270A: 1-12, February 2014, San Francisco, CA.

- K. Tahboub, **N. Gadgil**, M. L. Comer, and E. J. Delp, "An HEVC compressed domain content-based video signature for copy detection and video retrieval," *Proceedings of the IS&T/SPIE Electronic Imaging: Imaging and Multimedia Analytics in a Web and Mobile World*, vol. 9027, pp. 90270E: 1-13, February 2014, San Francisco, CA.

- **N. Gadgil**, M. L. Comer, and E. J. Delp, "Adaptive error concealment for multiple description video coding using error estimation," *Proceedings of the Picture Coding Symposium*, pp. 97-100, December 2013, San Jose, CA.

- **N. Gadgil**, M. Yang, M. L. Comer, and E. J. Delp, "Adaptive error concealment for multiple description video coding using motion vector analysis," *Proceedings of the IEEE International Conference on Image Processing*, pp. 1637-1640, October 2012, Orlando, FL.

2. ERROR RESILIENT VIDEO CODING USING ADAPTIVE ERROR CONCEALMENT FOR MDC

As described earlier, an MDC encoder generates multiple correlated descriptions so they contain redundancy when more than one descriptions are received. In case of packet loss, parts of one or more descriptions are lost. The decoder can use the redundancy in the descriptions to do error concealment that essentially provides an estimation of the lost video signal. Depending on the nature of redundancy between the descriptions, it is possible to have more than one concealment method available at the decoder so that the decoder can select one of them based on various receiving scenarios of the descriptions and available bitstream parameters.

For a subsampling-based *Class A* MDC [60], the descriptions have spatial and temporal redundancy that can be used at the decoder to apply either spatial or temporal concealment method. In a related previous work [64], this selection of concealment method for a four-description MDC is made only based on the type of received descriptions. The concealment performance is further improved by using an "adaptive" concealment strategy such that the decoder selects a method based on some analysis for each lost unit such as a frame or an MB from the H.264 standard. A frame-level method uses error tracking to adaptively select the concealment strategy for the four-description MDC [91]. Another adaptive method uses foreground-background and distortion mappings to make the selection for each lost MB [92]. A detailed description of these methods is presented in [29, 230]. However, both of these adaptive methods require reliable transmission of the additional or "side" information to the decoder. Sending the side information also adds to the data rate. They also require a considerable pre-processing at the encoder. This is not always feasible in many real-time transmission scenarios. In this chapter, we describe our proposed adaptive

concealment methods for subsampling-based MDC architectures [96–98]. Our adaptive methods make use of the bitstream-embedded parameters, hence not requiring any side information.

2.1 Temporal-Spatial Four Description MDC

A four description MDC architecture is shown in Figure 2.1. This architecture is used to develop various non-adaptive and adaptive error concealment approaches that are discussed in [29,64,91,92,96,97,231]. The original video sequence is first split along
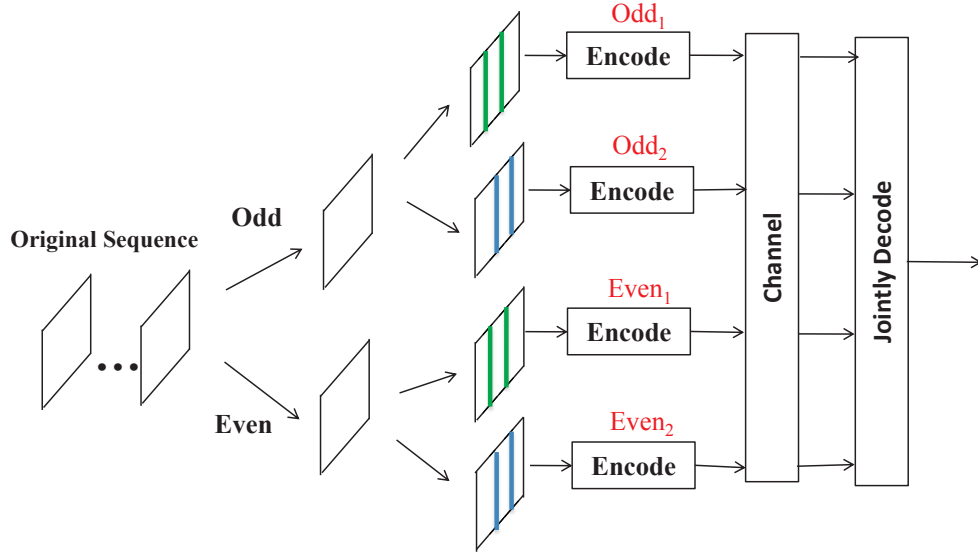
Fig. 2.1.: Temporal-spatial four description MDC architecture

temporal dimension into two subsequences. One is from all the odd-numbered frames and the other is from all the even-numbered frames. Each of the two subsequences is then partitioned along spatial dimension into two descriptions through horizontal downsampling. Odd columns in a frame form one description, and even columns form the other description. As a result, $Even_1$ and Odd_1 are from odd columns of even-numbered frames and odd-numbered frames respectively, while $Even_2$ and Odd_2 are from even columns of even-numbered frames and odd-numbered frames respectively. "Even" and "Odd" refer to even-numbered frames and odd-numbered frames. The

subscripts "1" and "2" denote odd columns and even columns of each frame at original resolution. For example, "$Even_1$" denotes the odd columns from even-numbered frames after horizontal downsampling. The newly generated four descriptions are encoded independently and are sent through the same or different channels. This is a Class A method that we discussed in Chapter 1 and which has good mismatch control at the cost of losing some coding efficiency [60]. We call two descriptions generated from the same frames "spatially neighboring" descriptions. Two pairs of spatially neighboring descriptions are: $Even_1$-$Even_2$ and Odd_1-Odd_2. We call two descriptions generated from the same column locations as "temporally neighboring" descriptions. Two pairs of temporally neighboring descriptions are: $Even_1$-Odd_1 and $Even_2$-Odd_2. The descriptions generated with the proposed splitting mechanism possess inherent spatial and temporal correlations. Spatial concealment is performed by applying a two-neighbor bilinear filter. For temporal concealment, the pixel value of the same position from the previous frame is used.

When the four descriptions are correctly received, each description can be decoded independently. The final reconstructed video is the combination of the four decoded sub-videos. However, when packet loss occurs during transmission, joint decoding is performed. Depending on the packet the network has suffered from, the decoder is presented with a particular decoding scenario. For four descriptions, there are 16 such scenarios. The following two schemes are developed using spatial and temporal concealment as default methods respectively. In the "default-spatial" concealment scheme, spatial concealment is used as the primary method and temporal concealment as a secondary method.

Table 2.1 shows the error concealment scheme for each packet loss scenario, where the first row indicates which descriptions of the Odd sequence are received and the first column indicates which descriptions of the Even sequence are received. "Odd_1+Odd_2" means both Odd_1 and Odd_2 sequences are received, "Odd_1" means only Odd_1 of the Odd sequence is received and "Loss" in the first row denotes that neither Odd_1 nor Odd_2 is received. Similarly, "$Even_1$+$Even_2$" denotes both $Even_1$ and $Even_2$ sequences

Concealment Methods	$Odd_1 + Odd_2$	Odd_1	Odd_2	Loss
$Even_1 + Even_2$	N/A	Spatial	Spatial	Temporal
$Even_1$	Spatial	Spatial	Spatial	Spatial-Temporal
$Even_2$	Spatial	Spatial	Spatial	Spatial-Temporal
Loss	Temporal	Spatial-Temporal	Spatial-Temporal	Frame Repeat

Table 2.1: Error concealment schemes: Default spatial scheme

are received, "$Even_1$" means only $Even_1$ of the Even sequence is received and "Loss" in the first column denotes neither $Even_1$ nor $Even_2$ is received. "Spatial-Temporal" means spatial concealment is done first then temporal concealment is done.

According to Table 2.1, when one description is received, such as Odd_1, we use spatial concealment to conceal Odd_2 first and then use temporal concealment to conceal $Even_1$ and $Even_2$. When two descriptions are received, if the two descriptions are from the same spatial correlation such as Odd_1 and Odd_2, we use temporal concealment to conceal $Even_1$ and $Even_2$, otherwise we use spatial concealment. When three descriptions are received, we use spatial concealment. When the four descriptions are lost at the same time, we use the previously decoded reference frame for concealment which is called "Frame Repeat" in the table.

Concealment Methods	$Odd_1 + Odd_2$	Odd_1	Odd_2	Loss
$Even_1 + Even_2$	N/A	Temporal	Temporal	Temporal
$Even_1$	Temporal	Spatial	Temporal	Spatial-Temporal
$Even_2$	Temporal	Temporal	Spatial	Spatial-Temporal
Loss	Temporal	Spatial-Temporal	Spatial-Temporal	Frame Repeat

Table 2.2: Error concealment schemes: Default temporal scheme

The *default temporal* scheme is summarized in table 2.2. The difference from the *default spatial* is that it uses temporal concealment as the primary method and spatial concealment as the secondary method. Note that in table 2.1 and table 2.2,

the concealment methods differ in 6 scenarios shaded in blue. For these scenarios, the *default Spatial* scheme uses the spatial concealment, whereas the *default Temporal* scheme uses the temporal concealment.

2.2   H.264 Bitstream-Based Adaptive Error Concealment

The above described *default spatial* and *default temporal* methods give good concealment performance only for either "higher" or "lower" motion videos respectively [64]. Table 2.3 summarizes our MB-level adaptive concealment scheme for different MB receiving scenarios. Our adaptive concealment approach for various MB

Concealment Methods	Odd_1 + Odd_2	Odd_1	Odd_2	Loss
$Even_1$ + $Even_2$	N/A	Adaptive	Adaptive	Temporal
$Even_1$	Adaptive	Spatial	Adaptive	Spatial-Temporal
$Even_2$	Adaptive	Adaptive	Spatial	Spatial-Temporal
Loss	Temporal	Spatial-Temporal	Spatial-Temporal	Frame Repeat

Table 2.3: Error concealment schemes: Adaptive scheme

loss scenarios is summarized in Table 2.3. The MBs from different descriptions having the same MB index (raster-scan order) are labeled as M_k; where "k" is the description index such that $k\epsilon\{Odd_1, Odd_2, Even_1, Even_2\}$.

1) **No Loss:** If each M_k is received, joint decoding without any concealment is done.

2) **One Description Loss:** When one of the M_k's is lost, *Adaptive Concealment Using Motion Vector Analysis* is done.

3) **Two Description Loss:** When two of the M_k's are lost, there are three possible scenarios: (a) When MBs from spatially correlated descriptions are lost (e.g. M_{Even_1} and M_{Even_2}), then the MBs are concealed from their respective temporally correlated descriptions (e.g. M_{Odd_1} and M_{Odd2}) using *Temporal Concealment*. (b) When MBs from temporally correlated descriptions are lost (e.g. M_{Even_1} and M_{Odd_1}), then the lost MBs are concealed from their respective spatially correlated descriptions (e.g.

M_{Even_2} and M_{Odd_2}) using *Spatial Concealment*. (c) When MBs from different spatial-temporal correlation are lost (e.g. M_{Even_1} and M_{Odd2}), then *Adaptive Concealment Using Motion Vector Analysis* is used.

4)**Three Description Loss:** When three of M_k's are lost (e.g. M_{Even_1}, M_{Even_2} and M_{Odd_1}), then first, *Spatial Concealment* and *Temporal Concealment* are done to conceal the correlated descriptions (e.g. M_{Odd_1} and M_{Even_2} respectively) and then *Spatial Concealment* is used to obtain the MB from the remaining description (e.g. M_{Even_1}).

5)**All Description Loss:** When MBs from all descriptions are lost, they are concealed from the previous frame by *Temporal Concealment.*

Notice that, Adaptive concealment method is used in 6 out of 16 possible scenarios. As seen earlier in Table 2.1 and Table 2.2, this scheme also differs in the same six scenarios (highlighted in yellow) where any of the basic concealment methods can be used for concealment. So, the decoder decides adaptively which of these basic methods are to be used based on a certain mathematical criteria.

We next describe our proposed two adaptive concealment methods that can select either spatial or temporal concealment for each lost MB based on H.264 MDC bitstream parameters. Our first method is based on using motion vectors (MVs) and our second method is to use error estimation for selecting the type of concealment.

2.2.1   Motion Vector Analysis Method

Our first adaptive method is based on the analysis of motion vectors (MVs) to classify an MB as *"Higher"* or *"Lower"* motion and then selecting either spatial or temporal concealment [96]. Recall that in a typical video encoder Inter prediction is done using motion compensated prediction (MCP). Motion estimation (ME) generates MVs to specify the amount of translational motion to do the MCP for Inter-coded pixel-blocks. ME is out of the scope of a typical standard, yet a "good" encoder implements ME that gives a fair idea of the amount of movement of that MB as compared to previous frame. We do not suggest any particular ME method for the encoder

but assume that it implements a fair ME algorithm. This is typically the case with most standard-compliant bitstreams utilizing the compression potential the way it is intended. We extract the motion information in the form of MVs from the received bitsream and use it to estimate the motion from a lost MB. This is described next. The term "macroblock (MB)" refers to a submacroblock in one of the four representations encoded using the H.264 standard [8].

Average Motion of a Macroblock: For a received MB, we estimate the average motion within it as the weighted average of all motion vectors associated with different partitions of that MB. Let M be a MB from a received slice and N be the total number of partitions in M. Let the i'th partition of M have pixel dimensions of $(P \times Q)^i$ and motion vectors as $((mv_x)^i, (mv_y)^i)$. For example, consider an MB with partition P_16x8 with motion vectors of (2,2) and (-4,6). In this case, $(P \times Q)^1 = (16 \times 8)$ and $(mv_x)^1 = 2$, $(mv_y)^1 = 2$. Similarly, $(P \times Q)^2 = (16 \times 8)$ and $(mv_x)^2 = -4$, $(mv_y)^2 = 6$. Now, $(\beta_x)_M$, the average absolute motion in the x-direction for M is given by:

$$(\beta_x)_M = \frac{\sum_{i=1}^{N} |(mv_x)^i| * (P \times Q)^i}{\sum_{i=1}^{N} (P \times Q)^i} \tag{2.1}$$

Similarly, in y-direction:

$$(\beta_y)_M = \frac{\sum_{i=1}^{N} |(mv_y)^i| * (P \times Q)^i}{\sum_{i=1}^{N} (P \times Q)^i} \tag{2.2}$$

where, $(\beta_y)_M$ is the average absolute motion in the y-direction for M. For the bipredictive mode present in a partition, the effective motion vectors are obtained by the weighted average of two motion vectors and used as $((mv_x)^i, (mv_y)^i)$ in the above equations. The average motion (γ) within the MB M is:

$$\gamma_M = \sqrt{(\beta_x)_M^2 + (\beta_y)_M^2} \tag{2.3}$$

Estimating Motion of a Lost Macroblock: Recall that the 4 descriptions are spatially and temporally correlated, we estimate the average motion of a lost MB from the average motion of the received MBs that are highly correlated to the lost MB. Let $\mathcal{M}$ be a set of correlated MBs; in which each member M_k has the same raster-scan order (MB index), but belongs to a different description. "k" is the description index such that $k \ \epsilon \ \{Odd_1, Odd_2, Even_1, Even_2\}$. Therefore $\mathcal{M} = \{M_{Odd_1}, M_{Odd_2}, M_{Even_1}, M_{Even_2}\}$ is a set of spatially and temporally correlated MBs for a particular MB index. The average motion within a lost MB ($M_l \ \epsilon \ \mathcal{M}$) is estimated from the received MBs from $\mathcal{M}$. When one of the M_k's is lost, the average motion within it ($\hat{\gamma}_{M_l}$) is estimated as:

$$\hat{\gamma}_{M_l} = \frac{\sum_{i=1}^{R} \gamma_{M_i}}{R} \qquad (2.4)$$

where "R" is the number of received MBs from $\mathcal{M}$ and γ_{M_i} is the average motion of a received MB M_i ($\epsilon \ \mathcal{M}$). If $R = 0$, no motion data is available for that MB.

Adaptive Concealment Using Motion Vector Analysis: When the concealment data is available from both the concealment methods above, the joint decoder makes a choice between them based on the motion analysis. Then a lost MB i.e. M_l ($\epsilon \mathcal{M}$) is concealed in the following steps:

1) The received and the lost MBs within the set
$\mathcal{M} = \{M_{Odd_1}, M_{Odd_2}, M_{Even_1}, M_{Even_2}\}$ are identified.

2) The average motion values of all the received MBs from $\mathcal{M}$ are obtained by equation 2.11.

3) The average motion within M_l is estimated by equation 2.6

4) A globally-defined threshold (T) is used on this estimated value of average motion to label the MB as either *"Higher"* or *"Lower"* motion. This threshold has units of a quarter-pixel and has a constant value for all video sequences.

5) If the MB is labeled as *"Higher"* motion, *Spatial Concealment* is used; otherwise (for an MB labeled as *"Lower"* motion) the *Temporal Concealment* is used.

2.2.2   Error Estimation Method

The goal of error concealment is to minimize the amount of error caused due to packet loss. An adaptive choice can be made using estimates of error introduced by using each method to conceal the video contents within the lost packets. This error depends on the codec and its parameters, the packetization strategy, the decoder error concealment and the video content [232]. We next describe our second adaptive concealment method based on error estimation at the decoder [97].

In this method, our MDC encoder is configured to preprocess a short segment of the sequence to compute sequence-specific parameters that are sent to the decoder with the picture parameter set (PPS) of the H.264 [8]-encoded bitstream. We develop a model to estimate total error (in terms of MSE) caused due to concealment that is used for the lost MBs. We call this "concealment error." The error due to quantization (quantization error) is modeled as being independent of the error due to packet loss (concealment error). We consider using an NR method based on analyzing lost MBs based on the parameters extracted from the bitstream. For a particular lost MB, only one concealment scheme (spatial or temporal) is used. At each concealment, one new type of concealment error is introduced and propagated across the reconstructed video. To adaptively select the best error concealment, it is useful to estimate the potential effects of a concealment when it is introduced. To be able to do this, we need to take into account the frame type (I/P or B) ($FRAMETYPE$), average distance from a reference frame (average of $DistToRef$), a property of the GOP structure, motion parameters such as the average estimated motion within a lost MB ($MOTM$) computed from neighboring motion vectors and an occurrence of scene change ($SceneChange$) [232, 233]. With these parameters, we assume that the contributions of spatial and temporal concealment errors to the total concealment error (denoted by $\hat{\epsilon}_c^2$) are separable. We assume this because for each lost MB, only one type of concealment is used [94].

Let $\hat{\epsilon}_c^2$ be the total concealment error. Let $\hat{\epsilon}_{Temp}^2$ and $\hat{\epsilon}_{Sp}^2$ be MSEs due to temporal and spatial concealment respectively.

$$\hat{\epsilon}_c^2 = \hat{\epsilon}_{Temp}^2 + \hat{\epsilon}_{Sp}^2 \tag{2.5}$$

Temporal Concealment Error

Error caused due to temporal concealment depends on *FRAMETYPE*, average of *DistToRef* and decoder concealment strategy. We copy pixels from temporally neighboring description to conceal a lost MB. Therefore, *MOTM* is also used as a model parameter.

Let $\hat{\epsilon}_{Temp,k}^2$ be the total MSE due to temporally concealed MBs of *FRAMETYPE* k ($k\epsilon\{I,P,B\}$).

$$\hat{\epsilon}_{Temp}^2 = \sum_{k\epsilon\{I,P,B\}} \hat{\epsilon}_{Temp,k}^2 \tag{2.6}$$

Now, consider $\hat{\epsilon}_{Temp,k}^2$. Let $\hat{\gamma}_{M_l}$ be the estimated average motion of a lost MB M_l as computed in [96]. Changing the notation slightly to accommodate *FRAMETYPE*, let $\hat{\gamma}_{j,i}$ be the estimated average motion of the ith temporally concealed MB of *FRAMETYPE* j ($j\epsilon\{P,B\}$). $\hat{\gamma}_{j,i}$ is used as a measure of *MOTM*. Let total number of temporally concealed j MBs be $N_{Temp,j}$. We use a linear model for $\hat{\epsilon}_{Temp,j}^2$.

$$\hat{\epsilon}_{Temp,j}^2 = C_{Temp,j} \sum_{i=1}^{N_{Temp,j}} \frac{\hat{\gamma}_{j,i}}{D_j} \tag{2.7}$$

where D_j is average of *DistToRef* that depends on *FRAMETYPE* and GOP structure. $C_{Temp,j}$ is a multiplier for the ith MB, modeled as being constant over all i's for a specific *FRAMETYPE* j.

Spatial Concealment Error

Spatial concealment is achieved by using a bilinear filter on the spatially neighboring description. Therefore we consider only *FRAMETYPE* and sequence-specific constants $C_{Sp,k}$ where $k\epsilon\{I,P,B\}$. We use the model that the total error due to spatial concealment is linearly related to the number of spatially concealed MBs.

$$\hat{\epsilon}_{Sp}^2 = \sum_{k\epsilon\{I,P,B\}} C_{Sp,k} \times N_{Sp,k} \tag{2.8}$$

where, $N_{Sp,k}$ is the count of lost k MB that are spatially concealed.

Adaptive Error Concealment Using Error Estimation The four description MDC encoder is configured to process a short segment of the sequence to determine sequence-specific parameters: $C_{Temp,j}$ ($j\epsilon\{P, B\}$) and $C_{Sp,k}$ ($k\epsilon\{I, P, B\}$) that are made available to the decoder using the sequence parameter set (SPS). For each *SceneChange* occurrence an SPS is sent with the new parameter estimates.

1) The decoder computes $\hat{\epsilon}^2_{Sp,M_l}$ and $\hat{\epsilon}^2_{Temp,M_l}$ for the lost MB (M_l) based on Equations 2.9 and 2.10.

$$\hat{\epsilon}^2_{Temp,M_l} = C_{Temp,j} \times \frac{\hat{\gamma}_{M_l}}{D_j} \tag{2.9}$$

$$\hat{\epsilon}^2_{Sp,M_l} = C_{Sp,k} \tag{2.10}$$

2) The decoder selects the concealment method as following:

$\hat{\epsilon}^2_{Sp,M_l} \leq \hat{\epsilon}^2_{Temp,M_l} \rightarrow$ *Spatial Concealment*

$\hat{\epsilon}^2_{Sp,M_l} > \hat{\epsilon}^2_{Temp,M_l} \rightarrow$ *Temporal Concealment*

In the next section, we present another subsampling-based MDC with adaptive error concealment for HEVC [9]-encoded bitstreams [98].

2.3   Spatial Subsampling-Based MDC

As shown in Figure 2.2, the original video sequence is split into two subsequences by using even and odd numbered columns of pixels in each frame. Each is independently encoded using an HEVC [9] encoder to produce two encoded bitstreams. We denote the descriptions as the *Even* and *Odd* descriptions. One description is known as the "neighboring" description of the other. Note that the frame rate for each encoded description is the same as the original video sequence. When transmitted across the network the descriptions may undergo packet-loss. The receiver uses an
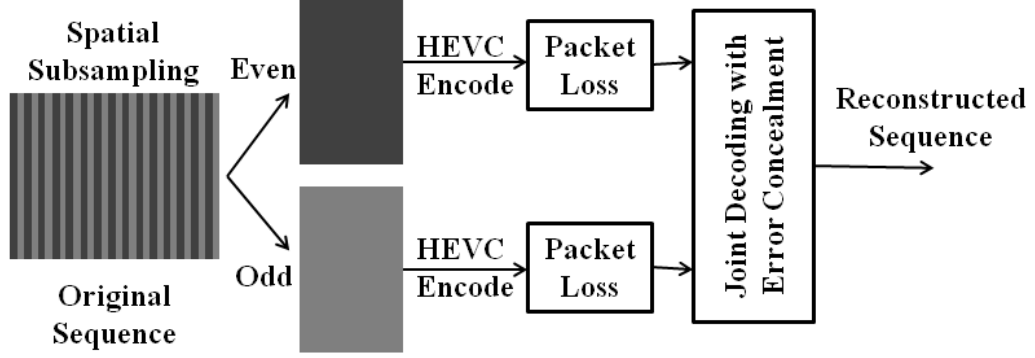
Fig. 2.2.: Our spatial subsampling-based MDC framework.

adaptive error concealment method to jointly decode the descriptions to reconstruct the video. The use of independent encoding loops for the two descriptions is used to provide a better robustness against error propagation.

The next section describes an adaptive error concealment for this architecture.

2.4 HEVC Bitstream-Based Adaptive Error Concealment

Coding tree unit (CTU) is the basic coding structure used in the HEVC standard [9]. It is analogous to MB defined in the H.264 standard. A slice consists of an integer number of CTUs. When a packet is not received before the display time of its video contents, all CTUs within that packet are lost. A lost coding tree unit (CTU) is concealed from a set of CTUs (if received) in the spatial and temporal neighborhood of the lost CTU. We consider the top (T), left (L), temporally collocated (C) and neighboring description (N) CTUs and the entire previous frame for concealment. As shown in Figure 2.3, we describe the following basic concealment methods.

Spatial Concealment (Sp)

A Lost CTU is concealed from the CTU with the same index in its neighboring description. A bilinear filter is used to interpolate the pixel values. e.g. a lost CTU from *Even* is concealed from *Odd*.

Temporal Concealment
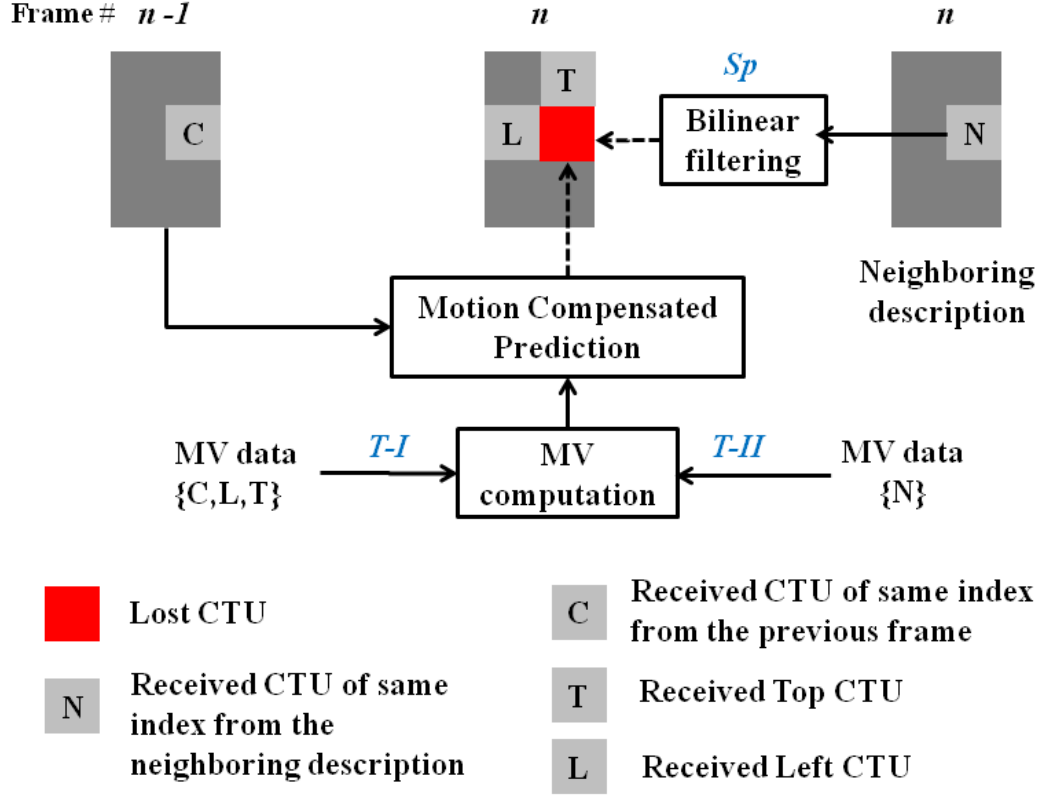
Fig. 2.3.: Basic concealment schemes.

A lost CTU is concealed using a motion compensated block of pixels using the previous frame as the reference frame and a motion vector denoted by mv_x^c, mv_y^c. We use this motion vector for the entire CTU and it is computed as the weighted median of all motion vectors belonging to the CTUs used for concealment in the following way:

We denote the set of CTUs used for concealment by $\mathcal{S}_c$. We extract the coded information such as prediction unit (PU) partitions, their motion vectors and the reference frames for CTUs belonging to $\mathcal{S}_c$. A motion vector is scaled using the distance to the reference frame. Distance to the reference frame is the difference in the display count of the current frame and that of the reference frame used to obtain the motion compensated prediction signal for that PU.

$$mv_{x,sc} = \frac{mv_x}{d}, mv_{y,sc} = \frac{mv_y}{d},\tag{2.11}$$

where d is the distance to the reference frame and $(mv_{x,sc}, mv_{y,sc})$ is the scaled motion vector.

Let $\mathcal{M}$ be a set of scaled motion vectors from the prediction units (PUs) belonging to $\mathcal{S}_c$ and $\mathcal{W}$ be a set of motion vector weights expressed as the numbers of pixels belonging to the corresponding PUs. For example, a motion vector $(3, -5)$ representing a PU of size 32×32 has the weight $32 \times 32 = 1024$. Therefore, the values 3 and -5 have the weight of 1024. Weights are considered as the number of occurrences of each vector. Median of m numbers each with multiple occurrences is computed by placing the numbers in ascending order with each number appearing so many times as its occurrence (or weight) and selecting the number at the center of the order. This is done separately for X- and Y-components of scaled motion vectors. Thus, mv_x^c and mv_y^c are computed as weighted median motion vector components using the set of scaled motion vectors, $\mathcal{M}$ and their corresponding weights, $\mathcal{W}$. Based on $\mathcal{S}_c$ i.e. the set of CTUs used for concealment, we describe the following two temporal concealment methods:

Method T-I: A Lost CTU is concealed using motion information from the CTU having the same index from the previous frame from the same description and the CTUs at top and left of the lost CTU in the current frame. Therefore, $\mathcal{S}_c = \{C, L, T\}$.

Method T-II: A Lost CTU is concealed using motion information from the CTU with same index in its neighboring description. Therefore, $\mathcal{S}_c = \{N\}$.

For temporal concealment, we use motion compensated concealment using the median motion vector computed from spatially or temporally neighboring CTU information. Therefore, *T-I* or *T-II* uses the previous reconstructed frame from the same description and the displacement vector computed using a set of neighboring CTUs. Whereas, *Sp* uses reconstructed pixels from the current frame of the neighboring description. To do adaptive concealment for a lost CTU, *Sp* and *T-II* are considered as candidates.

***Inter*-to-*Intra* Ratio:** To determine the most suitable of the two candidates, we first examine the same index-CTU from the neighboring description by counting the numbers of pixels coded as *Inter* and *Intra*. We denote the ratio by β_N.

$$\beta_N = \frac{\text{Number of } \textit{Inter} \text{ pixels in CTU ``}N\text{''}}{\text{Number of } \textit{Intra} \text{ pixels in CTU ``}N\text{''}} \tag{2.12}$$

where N denotes the same-index CTU from the neighboring description. A large number of *Intra* pixels relative to *Inter* is considered as lower dependency on the previous frame. This is due to the rate-distortion mode decision by the encoder that may be indicative of lower spatial prediction errors produced due to higher spatial consistency, a complex motion of the existing objects, new object being introduced or scene change in the current frame. Therefore, if β_N is lower than a threshold (τ_β), we use Sp that uses pixels from the current frame of the neighboring description, rather than *T-II*, a motion compensated temporal method.

Motion Non-Uniformity: If β_N is higher than τ_β, a second criterion, non-uniformity of motion vectors in a CTU, is tested. We process the motion vectors from *Inter*-coded PUs of the same-index CTU from the neighboring description. Let $(mv^c_{x,N}, mv^c_{y,N})$ be the weighted median motion vector using the same-index CTU from the neighboring description, as indicated in Method *T-II*. Let $(mv^i_{x,sc,N}, mv^i_{y,sc,N})$ be the scaled motion vector from ith PU of the same-index CTU from the neighboring description. Assume that ith PU consists of w_i pixels. Now, we compute motion non-uniformity γ_x and γ_y in X- and Y-direction using weighted sum of absolute differences (SAD):

$$\gamma_x = \frac{\sum_{i=1}^{R} w_i \cdot |mv^i_{x,sc,N} - mv^c_{x,N}|}{\sum_{i=1}^{R} w_i}, \tag{2.13}$$

$$\gamma_y = \frac{\sum_{i=1}^{R} w_i \cdot |mv^i_{y,sc,N} - mv^c_{y,N}|}{\sum_{i=1}^{R} w_i}, \tag{2.14}$$

$$\gamma = \gamma_x + \gamma_y, \tag{2.15}$$

where R is total number of PUs and γ is total motion non-uniformity in the CTU. If γ is higher than a threshold τ_γ, Sp is chosen. Otherwise *T-II* is done using $(mv^c_{x,N}, mv^c_{y,N})$ as motion vector for motion compensated temporal concealment.

Our adaptive method is summarized below.

Adaptive Concealment Method

Require: A lost CTU where its corresponding CTU from the neighboring description is received.

Get motion vector data and mode information of the received neighboring CTU.

if $\beta_N < \tau_\beta$ **then**

Conceal the lost CTU using *Sp*. {Higher spatial dependency}

else if $\gamma > \tau_\gamma$ **then**

Conceal the lost CTU using *Sp*. {Motion non-uniformity}

else

Conceal the lost CTU using *T-II*, with $(mv_{x,N}^c, mv_{y,N}^c)$ as motion vector for motion compensated temporal concealment.

The Proposed concealment Scheme

Our proposed adaptive concealment scheme is shown in Figure 2.4 using different loss scenarios. The CTUs from *Even* and *Odd* descriptions having the same index are considered during the lost CTU concealment.

1) **No Loss:** If both CTUs are received, decoding without any concealment is done.

Received Descriptions	Concealment Method
Even + Odd	None
Even	*Adaptive*
Odd	*Adaptive*
None	*T-I*

Fig. 2.4.: Adaptive concealment scheme.

2) **One Description Loss:** When a CTU from either *Even* or *Odd* description is lost, *Adaptive* concealment is done according to the method described above.

3) **Two Description Loss:** When the same-index CTUs from both descriptions are lost, they are concealed using Method *T-I*.

2.5 Experimental Results

In this section, we present the experimental results of our three adaptive error concealment methods for test video sequences [96–98].

2.5.1 Network Channel Model

In a typical network scenario, packet loss patterns are usually bursty. Therefore, simply using a symmetric packet loss model cannot represent the realistic network transmission characteristics. In this work, a Gilbert model is used as the channel model for burst packet loss [29, 234]. Packet loss rate considered is generally between 0 and 0.6, and burst-length is between 2 and 20. When packet loss rate is small, burst length is large; and vice versa [235]. The theoretical details of Gilbert model using state transition probabilities are presented in [29, 235]. The parameters for the Gilbert model in our experiments are listed in Table 2.4.

Table 2.4: Gilbert model parameters for various packet loss rates: Adaptive error concealment

Loss Rate	5%	10%	15%	20%	25%	30%
Burst Length	5	5	4	4	3	3

2.5.2 Motion Vector Analysis Method

We first compare the performance of our motion vector analysis method with the two non-adaptive *default spatial* (*Sp*) and *default temporal* (*Temp*) concealment methods. The temporal-spatial four description architecture with adaptive concealment methods is implemented by modifying the H.264 reference software: JM version 17.1. Three video sequences *Mother-Daughter*, *News* and *Foreman* are used in our experiments. The test sequences used are CIF resolution at 30 frames/sec with 200-

frame length, thus each description has 100 subframes at 15 frames/sec. The coding structure is "IBBBPBBBP...", with I-frame refresh every 15 subframes in each description. The quantization parameters for I frame and P frame are 18, 22, 26, 30, 34 and 38 and the deblocking filter is disabled.

I-frames are assumed error-free. Packetization was carried out as fixed MBs per packet (22 MBs/packet). All the experiments are run 100 times with different permutations of lost packets for each packet loss rate and the results represent the average of the PSNR obtained from these loss patterns.

Figures 2.5 and 2.6 show the packet loss performance of our adaptive method at packet loss rate from 5% to 30% for the *Mother-Daughter* and the *News* sequences respectively. For the *Mother-Daughter* sequence, *Sp* slightly outperforms *Temp* and for the *News* sequence, *Temp* clearly out performs *Sp*. In both the cases, our proposed method does better than *Sp* and *Temp*. This is indicative of the fact that our method is adaptive to the amount of motion present in the sequence. As seen in Figure 2.8 and 2.9, there is a clear visual difference between the quality of images obtained from different methods. Our proposed method has produced sharper images with less distortions.

For the *Foreman* sequence, the foreground figure occupies a large portion of an image, resulting into a large average absolute motion. Most MBs have significantly large motion vectors. As seen in Figure 2.7, *Sp* outperforms *Temp* in an obvious way. Our proposed method is close to *Sp*, but performs slightly worse than *Sp*. In this case, ideally, our proposed method should give the same performance as *Sp* if not better. However, our analysis of the motion is solely based on the received motion vectors which may not accurately represent the true motion. The threshold (T) that we use to classify motion, is fixed for all sequences. Hence, it adapts to the motion, with a fixed algorithm, to choose among the best concealment methods for each sequence. In Figure 2.7, our proposed method does better than *Temp* with a significant margin and is very close to *Sp*. This shows that our proposed method has identified the
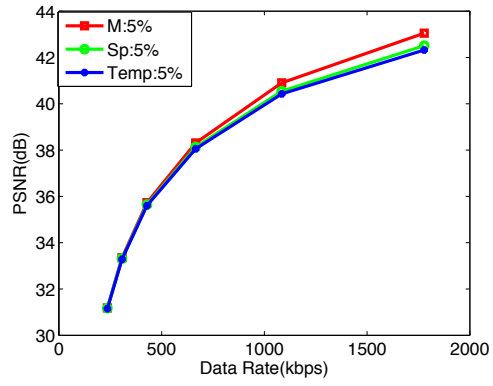
(a) *Packet Loss: 5%*
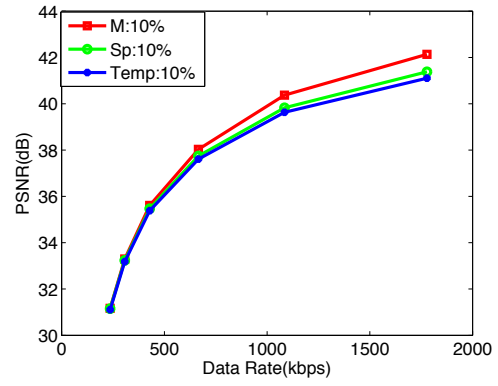
(b) *Packet Loss: 10%*
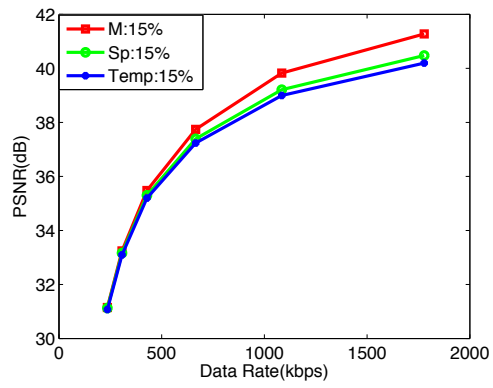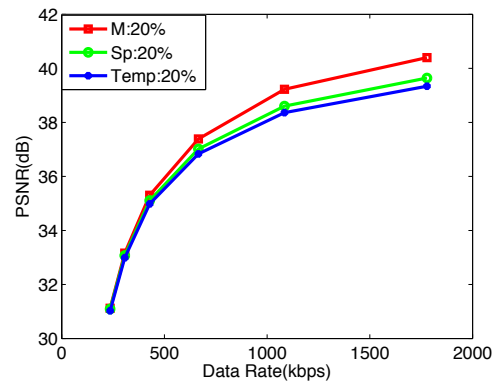
(c) *Packet Loss: 15%*
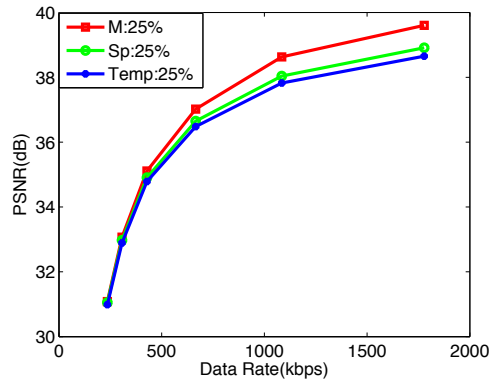
(d) *Packet Loss: 20%*
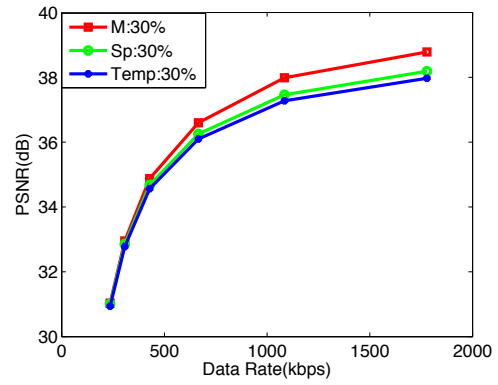
(e) *Packet Loss: 25%*

(f) *Packet Loss: 30%*

Fig. 2.5.: Packet loss performance comparison for the *Mother-Daughter* sequence.
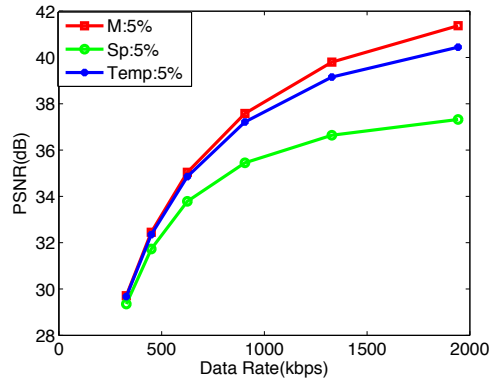
(a) *Packet Loss: 5%*
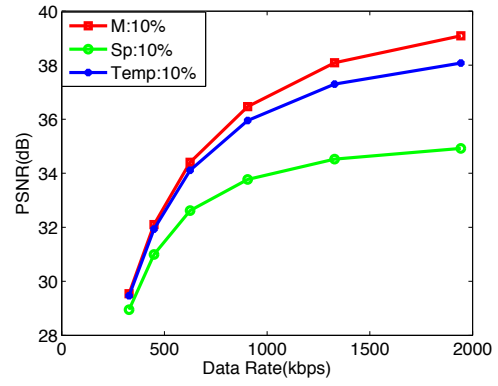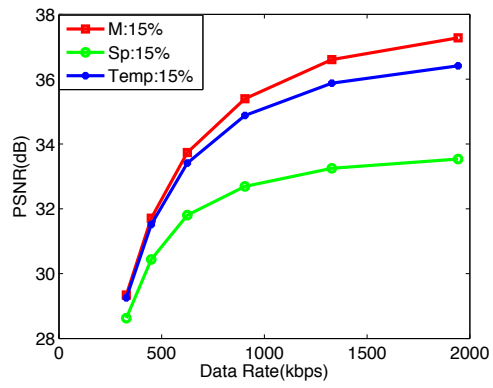
(b) *Packet Loss: 10%*
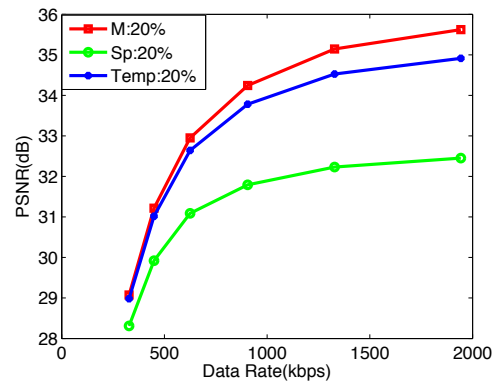
(c) *Packet Loss: 15%*
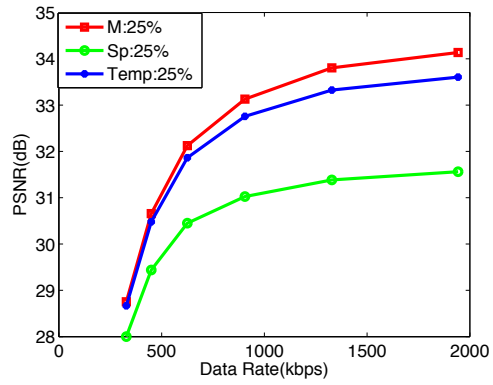
(d) *Packet Loss: 20%*
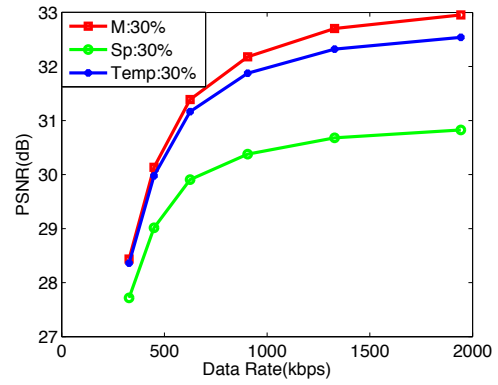
(e) *Packet Loss: 25%*

(f) *Packet Loss: 30%*

Fig. 2.6.: Packet loss performance comparison for the *News* sequence.

(a) *Packet Loss: 5%*

(b) *Packet Loss: 10%*

(c) *Packet Loss: 15%*

(d) *Packet Loss: 20%*

(e) *Packet Loss: 25%*
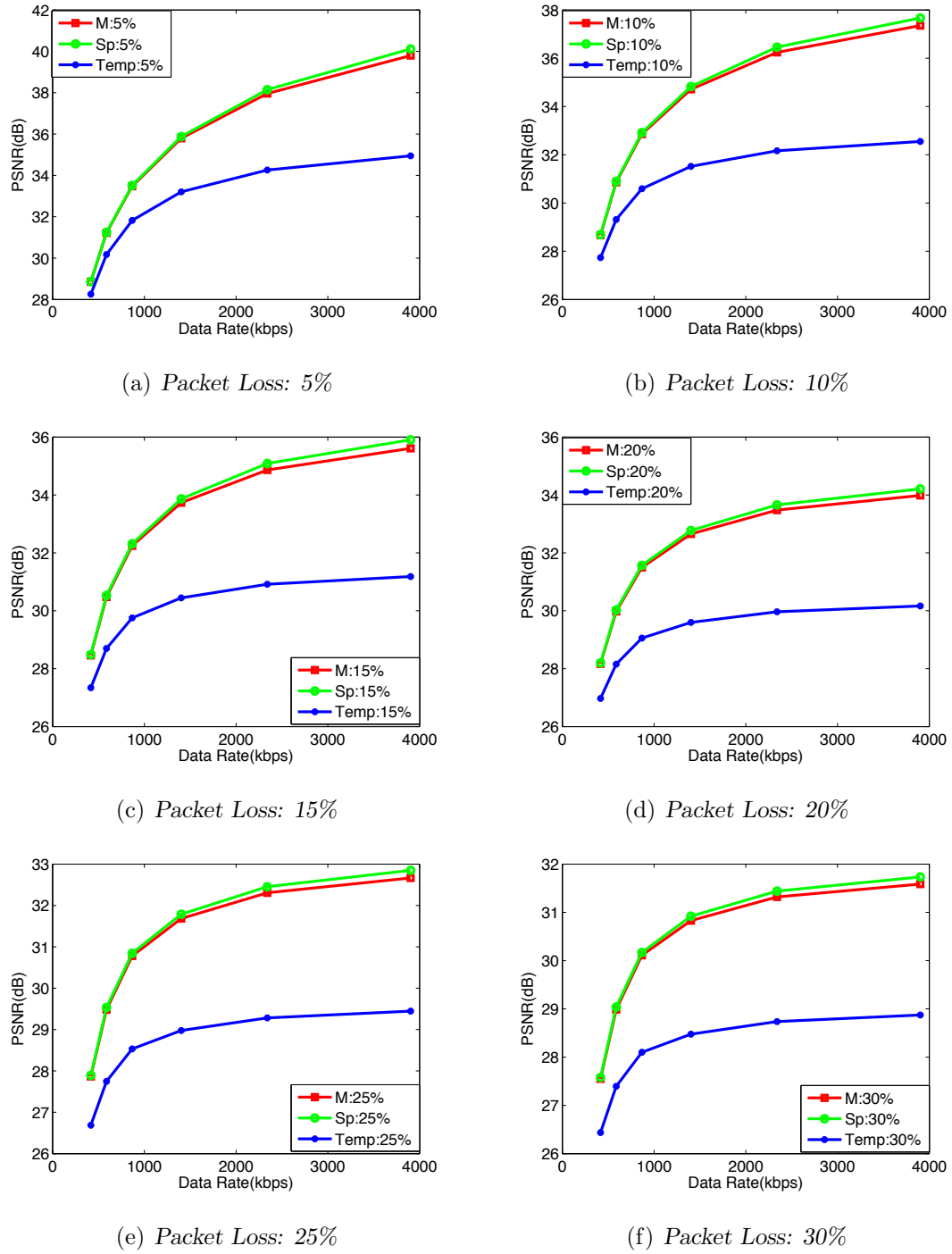
(f) *Packet Loss: 30%*

Fig. 2.7.: Packet loss performance comparison for the *Foreman* sequence.

presence of *"Higher"* motion based on the motion vector analysis and chosen the correct concealment method for most of the lost MBs.

(a) M

(b) Sp

(c) Temp

(d) No Packet Loss

Fig. 2.8.: Performance comparison for the *Mother-Daughter* sequence with identical packet loss against no packet loss.

Therefore, we can conclude from Figure 2.5 and 2.6 that our proposed method has an obvious improvement over the other two methods for sequences with a combination of *"Higher"* and *"Lower"* absolute motion. For a sequence containing *"Higher"* absolute motion, our proposed method has an obvious improvement over *Temp* and is

(a) M

(b) Sp

(c) Temp

(d) No Packet Loss

Fig. 2.9.: Performance comparison for the *News* sequence with identical packet loss against no packet loss.

close to *Sp* (as shown in Figure 2.7). In general, our proposed method has improved the previous work by adapting to any type of sequences.

2.5.3 Error Estimation Method

We first describe the error estimation performance with parameter estimation at the encoder and the next describes the adaptive error concealment performance at the

decoder subjected to simulated packet loss. This set of experiments was performed at the encoder using 25 frames of each description. The quantization parameters (QP) used for I frames are 24, 28 and 32. P and B frames used 26, 30 and 34 as QPs. Each experiment is repeated 25 times with different permutation of packet loss patterns. The sequence-specific parameters are determined as the ratio of the average (taken over all data points) actual MSE to the average (taken over the corresponding data points) of summation terms from the error estimation expression. Table 2.5 summarizes the estimated parameters for three tested sequences. These parameters are computed only at the beginning of each sequence and applied for the whole sequence of 200 frames.

Table 2.5: Estimated parameters for test sequences ($\times 10^{-3}$)

Sequence	$C_{Sp,P}$	$C_{Sp,B}$	$C_{Temp,P}$	$C_{Temp,B}$
Bridge-Close	4.14	2.07	1.78	0.89
Foreman	2.1	1.0	1.31	0.66
Football	3.75	1.87	5.16	2.58
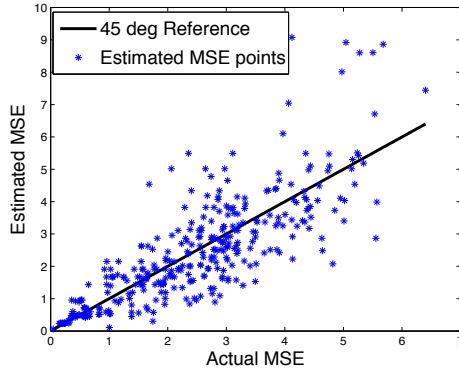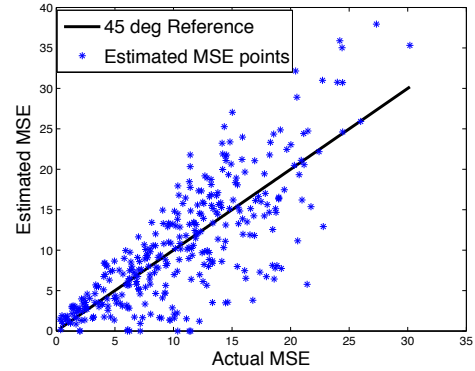
(a) *Spatial concealment error* (b) *Temporal concealment error*

Fig. 2.10.: Error estimation performance for the *Foreman* sequence.

(a) *Spatial concealment error* (b) *Temporal concealment error*
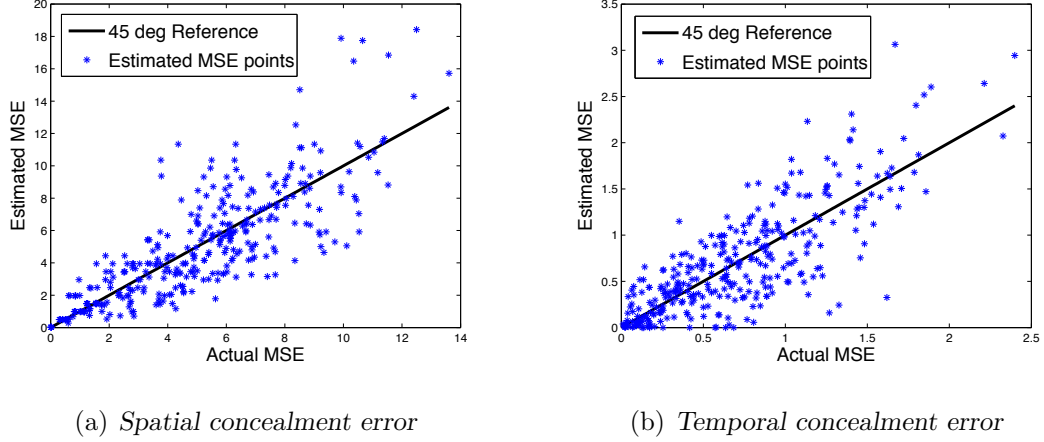
Fig. 2.11.: Error estimation performance for the *Bridge-Close* sequence.

Figure 2.10 and Figure 2.11 illustrate the results of our experiments with the *Foreman* and the *Bridge-close* sequences. The results are presented as scatter plots of estimated MSE Vs Actual MSE. The density of data points is higher close to the 45 degree reference line. Some estimates are falsely indicating higher/lower MSE than the actual MSE. This is because firstly, the encoder is allowed to process only a short sequence (25 frames/description) and secondly, the depth of bitstream-parsing is limited to extracting relatively high-level parameters e.g. DCT coefficient information is not considered in our framework. Both these constraints are used to maintain a reasonable computational complexity at the encoder and the decoder. The results presented below demonstrate that this level of accuracy is acceptable to adaptively select spatial or temporal concealment for a lost MB.

Now we compare the performance of the adaptive error concealment method using error estimation (*"Proposed"*) with the motion vector analysis method (*"Old-ad"*) and the non-adaptive *default spatial* (*Spatial*) method. The experimental setup was identical to that of motion vector analysis method.

Figure 2.12 shows the comparison for the *Bridge-Close* sequence. In this case, *"Proposed"* and *"Old-ad"* clearly outperform *"Spatial"* in terms of PSNR. *"Spatial"* is based on spatial concealment by default and *Bridge-close* has relatively lower mo-
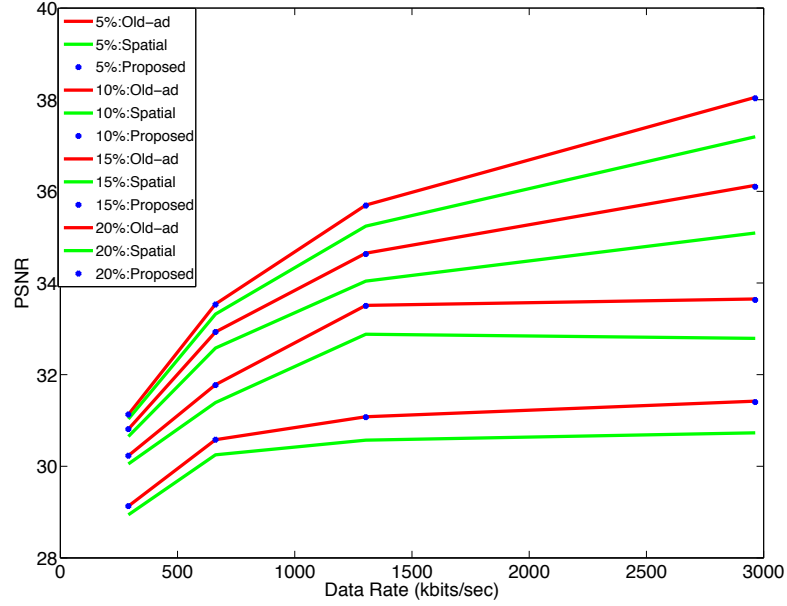
Fig. 2.12.: Packet loss performance for the *Bridge-Close* sequence.

tion. Therefore both adaptive methods give better performance by selecting temporal concealment for most lost MBs. Figure 2.13 shows the comparison for the *Foreman* sequence. Here, *"Proposed"* and *"Spatial"* perform significantly higher than *"Old-ad"* in terms of PSNR. This is indicative of the fact that our new adaptive method has correctly identified the best concealment strategy (spatial) for most lost MBs using error estimation, whereas *"Old-ad"* method which is based on a globally fixed threshold has failed to choose the best concealment for most MBs, hence performing worse than the other two. Figure 2.14 shows the comparison for the *Football* sequence, where *"Proposed"* outperforms *"Old-ad"* by a narrow margin and performs equally as that of *"Spatial"* in terms of PSNR. The margin is less because *Football* contains relatively higher motion and the *"Old-ad"* identifies correctly more MBs containing high motion.
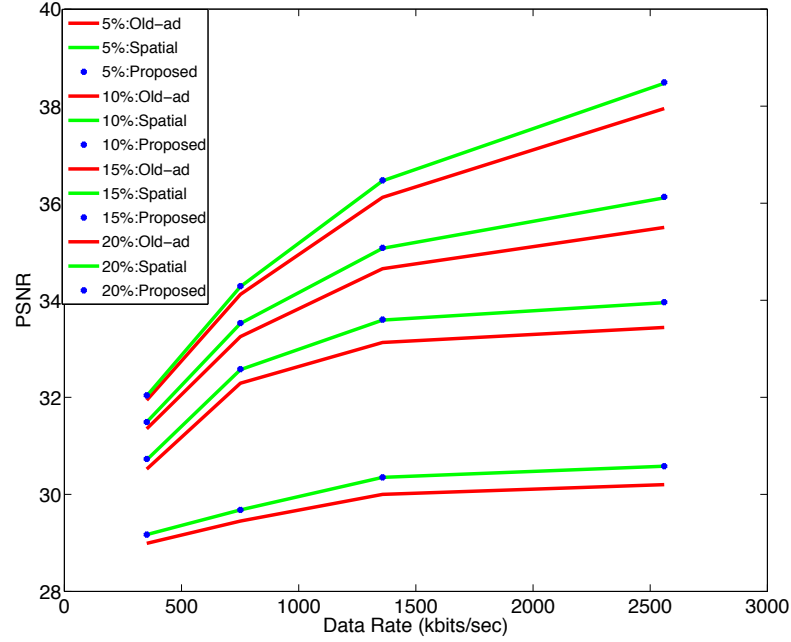
Fig. 2.13.: Packet loss performance for the *Foreman* Sequence.

Therefore, for each of the tested sequences, our proposed method outperforms one of the two other methods in terms of PSNR. It performs equally as the other, hence making itself adapt to any type of sequence based on error estimation.

2.5.4 HEVC Bitstream-Based Concealment Method

The experiments are implemented by modifying the HEVC reference software HM 8.2 decoder. Three video sequences *RaceHorses, BasketBallDrill* and *PartyScene* of resolution 832×480 are used for our experiments. The frame rate of *BasketBallDrill* and *PartyScene* is 50 fps and for *RaceHorses* it is 30 fps. For each sequence, 200 frames of the two spatially-subsampled descriptions are encoded using a "IPPP..." GOP structure with IDR-refresh every 16 subframes. The quantization parameters (QP) used are 20, 24, 28 and 32. The deblocking filter and SAO are disabled. Packetization is done such that each packet contains no more than 1500 bytes. We used
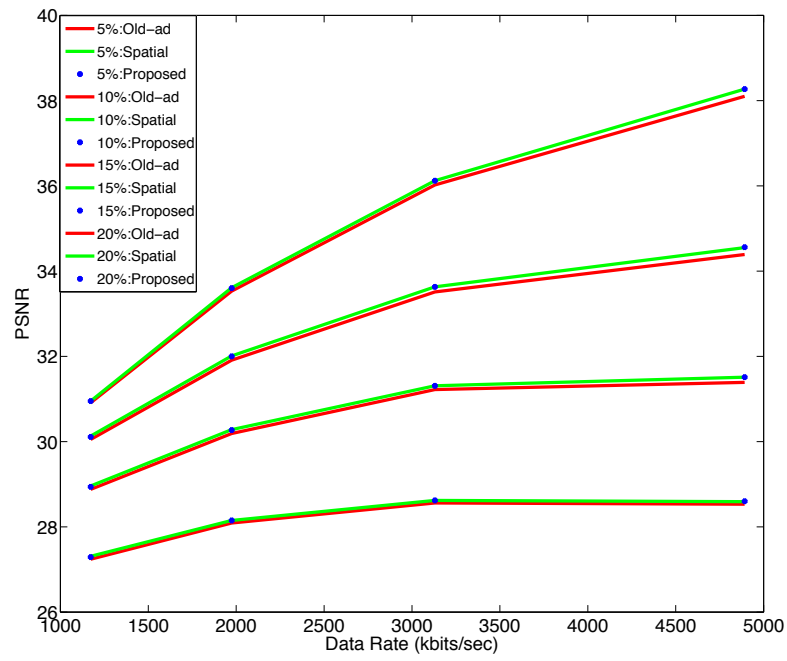
Fig. 2.14.: Packet loss performance for the *Football* Sequence.

(a) *Old-ad* (b) *Proposed*

Fig. 2.15.: Performance comparison for the *Foreman* sequence with identical (10%) PLR.

adaptive concealment thresholds $\tau_\beta = 10$ and $\tau_\gamma = 1$ for all sequences.

Side Reconstruction Performance: When the output video is reconstructed using only one description, this is known as side reconstruction and it is done by decoding the received description and concealing the lost one [34, 64]. When an *Even* or *Odd* description is totally lost, we use *Sp* to conceal the totally lost description. The results represent the average of the two loss cases i.e. lost-*Even* and lost-*Odd* description. Table 2.6 and Tabel 2.7 present the side reconstruction performance for two sequences, *RaceHorses* and *BasketBallDrill* respectively, encoded using different QPs. SDC represents single description reconstruction performance, while MDC has two cases: one description received (*Even* or *Odd*) and both descriptions received.

When the receiver can only receive a half of the data due to bandwidth limitation

Table 2.6: *RaceHorses* sequence

SDC		MDC			
		One (*Even* or *Odd*)		Two (*Even* + *Odd*)	
PSNR (dB)	Data Rate (kbps)	PSNR (dB)	Data Rate (kbps)	PSNR (dB)	Data Rate (kbps)
40.55	10775.1	32.52	7404.1	40.94	14808.2
36.95	5568.9	31.91	4187.3	37.54	8374.5
33.42	2693.5	30.79	2101.1	34.50	4202.2
30.54	1404.5	29.35	1032.7	32.00	2065.3

or network congestion, our proposed MDC method provides a graceful degradation by providing one description and concealing the loss. In this case a set of either even or odd columns from each frame is completely lost and concealed from the other set of columns. The concealment method produces an acceptable quality of the reconstructed video. When clients start receiving second description, the performance is improved. In case of single description coding, if clients cannot receive data at the

Table 2.7: *BasketBallDrill* sequence

SDC		MDC			
		One (*Even* or *Odd*)		Two (*Even* + *Odd*)	
PSNR (dB)	Data Rate (kbps)	PSNR (dB)	Data Rate (kbps)	PSNR (dB)	Data Rate (kbps)
40.87	6693.5	32.77	4803.5	41.40	9606.9
38.04	3732.5	32.38	2723.9	38.94	5447.7
35.28	2077.1	31.68	1496.1	36.44	2993.1
32.82	1166.6	30.66	823.15	34.12	1646.3

rate required for transmission of encoded video, to avoid disruptions in displaying the video contents, a lower quality video needs to be transmitted by the sender. Switching to a lower quality video can also involve a noticeable time delay due to the receiver sending feedback to the sender. Whereas, our proposed MDC method is able to reconstruct the contents based on the state of received descriptions.

Packet Loss Performance: Packet loss is caused when a network packet is not delivered at the receiver at all or delivered after the display time of its video contents. We now present the results of our experiments under packet loss conditions. A Gilbert model is used to simulate packet loss pattern. When packet loss rate is small, burst length is large; and vice versa [235]. We used burst length of 5 for 5% and 10%, 4 for 15% and 20%, and 3 for 25% packet loss. IDR-frames are assumed error-free. Three sequences *RaceHorses*, *BasketBallDrill* and *PartyScene* encoded with different QPs are tested for various loss rates. Each experiment is repeated 50 times with different permutation of packet loss patterns for loss rates of 5%, 10%, 15% (and 20%, 25% for *PartyScene*). The results represent the average of the Luma PSNR obtained from these different loss patterns. The performance of our proposed adaptive method (represented by *"MDC"*) is compared with SDC that uses *T-I* temporal concealment

(represented by *"SDC"*).

Figure 2.16 and Figure 2.17 present the performance for *RaceHorses* and *Basket-*

(a) Lossless

(b) Packet Loss = 5%.

(c) Packet Loss = 10%.
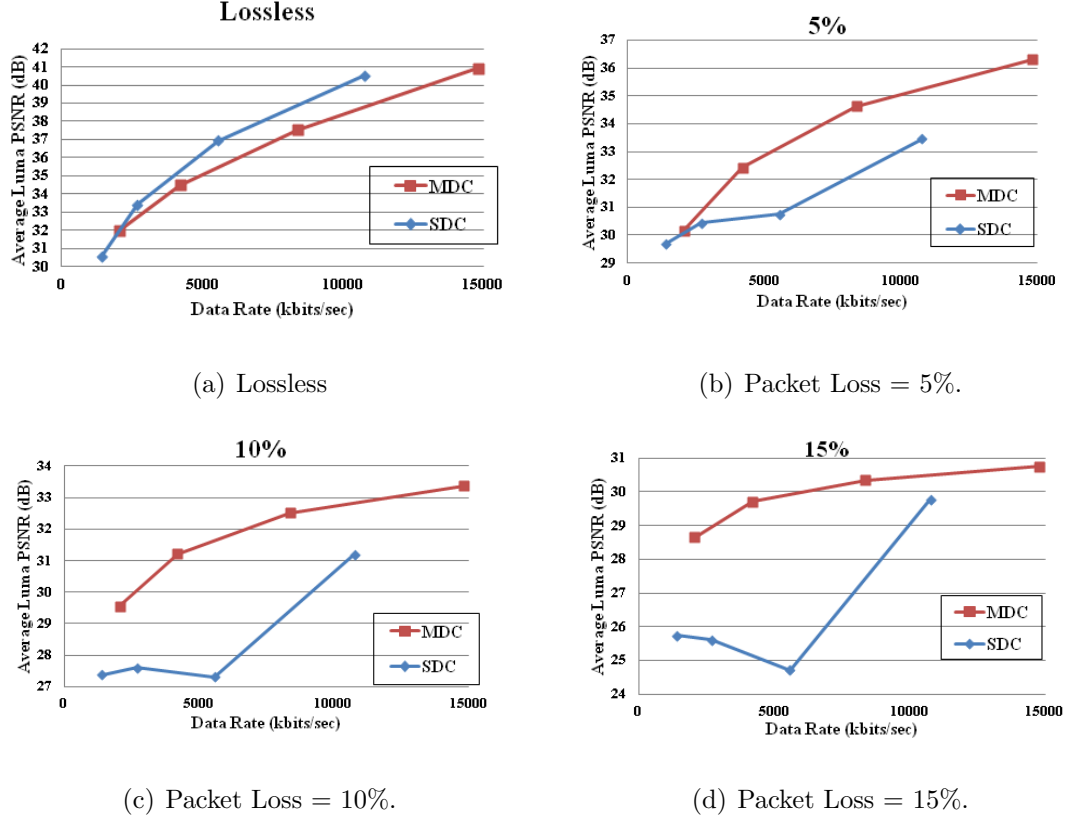
(d) Packet Loss = 15%.

Fig. 2.16.: Packet loss performance comparison for *RaceHorses* sequence: MDC Vs. SDC

BallDrill respectively. For both sequences, SDC outperforms MDC in "Lossless" case. Our MDC is based on spatial subsampling before encoding. Two descriptions are generated from separate HEVC encoding loops. Therefore, MDC loses some coding efficiency in the process as expected. When "5%" packets are lost, MDC begins to outperform SDC for higher data rates. For *RaceHorses*, MDC performs equally as or better than SDC for all data rates. For *BasketBallDrill*, MDC performs better than SDC at rates higher than approximately 3000 kbits/sec. For the "10%" and "15%" cases, MDC clearly outperforms SDC in terms of PSNR for both sequences.

(a) Lossless

(b) Packet Loss = 5%.

(c) Packet Loss = 10%.
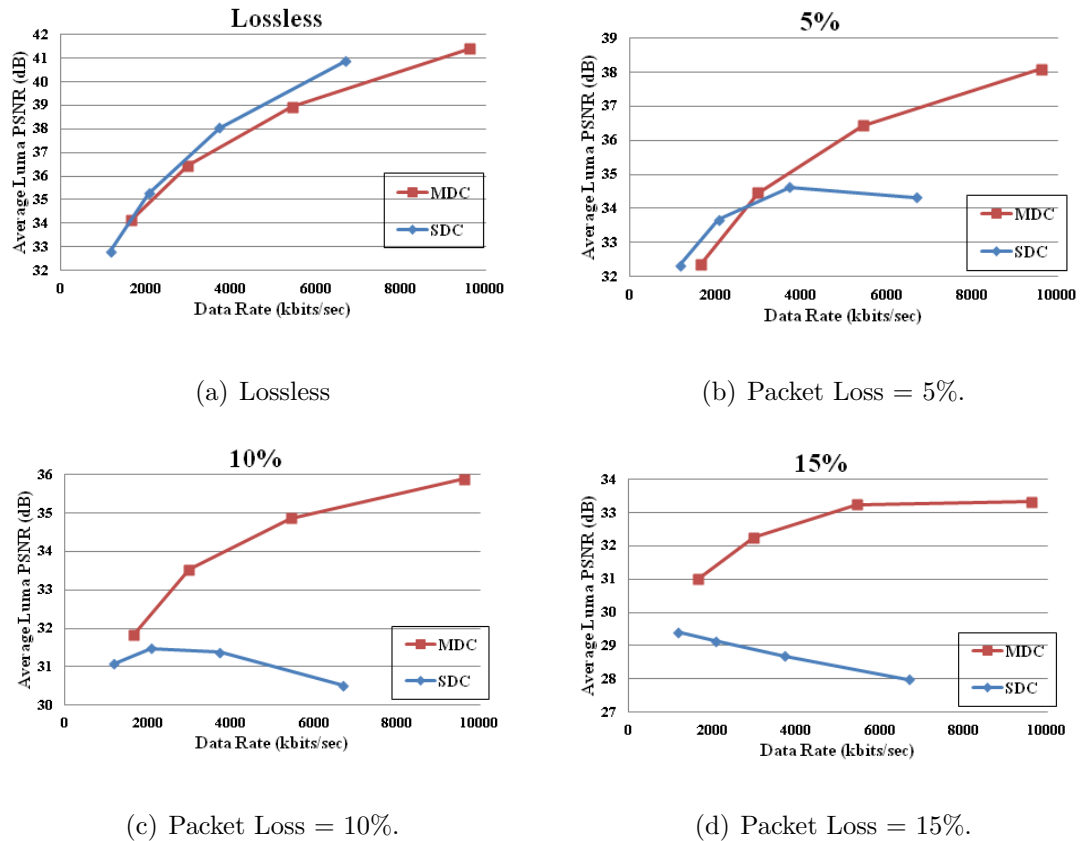
(d) Packet Loss = 15%.

Fig. 2.17.: Packet loss performance comparison for *BasketBallDrill* sequence: MDC Vs. SDC

Note that, SDC has produced non-smooth rate-distortion curves, indicating a severe degradation of performance. It shows that increase in the data rate does not ensure a better performance in terms of PSNR.

As shown in Figure 2.18, for *PartyScene*, SDC performs better than MDC for "Lossless", "5%" and "10%" cases. For cases "15%", "20%" and "25%", MDC and SDC curves are close to each other at most data rates except one high PSNR value for SDC, with one performing better than the other over certain ranges of data rates. We observe that, MDC curves are smooth, with PSNR increasing with the data rate. However, SDC curves exhibit unreliable trend of PSNR Vs. data rate. *PartyScene* sequence contains many pixel-level details and a lower QP may have been more effective
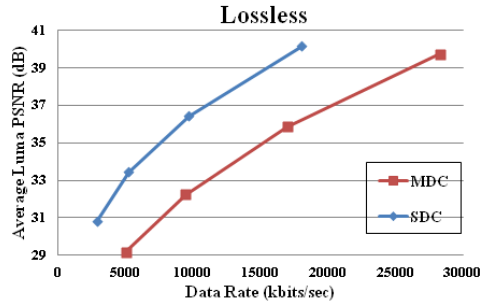
(a) Lossless
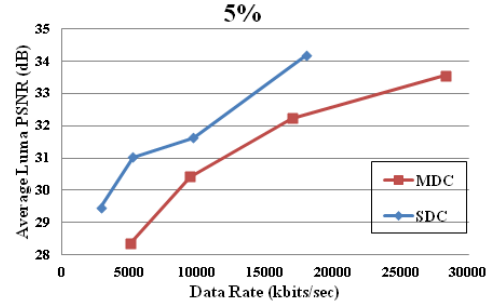
(b) Packet Loss = 5%.
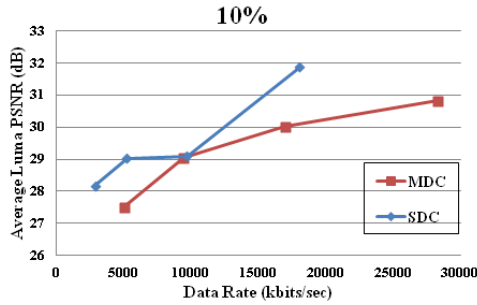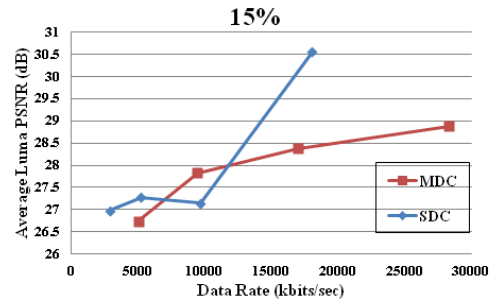
(c) Packet Loss = 10%.
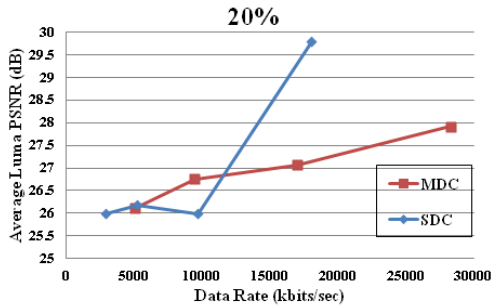
(d) Packet Loss = 15%.

(e) Packet Loss = 20%.
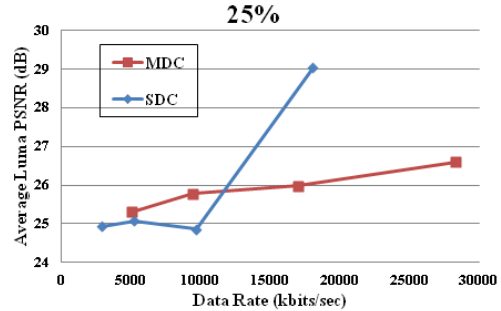
(f) Packet Loss = 25%.

Fig. 2.18.: Packet loss performance comparison for *PartyScene* sequence: MDC Vs. SDC

to produce better reconstructed pixels than the effects of adaptive error concealment method. A further analysis is needed to support this claim.
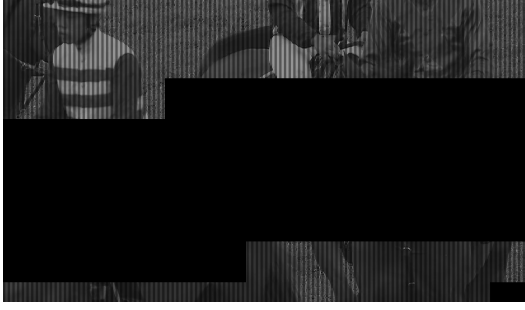
(a) Without Concealment   (b) With Adaptive Concealment

(c) Without Concealment   (d) With Adaptive Concealment

Fig. 2.19.: MDC adaptive error concealment performance (luma) for *RaceHorses* and *PartyScene.*

Figure 2.19 shows example frames from *RaceHorses* and *PartyScene* without and with concealment. Some CTUs from the *RaceHorses* frame has undergone one description loss, and some have suffered from both description loss. Some CTUs from the *PartyScene* frame have undergone one description loss. However, in both frames, our adaptive method has concealed the losses reasonably well.

In conclusion, our proposed adaptive concealment methods developed for the H.26x MDC bitstreams work well in terms of both PSNR and visual quality. They provide a graceful degradation in performance with packet loss.

3. VPX ERROR RESILIENT VIDEO CODING USING DUPLICATED PREDICTION INFORMATION

As described in Chapter 1, in a typical video coding system, each square block of pixels is predicted from previously decoded set of pixels. Each frame is divided into partitions (and blocks) by the encoder. This prediction can be Intra that uses the same frame or Inter that uses previously coded frame(s) and generally represented in terms of Intra-direction mode or motion vectors (MVs). Prediction error is transform-coded to provide additional information to the prediction signal [13]. A "good" encoder generates partition and prediction signal such that it minimizes the prediction error. Therefore, in case of frame-loss, it is important to recover its prediction signal that mainly consists of partition, mode and motion information.

In this chapter, we propose a VPx-based video coding system that uses duplication of frame-level macroblock (MB) prediction information to provide error resilience [236].

3.1 System Architecture

As shown in Figure 3.1, a video sequence is encoded using a standard VPx encoder with each encoded frame shown in blue. Our proposed system is designed to form an "error resilience packet" (shown in yellow) for a given interval of time. This packet consists of only the prediction information of each frame that was transmitted from the occurrence of last error resilience packet. This packet follows syntax specific to VPx standard. These packets are sent either embedded in the bitstream or over a separate channel. When the VPx decoder receives the bitstream from a lossy network (lost frames in red) it uses the corresponding error resilience packet (when available) to conceal the lost frames to produce a reconstruction signal.
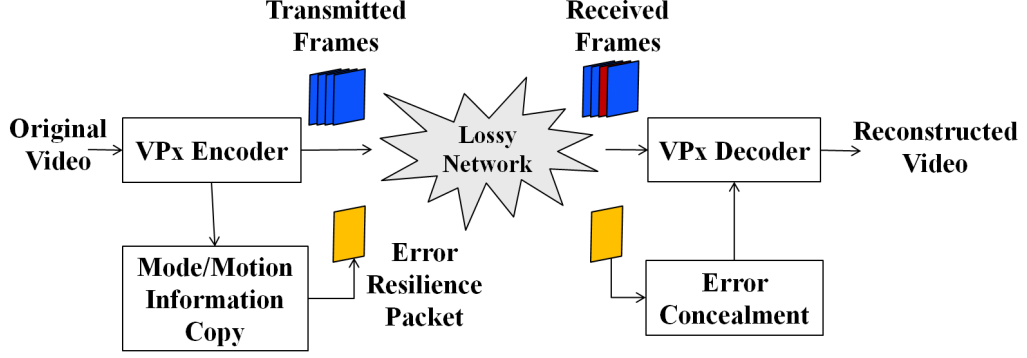
Fig. 3.1.: Our proposed coding architecture

In VPx standard [10], every compressed frame has three or more parts. It begins with an uncompressed data chunk comprising 10 bytes in the case of key (Intra) frames and 3 bytes for Interframes. This is followed by two or more blocks of compressed data known as partitions. The first compressed partition consists of two subsections: (a) Header information that applies to the frame as a whole and (b) Per-MB information specifying how each MB is predicted from the already-reconstructed data that is available to the decompressor. The rest of the partitions contain, for each block, the quantized DCT/WHT coefficients of the residue signal to be added to the predicted block values [10]

In our proposed system, we duplicate, for each frame, the uncompressed data chunk and the first compressed partition e.g. (a) header and (b) per-MB information. We concatenate this information from N frames to form our error resilience packet. This error resilience packet is sent after every N frames. The bitstream produced by our system is not compliant with the current VPx standard.

3.2 Error Concealment

When a packet loss occurs, our proposed method the decoder uses the encoded prediction information obtained using the error resilience packet to reconstruct the prediction signals for that frame. This is different than a conventional VPx decoder

that uses previous frame's encoded or pixel information to estimate the lost frame. For Interframe, the decoder forms motion compensated prediction signal using the motion information: mode, motion vectors and reference frame to reconstruct the Interframe prediction signal. Residue information cannot be recovered because it is not duplicated and sent via the error resilience packet. In case of a lost keyframe, the decoder forms Intra prediction signal for each coded MB. However, recovering a lost keyframe using this method is not very effective because often the keyframe relies on the transform coefficients to reconstruct the initial block.

3.3   Experimental Setup

We modified the VPx software available on the WebM website [16] for our preliminary experiments. We used the VP9 encoding options [11], mainly "codec", "good", "error-resilient", "cpu-used", "target-bitrate", "kf-max-dist" to obtain different encoded bitstreams [237]. Details of these parameters are listed in Table 4.1.

The following are our encoding commands:

```
./vpxenc -w <Width> -h <Height> --i420 --verbose --psnr -o <out.webm>
--codec=vp9 --good --cpu-used =<0/1/2> --end-usage=cbr --fps=<fps>/1
--passes=1 --target-bitrate=<500-15000> --kf-min-dist=0 --kf-max-dist=
<15/25> --error-resilient=1 <in.yuv>
```

We assume one encoded frame is sent per packet. We also assume that the error resilience packet is always loss-free. In our experiments we set "N" to `--kf-max-dist`. We used a Gilbert model as a packet loss simulator [64]. When the packet loss rate is small, burst length is large; and vice versa [235]. We used burst length of 5 for 5% and 10% packet loss rate. The test sequences used for our experiments are listed in Table 3.2.

Parameter	Description	Value
`-w`, `-h`	Spatial dimensions of a frame (width and height) of the video sequence in .yuv format	832×480 or 1280×720
`--fps`	Output frame rate, expressed as a fraction	$30/1$, $50/1$ or $60/1$
`--i420`	Input file uses 4:2:0 subsampling	–
`--good`	`--good` quality and `--cpu-used=0` typically gives quality that is usually very close to and even sometimes better than that obtained with `--best` with the encoder running approximately twice as fast.	–
`--cpu -used`	This sets target cpu utilization $= 100 \times \frac{(16-\text{cpu-used})}{16} \%$	0, 1 or 2
`--codec`	Codec to use VP8 or VP9	`vp9`
`--end -usage`	Rate control mode specifying constant bitrate, variable bitrate or constrained quality	`cbr`
`--target -bitrate`	Target bitrate in kbps	$500 - 15000$
`--error -resilient`	Specifies usage of video conferencing mode	1
`--kf-min -dist`	Minimum keyframe interval (frames)	0
`--kf-max -dist`	Maximum keyframe interval (frames)	15, 25 or 30
`--passes`	Specifies one-pass or two-pass encoding	1
`--psnr`	Shows PSNR in status line	–
`--verbose`	Shows encoder parameters	–

Table 3.1: VPx encoding parameters used in our experiments

Sequence	Spatial Resolution	Frame Rate	Number of frames	`--kf-max-dist`
BasketBallDrill	832×480	50	500	25
RaceHorses	832×480	30	300	15
PartyScene	832×480	50	500	25
KristenAndSara	1280×720	60	600	30
Johney	1280×720	60	600	30

Table 3.2: Test sequences used for our experiments

3.4 Results And Analysis

Figure 3.6 (a) - (e) show the packet loss performance of our proposed method for test sequences. For each test sequence, PSNR vs. data rate is depicted for lossless, 5% and 10% packet loss cases. "VP9-m" indicates our proposed method using the encoding parameter `--cpu-used`, where "m" can take values "0", "1" or "2". Encoded bitstreams for different data rates are obtained using the parameter `--target-bitrate`. Note that, due to the error resilience packets, our bitstreams have a higher data rate (as reported in Figure 3.6 (a) - (e)) than the actual value specified using `--target-bitrate` for each data point. For example, for *KristenAndSara*, using `--target-bitrate=2000` and `--cpu-used=1` actually produced 2387.78 kbits/sec using our proposed method. The additional data rate accounts for an error resilience packet sent after every N frames. Our reported data rate numbers obviously include the additional data rate due to error resilience packets.

For each sequence, our method produces acceptable PSNR values in presence of packet loss. As packet loss increases, our method shows a graceful degradation in performance. When the `--cpu-used` parameter is increased, PSNR performance is also slightly degraded. According to Table 4.1, `--cpu-used=0` means the highest CPU usage among the values we used ("0", "1" and "2"). Therefore, as this number in-

creases, PSNR typically decreases. This is because the amount of CPU used to encode the bitstream becomes smaller as the value of `--cpu-used` is increased, indicating less efforts taken for encoder-controlled operations such as motion estimation.

Figure 3.7 - 3.12 show visual performance of our proposed method using the luma component of some example frames of the test sequences. Each figure contains (a) original frame, (b) decoded frame without any packet loss and (c) decoded frame when it was lost during transmission and concealed at the decoder using our proposed method. As shown in examples from Figure 3.7 - 3.12, each lost frame is successfully concealed in most parts of the frame. For *BasketBallDrill* example shown in Figure 3.7, the areas near moving parts (e.g. the ball, players' hands and legs) contain blocky artifacts. In the example shown in Figure 3.8, 3.9 and 3.10 using *RaceHorses* sequence, the darker moving areas with relatively uniform pixel intensity (e.g. horse) contains blockiness. Figure 3.11 contains small artifacts in the high-texture area (e.g. the hair of the woman on the left) of *KristenAndSara* sequence. An example of *Johney* sequence, as shown in Figure 3.12, contains a small artifact in the moving area (e.g. man's face) of the frame. This is because the decoder only has the mode and motion information of that frame and lacks any pixel-wise residue information.

Figure 3.13 and 3.14 show examples of failure cases we encountered, when only the prediction information is not sufficient to produce an acceptable image quality at the decoder. However, sending pixelwise residue information means a significant increase in the data rate, which can be forbidden considering the bandwidth limitation imposed by the transmission media.

In general, our VPx bitstream-based error resilience method can generally produce an acceptable concealment outcomes both in terms of PSNR and the visual quality in the case of lost packets.
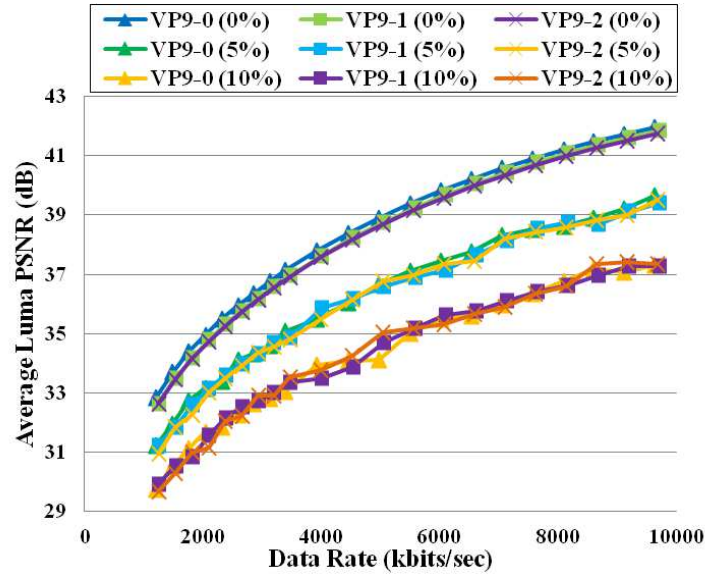
Fig. 3.2.: Packet loss performance for *BasketBallDrill*
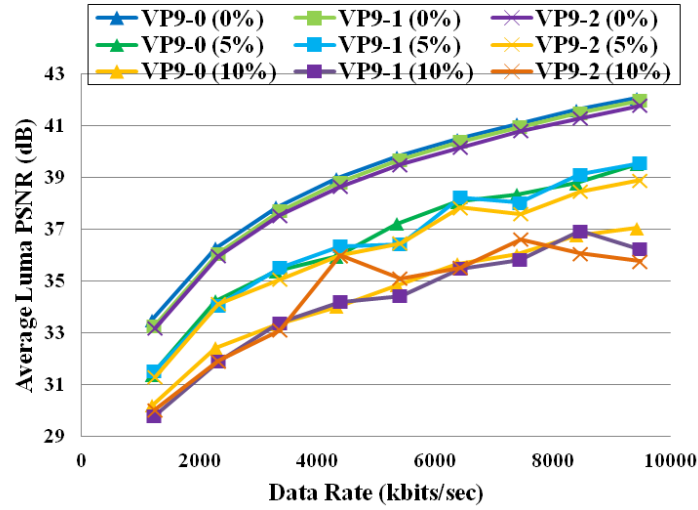
Fig. 3.3.: Packet loss performance for *RaceHorses*
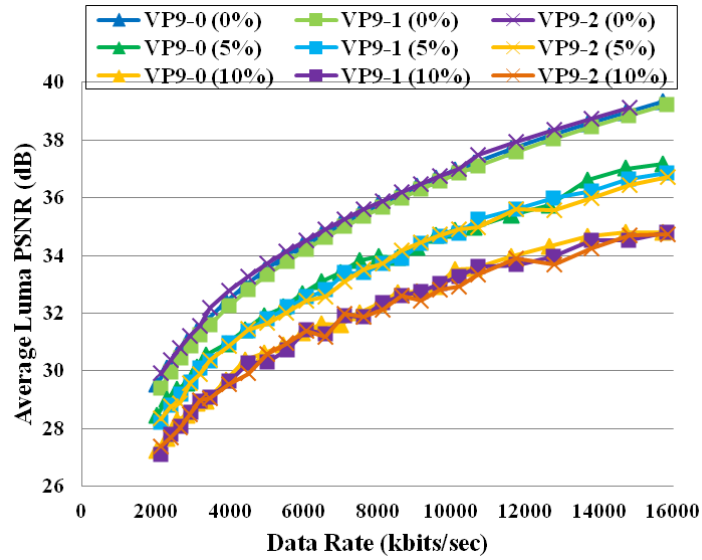
Fig. 3.4.: Packet loss performance for *PartyScene*
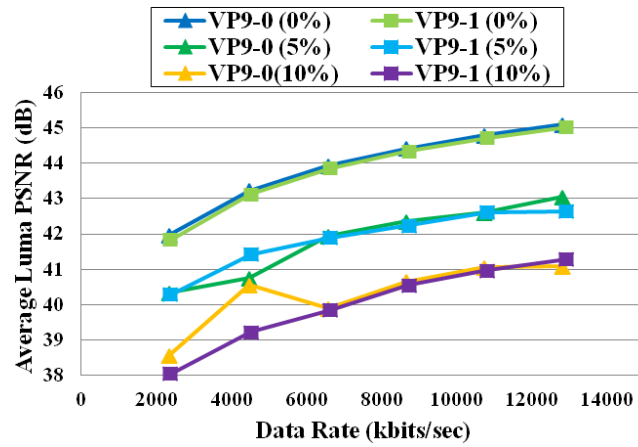
Fig. 3.5.: Packet loss performance for *KristenAndSara*
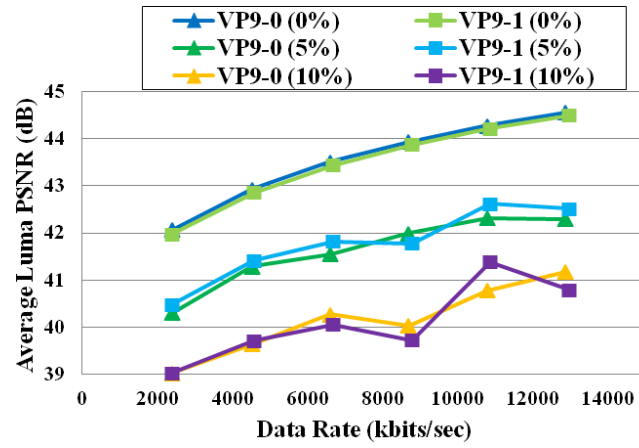
Fig. 3.6.: Packet loss performance for *Johney*

(a) Original   (b) No Loss   (c) Loss Concealed
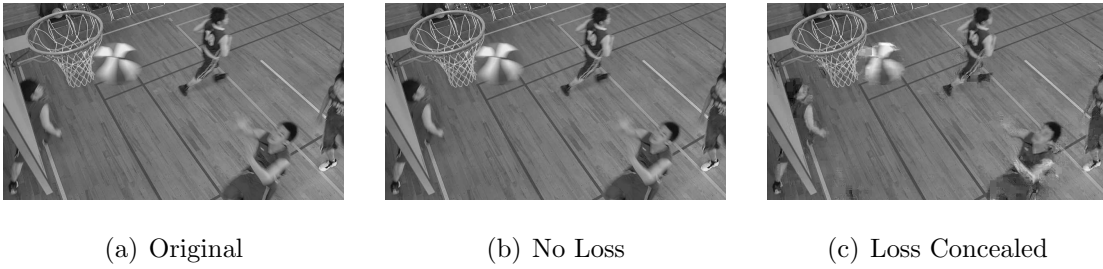
Fig. 3.7.: *BasketBallDrill* sequence (frame no. 284)

(a) Original   (b) No Loss   (c) Loss Concealed

Fig. 3.8.: *RaceHorses* sequence (frame no. 148)

(a) Original   (b) No Loss   (c) Loss Concealed

Fig. 3.9.: *RaceHorses* sequence (frame no. 177)

(a) Original   (b) No Loss   (c) Loss Concealed

Fig. 3.10.: *RaceHorses* sequence (frame no. 280)

(a) Original   (b) No Loss   (c) Loss Concealed

Fig. 3.11.: *KristenAndSara* sequence (frame no. 164)

(a) Original   (b) No Loss   (c) Loss Concealed
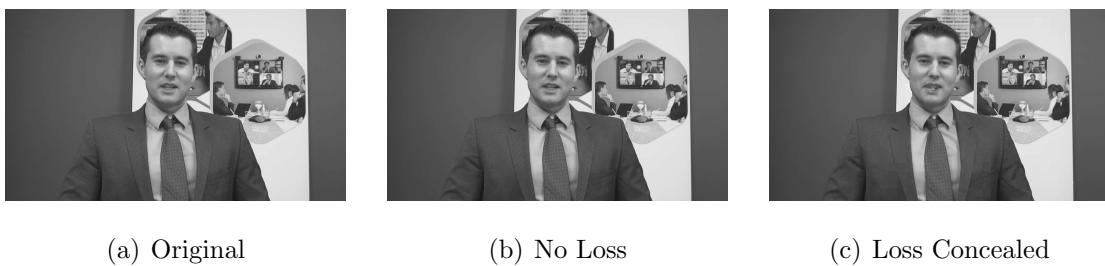
Fig. 3.12.: *Johney* sequence (frame no. 53)

(a) Original   (b) No Loss   (c) Loss Concealed

Fig. 3.13.: *RaceHorses* sequence (frame no. 183)

(a) Original   (b) No Loss   (c) Loss Concealed

Fig. 3.14.: *KristenAndSara* sequence (frame no. 360)

4. JELLY FILLING SEGMENTATION OF BIOLOGICAL STRUCTURES

In this chapter, we describe an iterative 3D segmentation method that we call "jelly filling" [238], for biological images containing "incomplete labeling," a specific problem seen in fluorescent microscopy images. We first discuss our image analysis goal.

4.1 Image Analysis Goal

(a)   (b)   (c)

(d)   (e)   (f)
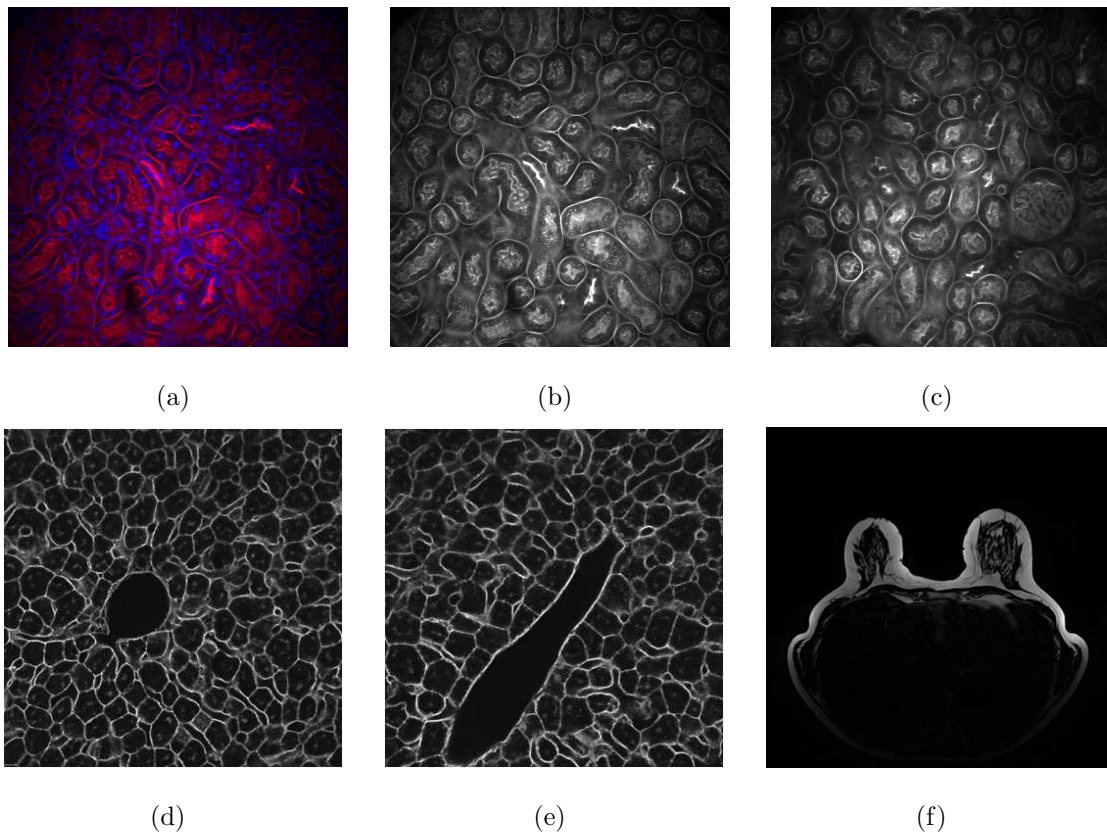
Fig. 4.1.: Examples of our image data containing incomplete labeling

Figure 4.1 shows some examples of our image data. Figure 4.1 (a), (b) and (c) show images taken from a rat kidney in an *in-vivo* experiment during which, images are taken from several hundred focal planes (in the depth dimension) representing a live 3D kidney specimen. The specimen contains proximal tubules with their associated brush borders (or "lumen") such that each tubule connects to a single glomerulus. A tubule and its attached glomerulus is called a nephron. Figure 4.1 (a) consists of data collected using two color channels: red and blue. The separated red channel is shown in Figure 4.1 (b) that represents a cross-section of proximal tubules. The objective of this study is to morphologically characterize single nephron by locating the proximal tubules and glomeruli. Figure 4.1 (c) is a sample red channel image from another kidney specimen. As seen in Figure 4.1 (b) and (c), the fluorescent dye that labeled the tubule boundaries also labeled the brush borders, resulting into a single color channel (the red channel) that represented two biological entities. We call this problem "incomplete labeling." Our image analysis goal is to identify the tubule boundaries separated from the brush borders in 3D, using the images from subsequent focal planes.

Figure 4.1 (d) and (e) are the red color channels of images taken from a rat liver. For these images, the fluorescence of the dye labels cell boundaries and endothelia. Our segmentation goal for this data is to highlight blood vessels and cell-cell junctions, in order to quantitatively characterize the vascular space and hepatocytes.

Figure 4.1 (f) shows an example of a DCE-MRI breast images taken using the TWIST Dixon pulse sequence technique [239]. It is intended that breasts from the images are highlighted and isolated from the body. Another goal is to quantify fat versus fibroglandular tissue inside each breast for quantification purposes.

All of the above types of data, thus contains "incompleteness in labeling" of different biological entities that can be discriminated based on their 3D structural properties. Our proposed approach is designed to provide 3D segmentation of biological images that consists of separable individual entities characterized by closed shapes outlined by their boundaries. Note that, we call the outer encapsulations (i.e. tubule,

cell or breast boundaries) simply "boundaries" and the inside remains (i.e. brush borders, endothelia/vascular space or fibroglandular tissues) "lumen" throughout the description of our method.

An overview of our approach followed by the detailed description is presented next.

4.2   Overview Of Our Approach

Figure 6.3 shows our proposed approach.
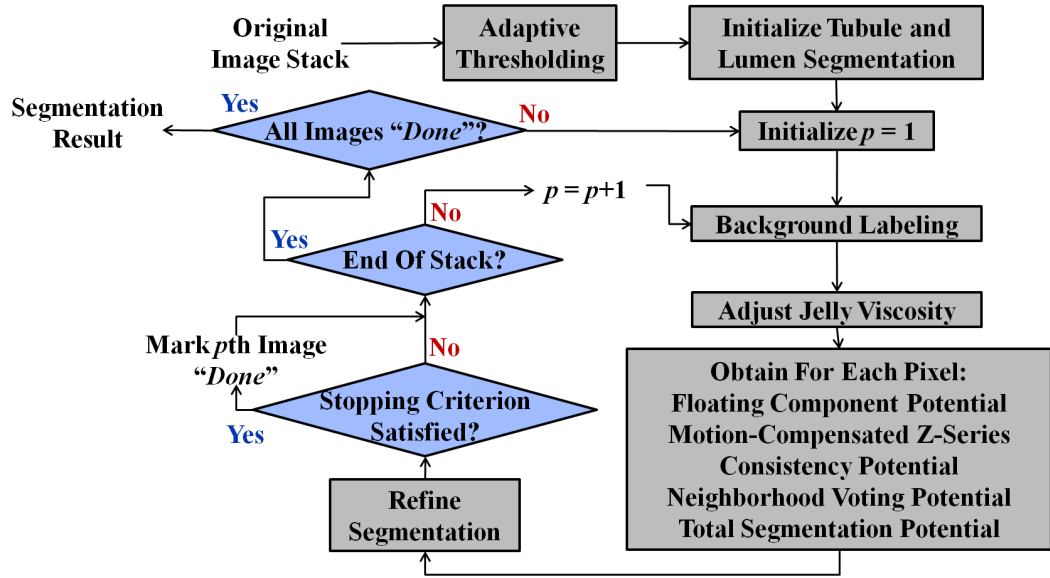
Fig. 4.2.: Proposed approach: Flowchart

As indicated above, the microscopy images (acquired from rat kidneys and livers) consist of two-channels (red and blue) that reflect the fluorescence of the two dyes added to the tissue. We first separate the red color channel to obtain grayscale images $\mathcal{I}_{z_p,c_1}$, where c_1 represents the red-channel data. Adaptive thresholding is effective in dealing with radial intensity drop in an image because a local statistic is used as a threshold to segment each pixel as foreground/background. So, we first use adaptive thresholding on $\mathcal{I}_{z_p,c_1}$ to produce a binary images.To be able to separate boundaries

and lumen, we use an iterative segmentation approach that aims to detect "floating" elements inside the disjoint regions corresponding to a biological encapsulations (such as tubules, a liver cells or breasts). Intuitively, our method is based on filling a disjoint region of an image with a "jelly-like" fluid with a unique label. This helps in the detection of components that are floating within a "labeled-jelly". The "viscosity" of the jelly can be controlled using simple morphological operations such as erosion, dilation using a specific structural element [164, 240].

The images generally have a lower sampling rate in the z-direction and lumen cannot be typically separated as an entirely separate 3D component from the tubule boundaries. Therefore, instead of using a 3D component analysis, we use a 2D component analysis to detect a part of lumen as a floating component in an image and then use it to "correct" the segmented images from the adjacent focal planes, consequently improving the 3D segmentation. We also use a 2D neighborhood voting potential to consider the effect of neighboring segmented pixels. This is conceptually based on region-growing techniques such as the one mentioned in [182, 183]. Each pixel is segmented/classified as either belonging to "boundary" or "lumen" using a potential function that considers the influence of these factors. This process is repeated until the relative change (expressed as a percentage) in pixel classification decreases below a fixed level for each image.

Details of the proposed segmentation method are provided next.

4.3   Jelly Filling Segmentation

Recall that $\mathcal{I}_{z_p, c_1}$ denotes the pth original red channel image. In this chapter, we use only the red channel, hence the grayscale images representing the red channel of the original images will be called $\mathcal{I}_{z_p}$ after dropping the color suffix in the notation throughout this chapter. So, $\mathcal{I}_{z_p}$ are grayscale input images for our segmentation method. Let $\mathcal{S}_{Th, z_p}$ denote the binary images after adaptive thresholding. The iterative segmentation process begins with an initial configuration of boundaries and

lumen denoted by $\psi_{B,z_p}^{(i)}$ and $\psi_{L,z_p}^{(i)}$, respectively. Let $\psi_{B,z_p}^{(k)}$ and $\psi_{L,z_p}^{(k)}$ denote respectively, the configuration of "boundaries" and "lumen", obtained from $\mathcal{I}_{z_p}$, after the kth ($k = 1, 2, 3, ...$) iteration. The final segmented configurations of $\mathcal{I}_{z_p}$ are denoted by $\psi_{B,z_p}^{(f)}$ and $\psi_{L,z_p}^{(f)}$. A pixel from an image is denoted by s. Now we describe each step of our proposed jelly filling segmentation.

Adaptive Thresholding: Our method employs initially an adaptive thresholding scheme that uses 3D neighborhood information. The main objective of this step is to separate the foreground that represents the presence of a biological structure. In particular, let the $w_1 \times w_2 \times w_3$ 3D window (Ω_{Th}) centered at pixel s and let $\tau_{z_p}(s)$ be the mean pixel intensity of the neighborhood Ω_{Th}. The local mean $\tau_{z_p}(s)$ is then used as the corresponding thresholding value for s as indicated by Eq. 5.1 below, where $\mathcal{I}_{z_p}(s)$ is used to denote the intensity of the pixel at location s:

$$\mathcal{S}_{Th,z_p}(s) = \begin{cases} 1 & \text{if } \mathcal{I}_{z_p}(s) \geq \tau_{z_p}(s) \\ 0 & \text{if } \mathcal{I}_{z_p}(s) < \tau_{z_p}(s) \end{cases} \tag{4.1}$$

The outcome of this step is used as an initial segmentation. In particular, for the pth image we set $\psi_{B,z_p}^{(0)} = \mathcal{S}_{Th,z_p}$ and $\psi_{L,z_p}^{(0)} \equiv 0$, that is all pixels that exceed their corresponding thresholds are initially labeled as belonging to boundaries. This also contains the pixels belonging to lumen, which are separated from the boundaries in subsequent steps of the iterative framework.

Background Labeling: This step separates the disjoint background regions from the output of the adaptive thresholding and assigns them with different labels. Because of the underlying biological structure, the background is composed of regions belonging to different biological compartments that can be separated into disjoint sets of pixels. Intuitively, this can be viewed as filling these disjoint compartments (or encapsulated regions) with a "jelly-like" fluid. The viscosity of this fluid reflects into the pixel neighborhood used for finding disjoint regions of the background.

Consider a boundary configuration $\psi_{B,z_p}^{(k-1)}$ obtained during the $(k-1)$th iteration that is to be used in the kth iteration. Let $\Lambda_{z_p}^{(k)}$ denote the background image at the kth iteration, and which is defined as:

$$\Lambda_{z_p}^{(k)} = \{s | \psi_{B,z_p}^{(k-1)}(s) = 0\}. \tag{4.2}$$

Assume that there exist M disjoint background regions in $\Lambda_{z_p}^{(k)}$. Each such disjoint background region is labeled as $\lambda_{m,z_p}^{(k)}$ such that

$$\Lambda_{z_p}^{(k)} = \bigcup_{m=1,2,\dots M} \lambda_{m,z_p}^{(k)} \tag{4.3}$$

and each $\lambda_{m,z_p}^{(k)}$ can be considered to identify a group of pixels belonging to the biological entity enclosed by the boundary. Each $\lambda_{m,z_p}^{(k)}$ is obtained by applying connected component labeling using a 4-point neighborhood to $\Lambda_{z_p}^{(k)}$ [156].

Segmentation Based on a Potential Function $P(\cdot)$: The goal is to separate the pixels belonging to lumen from those of boundaries. We consider three factors that influence this separation:

- A pixel belonging to a "floating" component is likely to be segmented as lumen.

- It is important to maintain structural consistency in the z-direction. A motion-compensated segmentation correction in the z-direction is developed to model this factor.

- The effect of segmentation of neighboring pixels of an image also needs to be considered to determine whether a pixel belongs to boundaries or lumen.

We consider influence of these factors in terms of values assigned to each pixel, obtained using potential functions each of which corresponds to a factor. The total potential is the summation of these individual potentials such that the sign of this summation determines whether a pixel is classified as boundary or lumen.

For each pth image I_{z_p}, $\psi_{B,z_p}^{(k)}$ and $\psi_{L,z_p}^{(k)}$ are updated based on $\mathcal{S}_{Th,z_p}$ and a potential function $P(\cdot)$ as follows:

Pixels classified as background pixels are not considered to be a part of either boundaries or lumen. Thus,

$$\psi_{B,z_p}^{(k)}(s) = \psi_{L,z_p}^{(k)}(s) = 0 \text{ for all } s \in \Lambda_{z_p}^{(k)} \tag{4.4}$$

Next, $P_{z_p}^{(k)}(s)$ is obtained for only the pixels s where $\mathcal{S}_{Th,z_p}(s) = 1$. Based on the sign of $P(s)$, each pixel s is assigned to be either a member of boundaries $\psi_{B,z_p}^{(k)}$ or lumen $\psi_{L,z_p}^{(k)}$ according to:

$$\psi_{B,z_p}^{(k)}(s) = \begin{cases} 1 & \text{if } s \in \mathcal{S}_{Th,z_p} \text{ and } P_{z_p}^{(k)}(s) \leq 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.5}$$

and

$$\psi_{L,z_p}^{(k)}(s) = \begin{cases} 1 & \text{if } s \in \mathcal{S}_{Th,z_p} \text{ and } P_{z_p}^{(k)}(s) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.6}$$

Now, $P_{z_p}^{(k)}(s)$ is the sum of three components, $P_{F,z_p}^{(k)}(s))$: floating component potential, $P_{M,z_p}^{(k)}(s)$: motion compensated z-series consistency potential and $P_{N,z_p}^{(k)}(s)$: neighborhood voting potential, that is:

$$P_{z_p}^{(k)}(s) = P_{F,z_p}^{(k)}(s) + P_{M,z_p}^{(k)}(s) + P_{N,z_p}^{(k)}(s) \tag{4.7}$$

Note that, $P(\cdot)$, $P_F(\cdot)$, $P_M(\cdot)$, and $P_N(\cdot)$ are defined only for all pixels s such that $\mathcal{S}_{Th,z_p}(s) = 1$. Henceforth, we will assume $\mathcal{S}_{Th,z_p}(s) = 1$ for all future references to s, unless specified otherwise.

Floating Component Potential $P_F(\cdot)$: This potential represents identifying a component that is "floating" in one background region and labeling it as lumen. A floating component is defined as a connected component with only one label surrounded by the background. To obtain the floating component potential during iteration k we consider $\psi_{B,z_p}^{(k-1)}$, the configuration of boundaries from the $(k-1)$th iteration, and $\lambda_{m,z_p}^{(k)}$ for $m = 1, 2, ...M$, the disjoint background regions, as described earlier. Let C denote the set of all connected components in $\psi_{B,z_p}^{(k-1)}$ obtained using

4-pixel neighborhood connectivity. Each $c \in C$ is a set of pixels belonging to a single connected component. The outer boundary of each c, denoted by b_c, is next found by selecting the boundary pixels of the morphological dilation of c using the same structural element used to account for the viscosity of the jelly. Also let C_F ($C_F \subseteq C$) denote the set of all floating components, that is

$$C_f = \{c | b_c \subseteq \lambda_{l,z_p}^{(k)}, \text{ for some } l \in \{1, 2, ...M\}\} \tag{4.8}$$

Now, we assign floating point potential (P_F) to each pixel s as:

$$P_{F,z_p}^{(k)}(s) = \begin{cases} \alpha_f & \text{if } s \in C_F \\ -\alpha_f & \text{otherwise} \end{cases} \tag{4.9}$$

where α_f is a positive constant, whose value is chosen in such a way so as to influence the labeling of "floating" components as lumen. The value of this constant is chosen to be a small positive number ($0 < \alpha_f \leq 2$) to indicate that the "floating" component is indeed a part of lumen.

Motion-Compensated Z-Series Consistency Potential $P_M(\cdot)$: While iteratively processing the images from successive focal planes, it is important to maintain structural continuity in all directions. This can be accomplished if the segmentation of neighboring images along the z-direction influence the segmentation of current image. To do this, we use w_Z boundary configurations in either direction along the z-axis (total of $2 \times w_Z$ images) from the previous i.e. $k-1$th iteration: $\psi_{B,z_{p+n}}^{(k-1)}$, $n \in \{-w_Z, ..., -1, 1, ..., w_Z\}$. Each of these configurations are first motion compensated with respect to the pth image ($\psi_{B,z_p}^{(k-1)}$) to counter any movement of the specimen while imaging *in-vivo* or other imaging effects that vary from one focal plane to another.

We do motion compensation for each $\psi_{B,z_{p+n}}^{(k-1)}$ individually, using only $\psi_{B,z_p}^{(k-1)}$ as the reference image. Let $\psi_{B,z_{p+n}}^{(k-1)\{MC\}}$ be the motion-compensated boundary configurations derived from the corresponding original $\psi_{B,z_{p+n}}^{(k-1)}$ for $n \in \{-w_Z, ..., -1, 1, ..., w_Z\}$ by

selecting the minimum sum of absolute difference (SAD) translational motion among the motion candidates from a square window Ω_{MC} a $(\pm w_{MC} \times \pm w_{MC})$ centered around the origin.

First, (m_x, m_y): the translational motion in the x-and the y-direction is obtained using the minimum-SAD,

$$(m_x, m_y) = \underset{(m_x^c, m_y^c) \in \Omega_{MC}}{\arg\min} \sum_{\text{All Pixels}} |\psi_{B,z_{p+n}}^{(k-1)}(s_x + m_x^c, s_y + m_y^c) - \psi_{B,z_p}^{(k-1)}(s_x, s_y)| \quad (4.10)$$

where (s_x, s_y) represents the x-y indices of pixel s.

Then, we compute the corresponding motion compensated boundary and lumen configurations: $\psi_{B,z_{p+n}}^{(k-1)\{MC\}}$ and $\psi_{L,z_{p+n}}^{(k-1)\{MC\}}$ respectively.

$$\psi_{B,z_{p+n}}^{(k-1)\{MC\}}(s_x, s_y) = \psi_{B,z_{p+n}}^{(k-1)}(s_x + m_x, s_y + m_y) \quad (4.11)$$

$$\psi_{L,z_{p+n}}^{(k-1)\{MC\}}(s_x, s_y) = \psi_{L,z_{p+n}}^{(k-1)}(s_x + m_x, s_y + m_y) \quad (4.12)$$

Now, we employ a one dimensional (1D) Gaussian function of length $(2w_z + 1)$: $f_z(n) = (1 - \delta(n)) \cdot e^{-\frac{|n|^2}{2^2}}$, $n = -w_z, .., 0, .., w_z$ and define $P_z(s)$ to be:

$$P_{M,z_p}^{(k)}(s) = \sum_{n=-w_z}^{w_z} \{\psi_{L,z_{p+n}}^{(k-1)\{MC\}}(s) - \alpha_z \cdot \psi_{B,z_{p+n}}^{(k-1)\{MC\}}(s)\} \cdot f_z(n), \quad (4.13)$$

where α_z is a constant whose value is set to provide suitable z-series consistency for boundary and lumen detection.

Note that the objective is to provide a stable final configuration that undergoes practically negligible changes in boundary and lumen configurations after fulfilling the stopping criterion. Recall that we chose the initial configuration "all boundary." Therefore, it is important to set $\alpha_z < 1$ to avoid incorrectly converging to an intermediate configuration close to the initial one. Setting $\alpha_z \approx 0$ may cause the method to go to an undesired "all lumen" configuration. A desired range of α_z was found experimentally to be $0.1 \leq \alpha_z \leq 0.9$. In general, a low value of α_z in this range leads to fast convergence.

Neighborhood Voting Potential $P_n(\cdot)$: To clearly define the separation between

boundary and lumen segments, a 2D Gaussian voting function is used. It is conceptually similar to the voting-based distributing function used in the active mask framework [182]. We define $P_n(s)$ to be:

$$P_{N,z_p}^{(k)}(s) = \{(\psi_{L,z_p}^{(k-1)} - \psi_{B,z_p}^{(k-1)}) * f_n\}(s) \tag{4.14}$$

where $*$ represents 2D convolution and f_n is a truncated 2D Gaussian function of size $(2w_n + 1) \times (2w_n + 1)$:

$$f_n(x,y) = \frac{1}{F_{w,n}} \cdot e^{-\frac{(|x|^2+|y|^2)}{2^2}} \quad, x, y = -w_n, .., 0, .., w_n \tag{4.15}$$

where,

$$F_{w,n} = \sum_{i_n=-w_n}^{w_n} \sum_{j_n=-w_n}^{w_n} e^{-\frac{(|i_n|^2+|j_n|^2)}{2^2}}.$$

Morphological Opening: In order to adjust the viscousity of the jelly or the background $(\Lambda_{z_p}^{(k)})$, we use morphological opening to the original background image using a circular structuring element [240]. We use a circular disk of radius $r = 1$ pixel as the strucural element for all of our images.

Clean-Up: Tiny clusters of pixels i.e. connected components can be safely removed to preserve a high-level structural continuity. We call this operation clean-up, in which the values of pixels belonging to components smaller than γ pixels are assigned to 0. γ represents the number of connected pixels that can be safely eliminated from the image and it is typically very small ($10 \leq \gamma \leq 100$) as compared to total number of pixels in an image.

Stopping Criterion: As stated above we use percentage change in number of boundary pixels as the stopping criterion. In particular, we define

$$\Delta_{z_p}^{(k)} = \frac{\text{Diff}(\psi_{B,z_p}^{(k)}, \psi_{B,z_p}^{(k-1)})}{\text{Total pixels}} \times 100 \tag{4.16}$$

where Diff indicates the number of changed pixels, that is $\text{Diff}(A, B) = \sum_{Allpixels} (A$ XOR $B)$. The stopping criterion is then:

$$\Upsilon : \text{Is } \Delta_{z_p}^{(k)} < \epsilon \text{ for every } p? \tag{4.17}$$

Typically, $\epsilon = 1$ or 0.1 works well for our data with practically no change in segmented pixels during iterations after the stopping criterion is met.

The steps of our proposed segmentation method are outlined below.

Jelly Filling Segmentation

Require: Input images $\mathcal{I}_{z_p}$, $p = 1, 2, ...P$

Do **Adaptive Thresholding** to $\mathcal{I}_{z_p}$ to obtain $\mathcal{S}_{Th,z_p}$ for $p = 1, 2, ...P$

Initialize: $\psi_{B,z_p}^{(i)} = \psi_{B,z_p}^{(0)} = S_{Th,z_p}$, $\psi_{L,z_p}^{(i)} = \psi_{L,z_p}^{(k)} \equiv 0$ for $p = 1, 2, ...P$

Initialize $k = 0$ to begin the iterative process

while (All images are not *done*) **do**

  for Each pth image **do**

    Do **Morphological Opening** of the background $\Lambda_{z_p}^{(k)}$ to account for the viscosity of the "jelly"

    Clean-up: Remove small components of $\Lambda_{z_p}^{(k)}$ ($< \gamma$ pixels)

    Do **Background Labeling** using 4-pixel neighborhood

    for Each pixel s such that $\mathcal{S}_{Th,z_p}(s) = 1$ **do**

      Obtain **Floating Component Potential** $P_{F,z_p}^{(k)}(s)$ using α_f

      Obtain **Motion-Compensated Z-Series Consistency Potential** $P_{M,z_p}^{(k)}(s)$ using α_z, w_z, w_{MC}

      Obtain **Neighborhood Voting Potential** $P_{N,z_p}^{(k)}(s)$ using w_n

      Obtain **Potential Function** $P_{z_p}^{(k)}(s)$ using $P_{F,z_p}^{(k)}(s)$, $P_{M,z_p}^{(k)}(s)$ and $P_{N,z_p}^{(k)}(s)$

      Do segmentation to get $\psi_{B,z_p}^{(k)}(s)$ and $\psi_{L,z_p}^{(k)}(s)$

    Clean-up: Remove small components of $\psi_{B,z_p}^{(k)}$ ($< \gamma$ pixels)

    Compute the change in pixels $\Delta_{z_p}^{(k)}$

    if Stopping Criterion Υ is satisfied:$\Delta_{z_p}^{(k)} < \epsilon$ **then**

      Declare pth image is *done*

    Increment k to go to the next iteration

  for Each pth image **do**

    Assign $\psi_{B,z_p}^{(f)}$ and $\psi_{L,z_p}^{(f)}$ as boundary and lumen segmentation configurations obtained in the last iteration

---

4.4 Experimental Results

We implemented jelly filling segmentation using MATLAB. An ImageJ [192] plugin is currently under development.

Image Data: For our experiments, we used three types of data: images from kidney (K) and liver (L), and mammograms (M).

Data *K-I*, *K-II*, *K-III* and *K-IV* each contained 8-bit, 3 color channels, images (512×512 pixel dimensions) of rat kidney specimen obtained using fluorescence microscopy, containing 512, 23, 41 and 23 images respectively [1]. Images from *K-I* were labeled with TexasRed-phalloidin and that from *K-II*, *K-III* and *K-IV* were labeled with Alexa488-phalloidin. The fluorescence of phalloidin (which labels filamentous actin) labeled two structures in the tissue, the basement membrane of the tubules and the brush border (or lumen) of the proximal tubules.

Data *L-I*, *L-II*, *L-III*, *L-IV*, *L-V*, *L-VI* and *L-VII* each contained 8-bit, 3 color channels, images (512×512 pixel dimensions) of rat liver specimen obtained using fluorescence microscopy. *L-I* consists of 36 images and *L-II*, *L-III*, *L-IV*, *L-V*, *L-VI*, *L-VII* are single images [2]. The liver samples are labeled with a fluorescent tomato lectin, which labels cell boundaries and endothelia.

The third type of data is composed of DCE-MRI breast images that use the TWIST Dixon pulse sequence technique [239]. This data consisted of 4 sets (*M-I*, *M-II*, *M-III*, *M-IV*) of 128 grayscale mammograms of 512×512 pixel dimensions [3].

Parameter Selection: For an iterative process, it is important to address its convergence. In particular, the parameters α_f and α_z should be selected such that a right

---

[1] *K-I* was provided by Malgorzata Kamocka of Indiana University and was collected at the Indiana Center for Biological Microscopy. *K-II*, *K-III* and *K-IV* were provided by Tarek Ashkar of the Indiana University Division of Nephrology.

[2] The liver data was provided by Sherry Clendenon and James Sluka of the Biocomplexity Institute, Indiana University at Bloomington.

[3] The mammography data was provided by Yuan Le, Randall Kroeker, Hal Kipfer, and Chen Lin and was collected at the Department of Radiology and Imaging Science, Indiana University.

balance among the potentials is maintained. Although in our work, we do not discuss theoretical convergence, our experiments indicated a stable convergence for a range of parameter values without the need of fine-tuning which may give even a better performance than reported in this work. We used the same set of parameters for our experiments with all images, as summarized in Table 4.1 in which each parameter is listed with its description, the value used in experiments and a suggested reference range.

Parameter	Description	Value	Ref. Range
w_1, w_2, w_3	Thresholding window	15, 15, 3	–
α_f	Floating influence	1	$0 < \alpha_f \leq 2$
w_z	Z-series window	2	$1 \leq w_z \leq 5$
w_{MC}	Motion-search window	5	$1 \leq w_z \leq 10$
α_z	Z-series influence	0.25	$0.1 \leq \alpha_z \leq 0.9$
w_n	Neighborhood window	2	$1 \leq w_n \leq 5$
γ	Clean-up threshold	50	$10 \leq \gamma \leq 100$
ϵ	Stopping criterion	0.1	Typically 1/0.1

Table 4.1: Parameters used for our experiments

Illustration:

As illustrated in Figure 4.3 using *K-I* data, our iterative segmentation process begins with initial configurations ($k = 0$) for the $p = 112$th image: $\psi_{B,z_{112}}^{(i)} = S_{Th,z_{112}}$, $\psi_{L,z_{112}}^{(i)} \equiv 0$, where all pixels are segmented as boundaries. For subsequent iterations $k = 1, 2, ...$, intermediate configurations $\psi_{B,z_{112}}^{(k)}$, $\psi_{L,z_{112}}^{(k)}$ are generated using the three potentials: $P_{F,z_{112}}^{(k)}$, $P_{M,z_{112}}^{(k)}$ and $P_{N,z_{112}}^{(k)}$, until the stopping criterion (Υ) is satisfied. In the example shown, this occurs at $k = 24$, leading to the final segmentation results $\psi_{B,z_{112}}^{(f)} = \psi_{B,z_{112}}^{(24)}$ and $\psi_{L,z_{112}}^{(f)} = \psi_{L,z_{112}}^{(24)}$.

(a) Original (b) $k = 0$ (c) $k = 1$

(d) $k = 2$ (e) $k = 4$ (f) $k = 6$

(g) $k = 12$ (h) $k = 18$ (i) $k = 24$
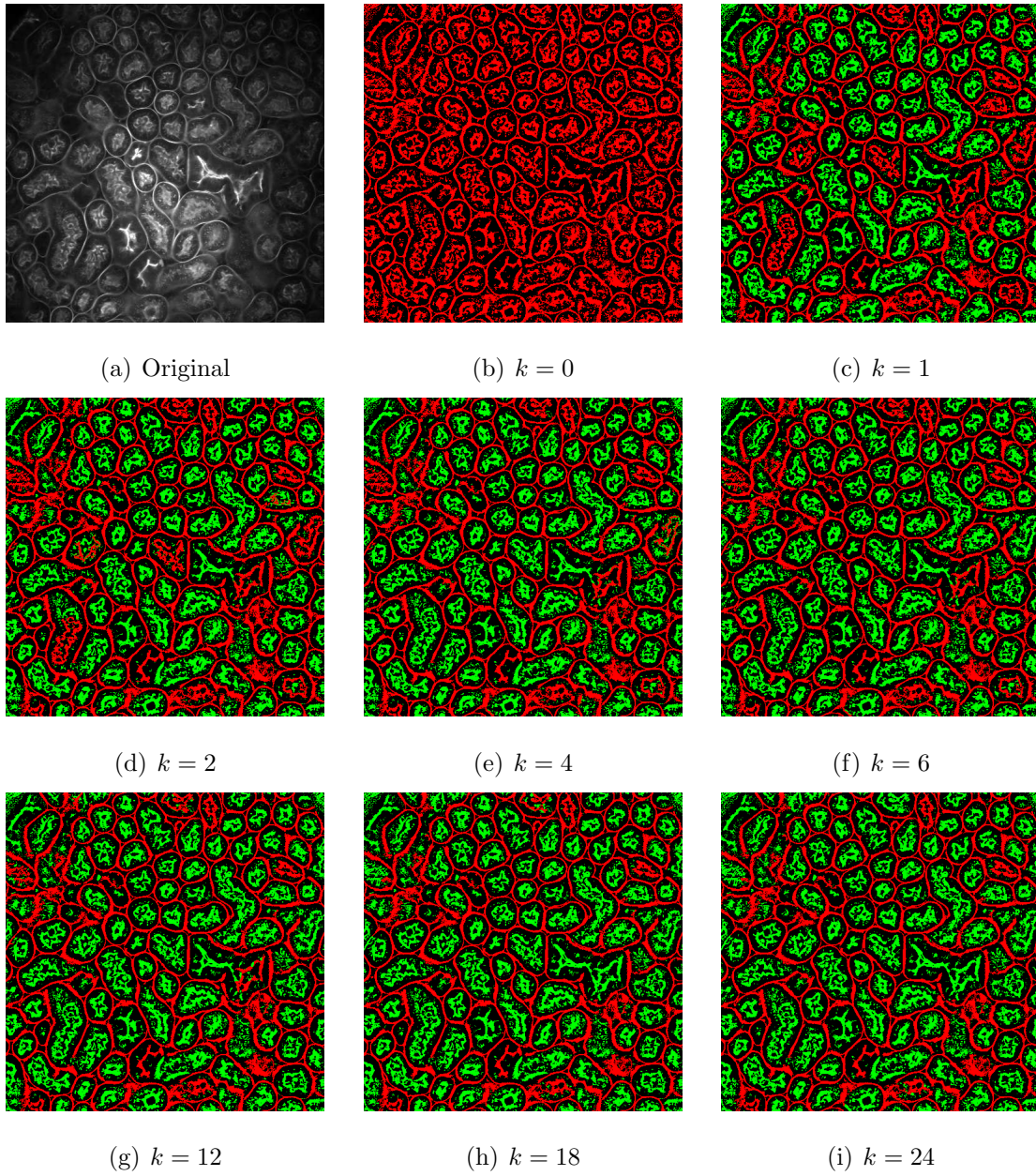
Fig. 4.3.: Illustration with iterations of our proposed method using K-I: starting from $k = 0$ (Initialization) to 24 (Final) at which the stopping criterion is satisfied, red: boundaries, green: lumen.

Segmentation Results: Figure 4.4 shows the results of *K-I*. Most tubule boundaries with their associated lumen are successfully segmented with all necessary details preserved. Note that, the result in Figure 4.4(f) also consists of a glomerulus, another biological entity that is important for the morphological characterization of a nephron of the kidney. The segmentation results obtained by our method significantly enhance the ability to visually identify individual contiguous tubules. There are a few missing tubules especially near the borders of the image and a few falsely detected tubules that should have been segmented/classified as lumen. The main reasons are significant blur and/or low illumination at those places.
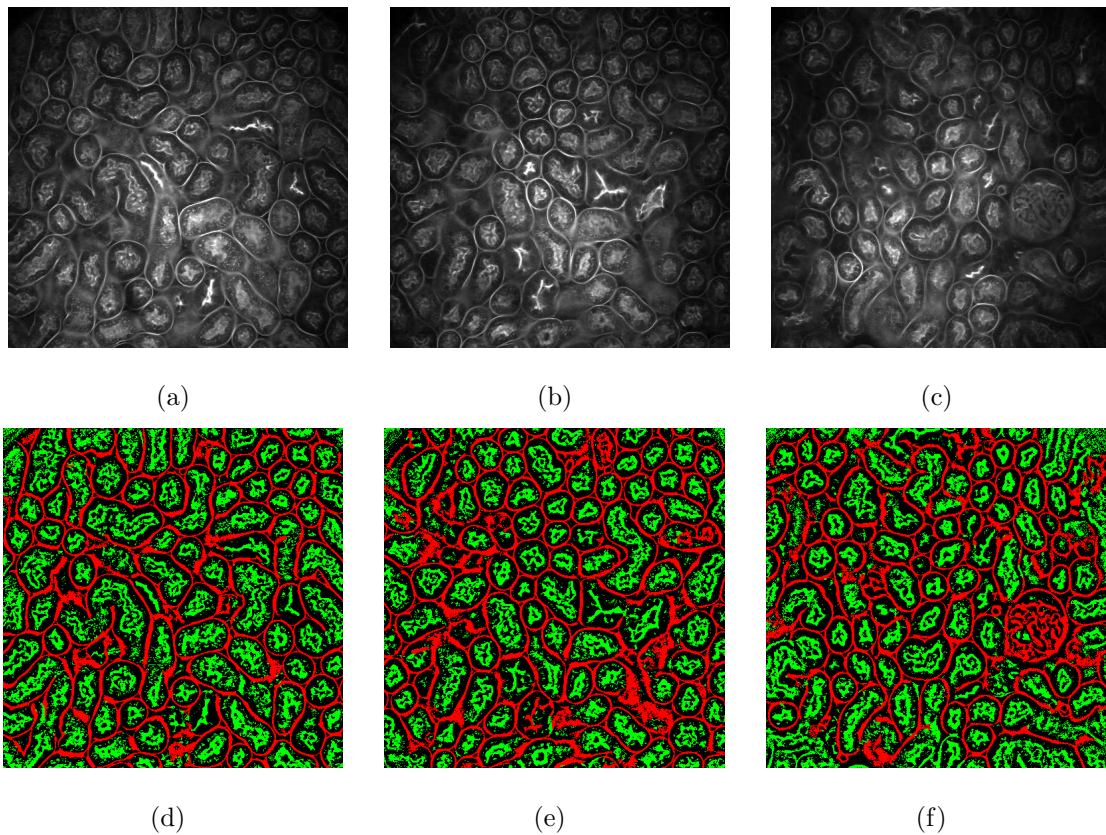
Fig. 4.4.: Segmentation results (for *K-I*): top row- original images, bottom row-boundaries (red) and lumen (green)

Figure 4.5 (a) and (b) show images from *K-II*, *K-III* and their corresponding segmentation results are shown in Figure 4.5 (d) and (e), respectively. At many places in the original images, ring-like lumen can be observed near to the center of an image, where the lumen shape closely resembles to the boundary of a circular tubule. Many lumen regions are considerably brighter than the tubule boundaries enclosing them. The segmentation results show most tubule boundaries with their lumen quantities segmented successfully. Some tubules are observed to contain small biological mass attached to their walls. This is segmented as a part of tubule boundary with most details preserved. Some lumen areas are wrongly classified as tubules mainly due to high relative pixel brightness as compared with their boundaries.

(a)   (b)   (c)

(d)   (e)   (f)
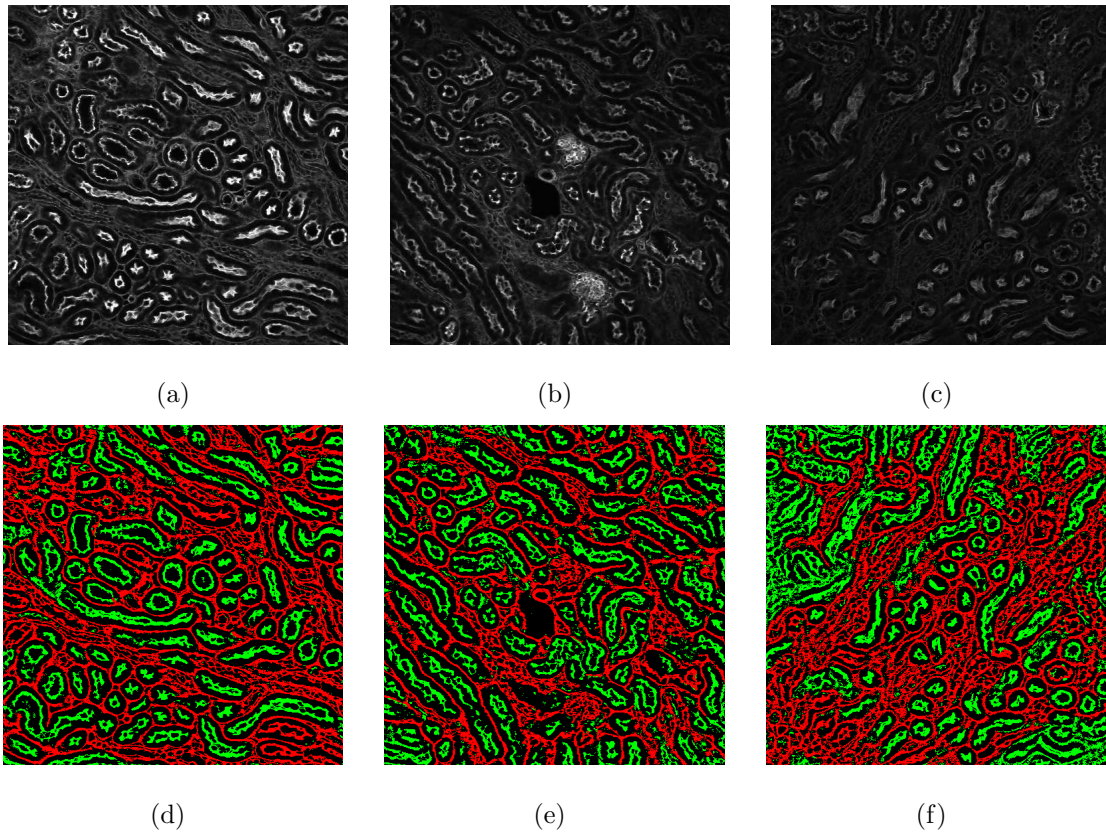
Fig. 4.5.: Segmentation results (for *K-II*, *K-III* and *K-IV*): top row- original images, bottom row- boundaries (red) and lumen (green)

As shown with an example image in Figure 4.5 (c), *K-IV* is more challenging because of very low pixel intensities. Most tubule boundaries are not clearly observable visually. As shown in Figure 4.5 (f), our method has still produced an acceptable segmentation results that may be difficult to obtain even by a human observer.

Objective evaluation of our method proves to be difficult because of the lack of ground truth data, for which the true shape and position of each object in the volume is known [175]. In fact, ground truth is impossible to obtain in fluorescent microscopy, since both the shape and position of an object are fluid in living animals, and are inevitably altered in the process of isolating and fixing tissues. To obtain results from an expert clinician even for a single 3D volume becomes significantly difficult and tedious considering practical limitations in accurately rendering 3D data on 2D displays and requesting the expert to manually segment them.

Yet, we hand-segmented a few images and have the segmentation verified by expert clinician/biologists. We used the hand-segmentation for visual comparison and also as the ground truth to get accuracy, *Type-I* and *Type-II* errors for each method in the context of segmenting boundaries. To be fair to other methods that can segment only one physical quantity in an image, we did not consider lumen as a segmented quantity, but counted as a part of background. Accuracy is obtained as the ratio of number of correctly segmented tubule boundaries and background pixels to the total number of pixels. *Type-I* error is computed as the ratio of number of background pixels falsely detected as tubule boundaries (false detection) to the total number of pixels. *Type-II* error is computed as the ratio of number of tubule boundaries pixels falsely detected as background (missed detection) to the total number of pixels.

Method	Accuracy	*Type-I* error	*Type-II* error
Jelly filling (proposed method)	87.3%	7.1%	5.7%

Table 4.2: Average performance of our proposed method for images of *K-I*.

Table 4.2 summarizes the performance of our method in terms of % accuracy, *Type-I* and *Type-II* error numbers that represent the averages taken over 40 images. We observe that the average accuracy is acceptable with low *Type-I* and *Type-II* errors.

Next, we consider the boundary segmentation accuracy at various iterations of our proposed method for the same 40 images. Figure 4.6 provides a plot of % accuracy of our proposed method at successive iterations as the method produces a final segmentation outcome, for a set of *K-I* images.
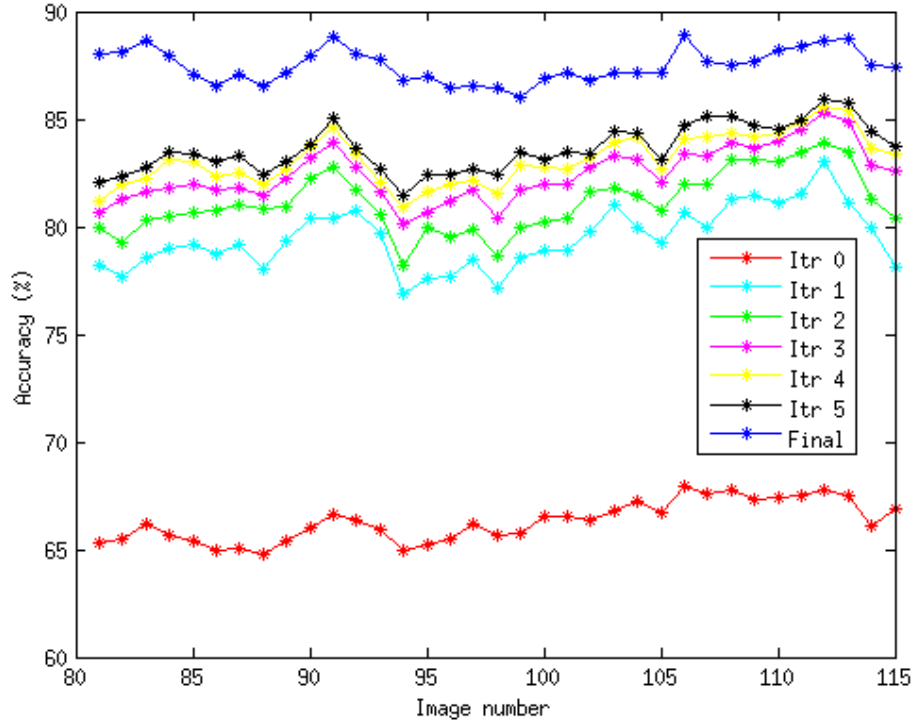
Fig. 4.6.: Percent accuracy of our proposed method at various iterations

Each color represents a specific iteration of our jelly filling method. `Itr 0` has the lowest accuracy $(60-70\%)$, since it was the initial configuration of boundaries same as the result of adaptive thresholding. Recall that it is an "all-boundary" configuration without any pixel labeled as lumen. This configuration is similar to the

results obtained using a typical segmentation method that can discern only biological entity based on the pixel intensities. There is a significant increase in accuracy from `Itr 0` to `Itr 1` because the first jelly filling iteration has detected many "floating" components that are removed from boundary configuration in the 1st iteration. Is can be observed that the accuracy further increases quite uniformly in the subsequent iterations: `Itr 2`, `Itr 3`, `Itr 4` and `Itr 5`. For later iterations, accuracy increases in smaller steps (not shown in the Figure) to satisfy our stopping criterion in the final iteration. The accuracy of the final results, shown in the blue line is in between $85 - 90\%$. It is also interesting to note that the accuracy changes across images that suggests that some images are "easy" for the method that may also help improve the segmentation of the neighboring images, whereas some images are "difficult" representing significant structural discontinuities or noise.

Figure 4.7 depicts a visual comparison of our results for *K-I* images and that obtained using some popular segmentation methods. Active contour model from [173] and JFilament 2D plugin from [174], both semi-automatic (SA) methods, were used to obtain Figure 4.7 (h) and (e), with the initial contour configuration provided manually as shown in Figure 4.7 (g) and (d) respectively. Among the automatic (A) methods, we obtained a 2nd order response using SteerableJ plugin [187] and used mean pixel intensity of the whole filtered image as segmentation threshold to produce 4.7 (f). We used "Squassh" [186] from "Mosaic" toolkit to obtain Figure 4.7 (i). A GPU-based 3D level set software [241] was used with minimal parameter tuning. All other methods used default parameters unless specified.

As observed from Table 4.3, our proposed method clearly outperforms all other methods in terms of accuracy. Our method also produced the lowest *Type-II* errors indicating lowest missed tubule boundaries. Our *Type-I* errors are reasonably low. Active contour with the lowest *Type-I* errors suffered from a considerably high percentage of missed detections. As seen in Figure 4.7, our method successfully produced significantly better visual outcomes than every other method and is very close

(a) Original

(b) Hand-segmentation

(c) Jelly filling (proposed)

(d) JFilament [174] init.

(e) JFilament [174] final

(f) SteerableJ [187]

(g) Active contour [173] init.

(h) Active contour [173] final

(i) Squassh [186]
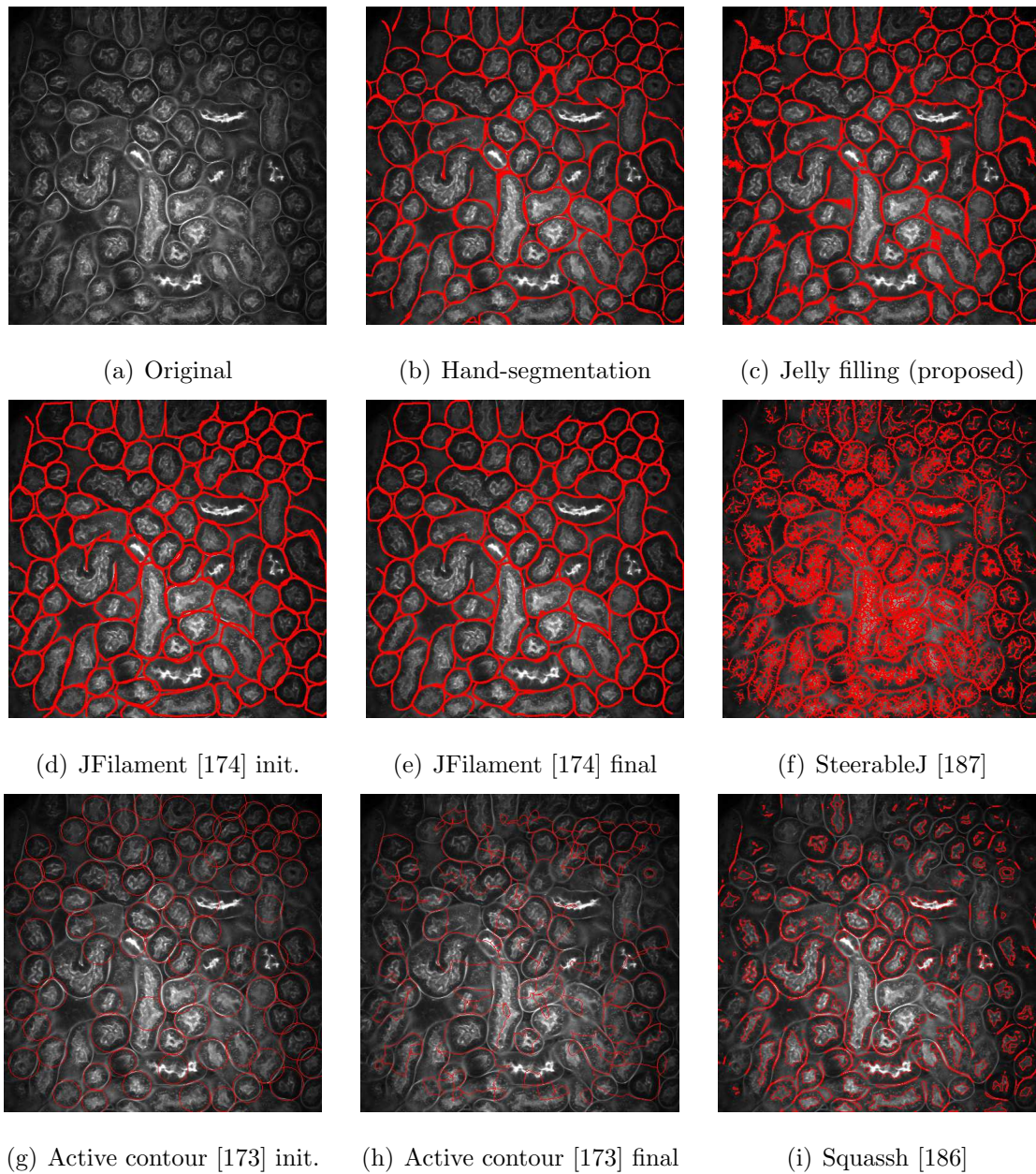
Fig. 4.7.: Visual comparison of segmentation results overlaid on the original image

to the hand-segmented ground truth and the result obtained using the JFilament [174] method which required a considerable user interaction and time.

Figure 4.8, 4.9, 4.10, 4.11 depicts a visual comparison of our results and that obtained using other segmentation methods for *L-I*, *L-II*, *L-III* and *L-IV* respectively.

Method	Class	Accuracy	*Type-I*	*Type-II*	Time
Active contour [173]	SA	86.3%	2%	11.7%	50 min
JFilament [174]	SA	90.4%	6.1%	3.5%	40 min
SteerableJ [187]	A	72.4%	22.3%	5.3%	10 sec
3D level set [241]	A	80.2%	8.1%	11.7%	10 sec
Squassh [186]	A	83.5%	5.6%	11%	20 sec
Jelly filling (proposed)	A	91.2%	6.1%	2.7%	80 sec

Table 4.3: Performance comparison of our proposed method with other popular segmentation approaches

Figure 4.8 (c), (d) and (e) are the results of our proposed method for *L-I*. Our method is able to segment the cell boundaries and highlight the vascular space. Similar results are obtained for *L-II*, *L-III* and *L-IV*. Figure 4.12 depicts more examples of results obtained using our proposed method. The images are from *L-V* and *L-VI* and or proposed method has produced an acceptable outcome.

Table 4.4, 4.5, 4.6 and 4.7 provides accuracy, *Type-I* and *Type-II* errors for *L-I*, *L-II*, *L-III*, *L-IV* respectively. Among all other comparison methods, Squassh [186] seems to perform the closest to our proposed method. As observed in Table 4.4, Squassh [186] gives a better accuracy of segmenting boundaries than our proposed method for data *L-I*. For *L-II*, (as indicated in Table 4.5), it has the same accuracy as that of our proposed method. However, for *L-III* and *L-IV*, our proposed method clearly outperforms other methods. It has also produced an acceptable *Type-I* and *Type-II* errors.

Figure 4.13 depicts examples of segmentation results for *M-I-M-IV* mammogram images. It is intended that breasts from the images are highlighted and isolated from the body. Further, the fat tissue inside each breast should be isolated for quantification purposes. The breast boundaries and body outline are segmented and shown in red and the tissues inside breasts shown in green. Therefore, our method is also

(a) Original

(b) Hand-segmentation

(c) Jelly filling (proposed)

(d) Jelly filling (proposed)

(e) Jelly filling (proposed)

(f) SteerableJ [187]

(g) Region Competition [183]

(h) Active contour [173] final

(i) Squassh [186]
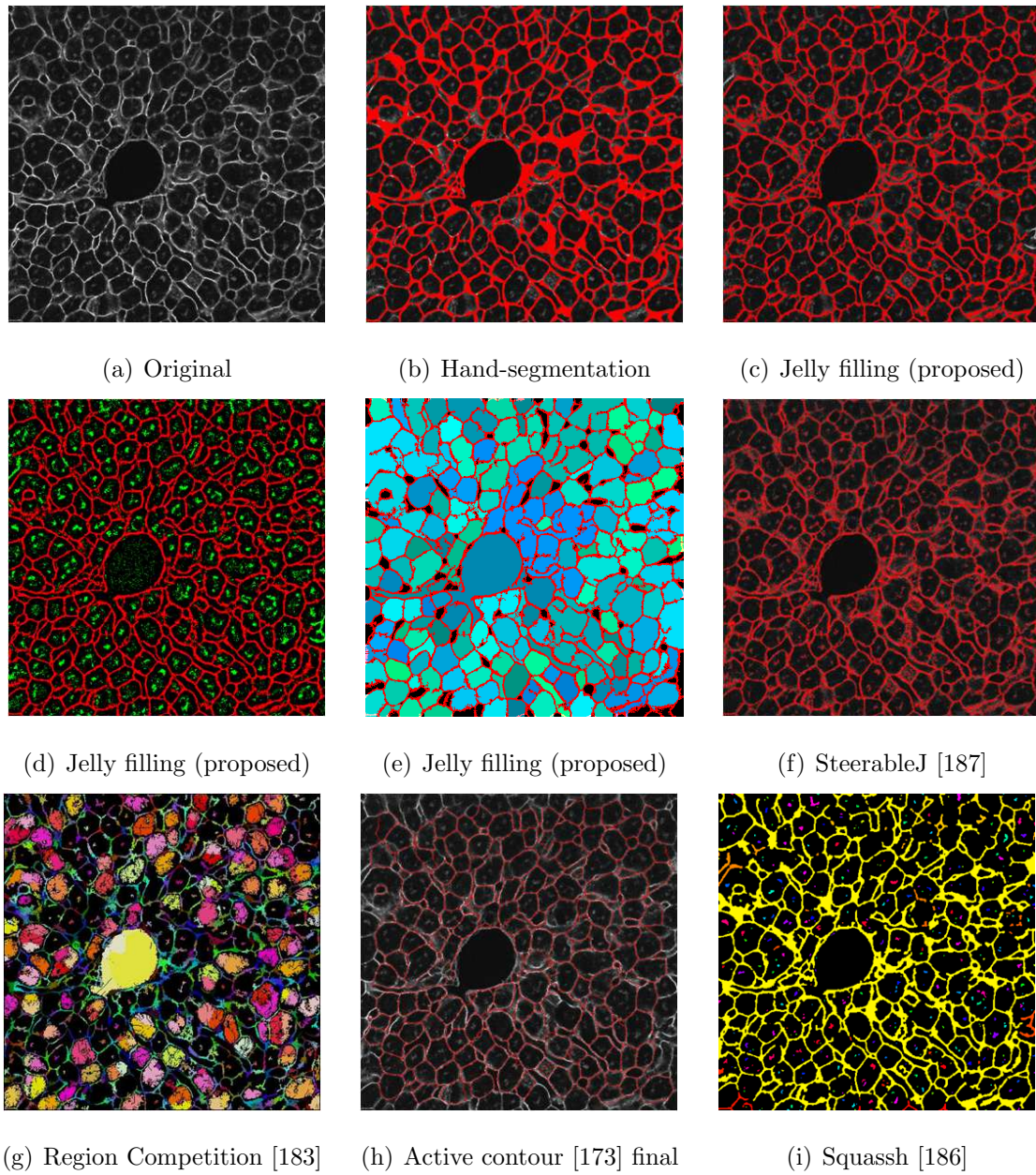
Fig. 4.8.: Visual comparison of segmentation results (*L-I*)

effective in segmenting breasts regions and isolating fat tissues in MRI mammography images.

As shown in Figure 4.14 using examples from *K-I* and *L-VII*, our proposed method failed to produce desirable outcome. In the example shown as Figure 4.14 (d), our

(a) Original   (b) Hand-segmentation   (c) Jelly filling (proposed)

(d) Jelly filling (proposed)   (e) Jelly filling (proposed)   (f) SteerableJ [187]

(g) Region Competition [183]   (h) Active contour [173] final   (i) Squassh [186]
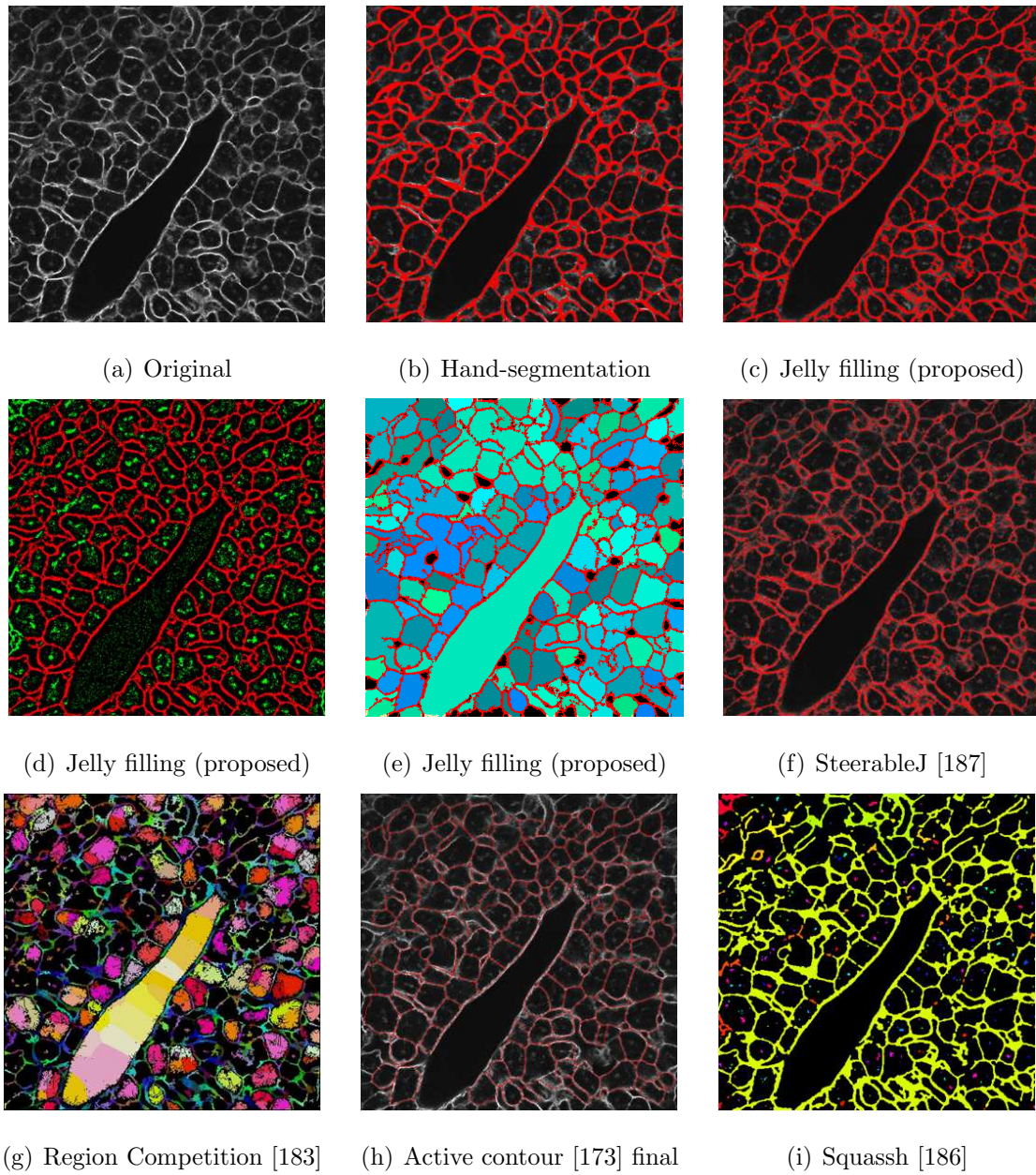
Fig. 4.9.: Visual comparison of segmentation results (*L-II*)

method successfully highlighted the glomerulus in the image as a part of tubule, since they are structurally connected. However, it falsely detected many tubule boundaries while it was a part of brush border of the kidney. In the example shown in Figure 4.14 (e), much of the cell boundary part of the liver was missed and falsely classified as lu-

(a) Original   (b) Hand-segmentation

(c) Jelly filling (proposed)   (d) Jelly filling (proposed)   (e) Jelly filling (proposed)

(f) Region Competition [183]   (g) Squassh [186]
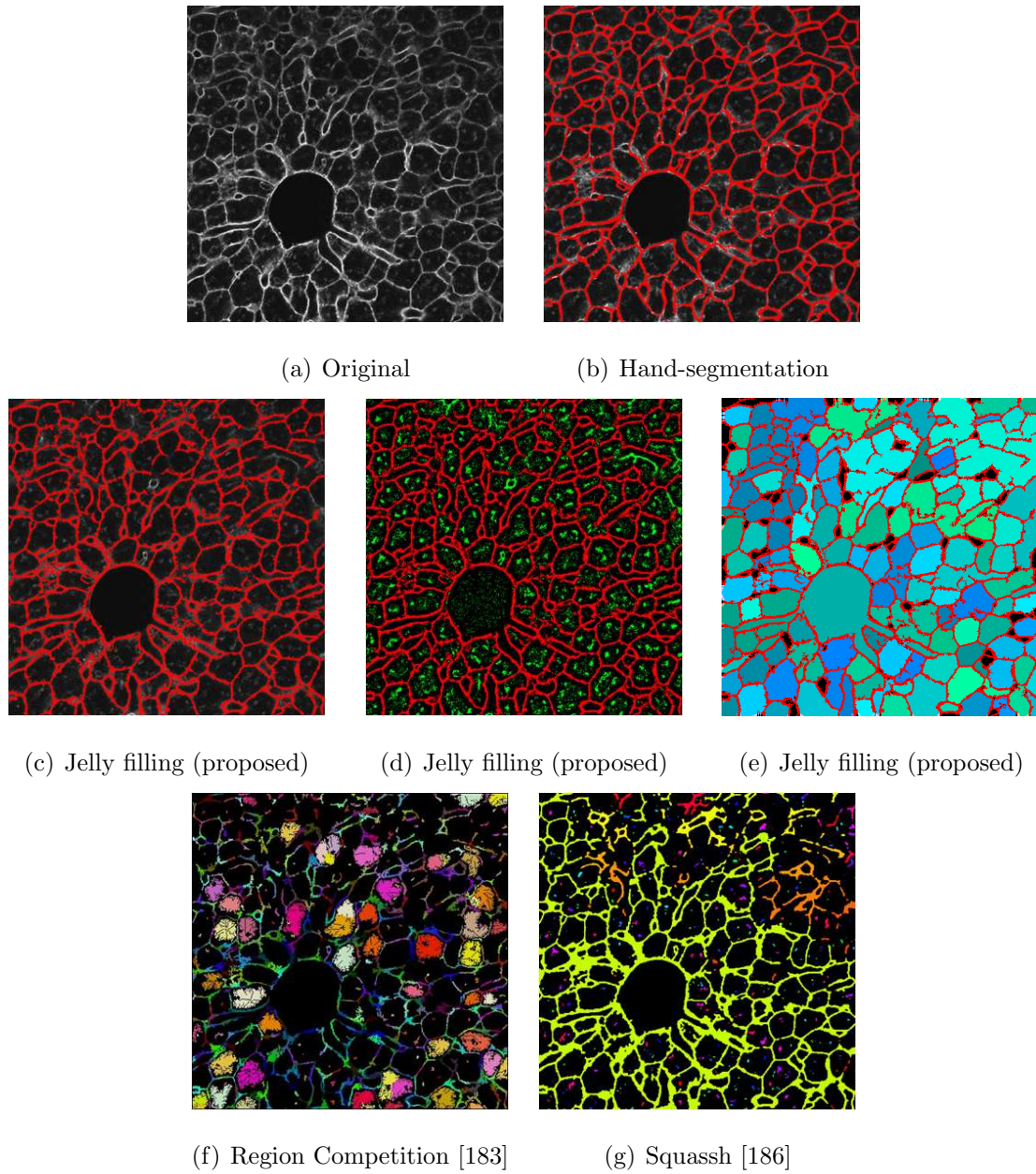
Fig. 4.10.: Visual comparison of segmentation results (*L-III*)

men. Both of these images are from deeper focal planes of the tissues. Therefore, the pixel intensities in the original images are low and the edges are not defined clearly. The images in the z-direction neighborhood also suffer from these artifacts. Therefore the z-direction correction is not considerably effective. Image from Figure 4.14 (f) is

(a) Original · (b) Hand-segmentation

(c) Jelly filling (proposed) · (d) Jelly filling (proposed) · (e) Jelly filling (proposed)

(f) Region Competition [183] · (g) Squassh [186]
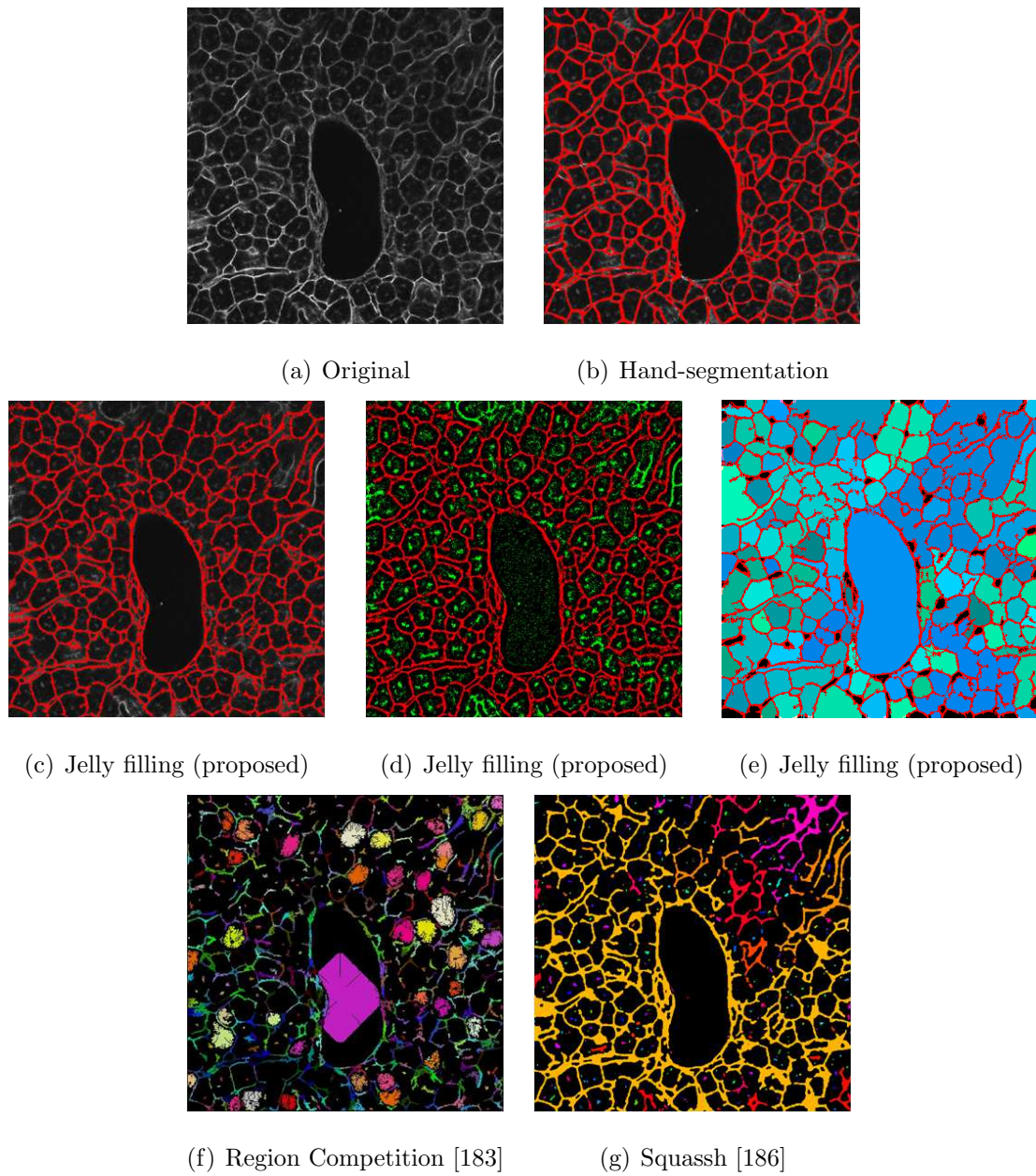
Fig. 4.11.: Visual comparison of segmentation results (*L-IV*)

the result of a single image *L-VII*. The reason of failure is again, low pixel intensity, noise and blurriness in the original image.

3D Visualization: Figure 4.15 show a 3D visualization of the segmentation results

(a)

(b)

(c)

(d)
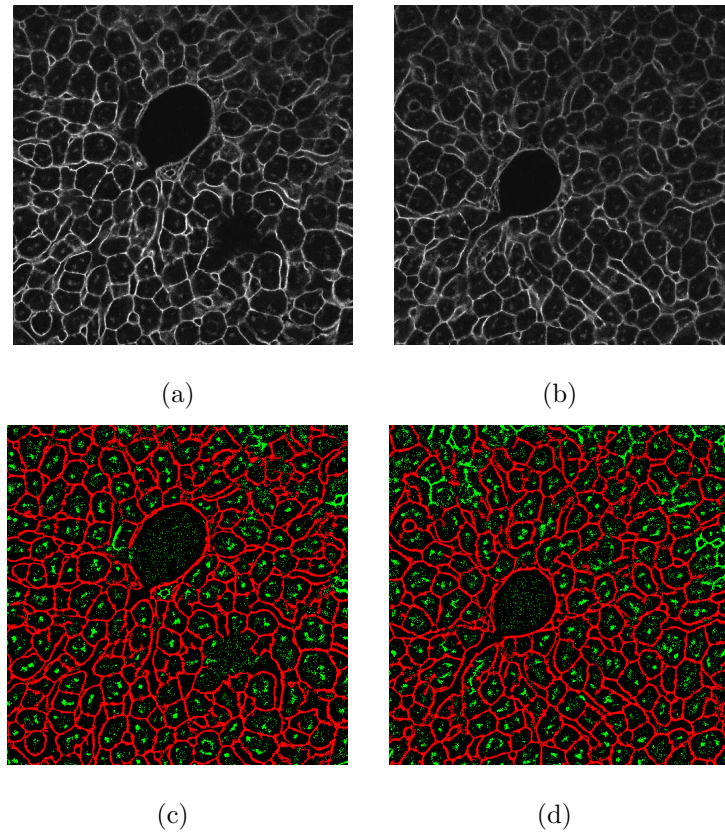
Fig. 4.12.: Segmentation results (for *L-V* and *L-VI*): top row- original images, bottom row- boundaries (red) and lumen (green)

Method	Class	Accuracy	*Type-I*	*Type-II*	Time
Active contour [173]	SA	78.8%	0.6%	20.5%	50 min
SteerableJ [187]	A	82.4%	4.2%	13.4%	10 sec
Squassh [186]	A	87.6%	5.4%	7.1%	20 sec
Jelly filling (proposed)	A	86.5%	4.2%	9.2%	80 sec

Table 4.4: Performance comparison: *L-I*

for *K-I* and *L-I*, obtained using the 3D visualization tool Voxx [195]. Figure 4.15 (a) - Figure 4.15 (c) depict the structure of tubule boundaries (red) and lumen (green) in the kidney. In Figure 4.15 (c), the glomeruli from the specimen connected to tubules

Method	Class	Accuracy	*Type-I*	*Type-II*	Time
Active contour [173]	SA	82%	0.7%	17.3%	50 min
SteerableJ [187]	A	84.7%	3.5%	11.8%	10 sec
Squassh [186]	A	87.9%	6.7%	5.4%	20 sec
Jelly filling (proposed)	A	87.9%	5.3%	6.8%	80 sec

Table 4.5: Performance comparison: *L-II*

Method	Class	Accuracy	*Type-I*	*Type-II*	Time
SteerableJ [187]	A	85.9%	9.2%	4.9%	10 sec
Squassh [186]	A	86.7%	8%	5.3%	20 sec
Jelly filling (proposed)	A	87.9%	6.7%	5.4%	80 sec

Table 4.6: Performance comparison: *L-III*

Method	Class	Accuracy	*Type-I*	*Type-II*	Time
SteerableJ [187]	A	83.9%	2.6%	13.5%	10 sec
Squassh [186]	A	85.4%	6.9%	7.7%	20 sec
Jelly filling (proposed)	A	87.5%	5.1%	7.5%	80 sec

Table 4.7: Performance comparison: *L-IV*

is visible and clearly identifiable. As seen in Figure 4.15 (d) and Figure 4.15 (e), a 3D visualization of the liver is presented.

This demonstrates that our proposed method can produce the desired 3D segmentation that is useful for characterizing the structure and mechanisms of important biological entities. Note that the nuclei shown in Figure 4.15 (d) and Figure 4.15 (e) are segmented from the blue color channel using the method described in the next Chapter.

(a) (b) (c)

(d) (e) (f)
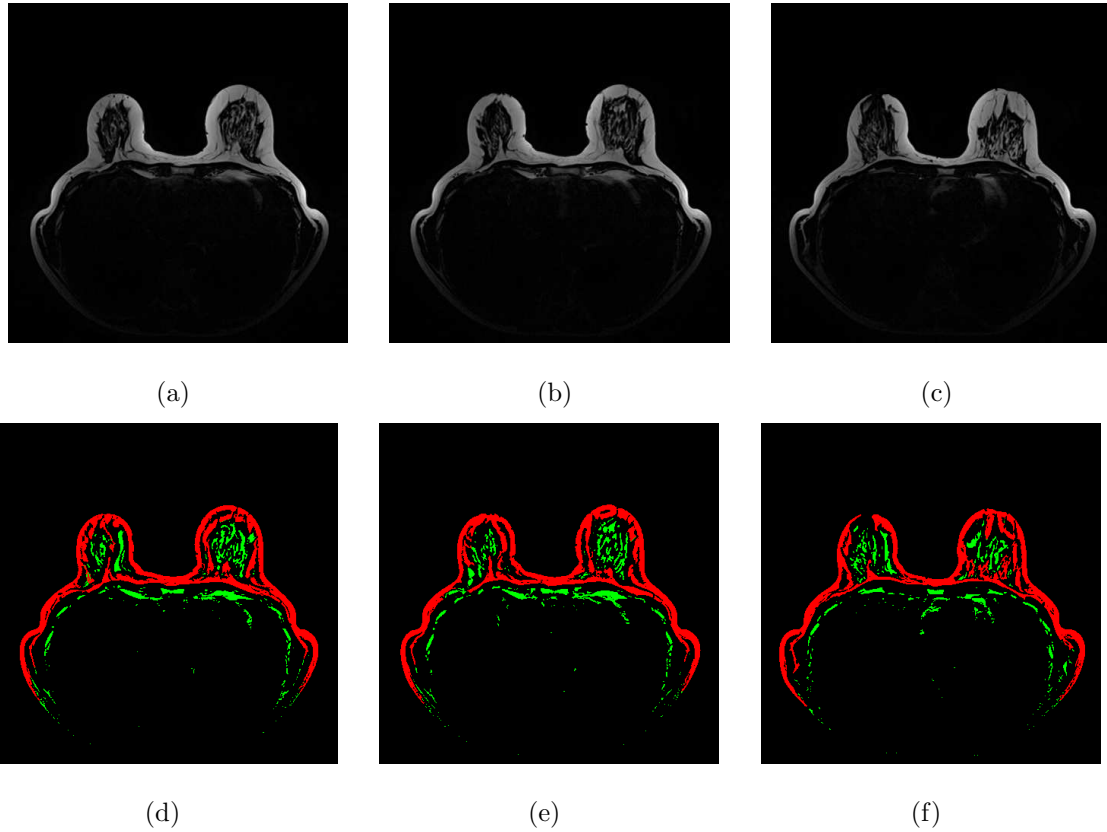
Fig. 4.13.: Segmentation results for *M-I-M-IV*, top row: original mammography Images, bottom row: breast and fat tissue segmentation

(a)　　　　　　　　(b)　　　　　　　　(c)

(d)　　　　　　　　(e)　　　　　　　　(f)
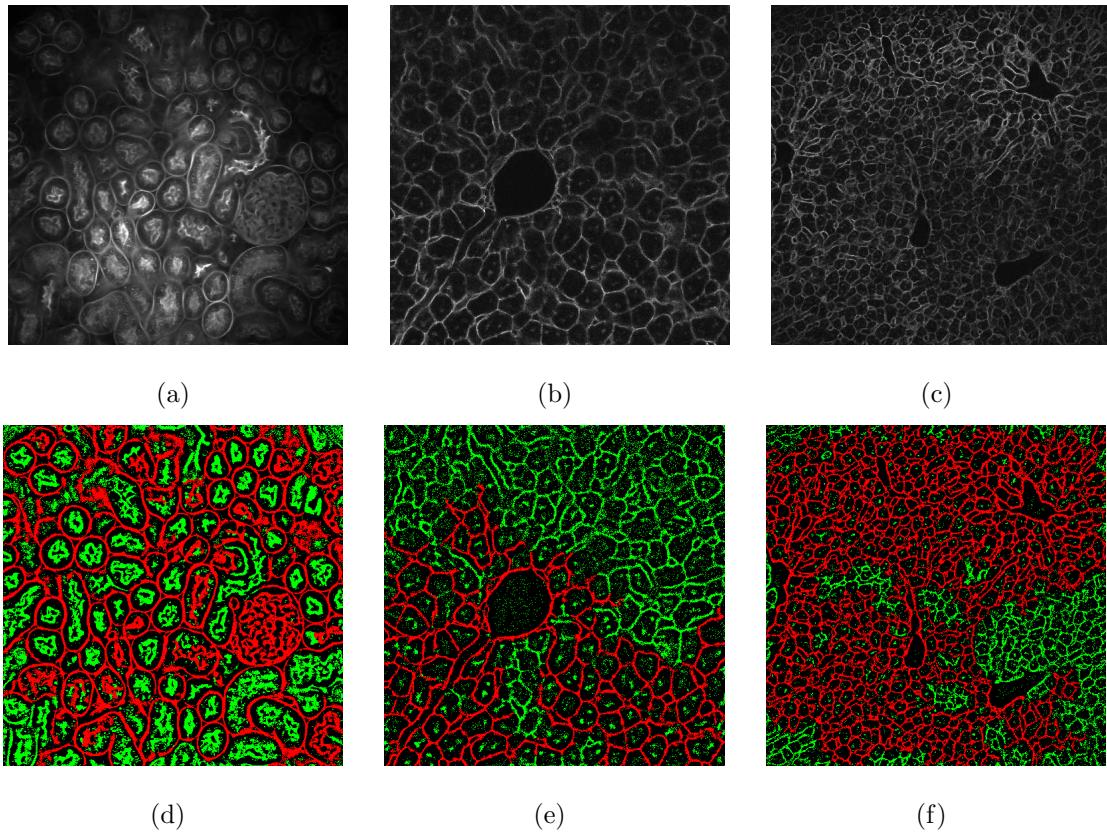
Fig. 4.14.: Segmentation results: failure cases (for *K-I*, *L-I* and *L-VII*): top row-original images, bottom row- boundaries (red) and lumen (green)
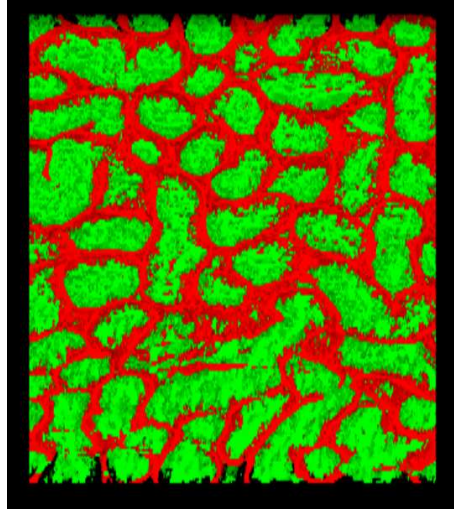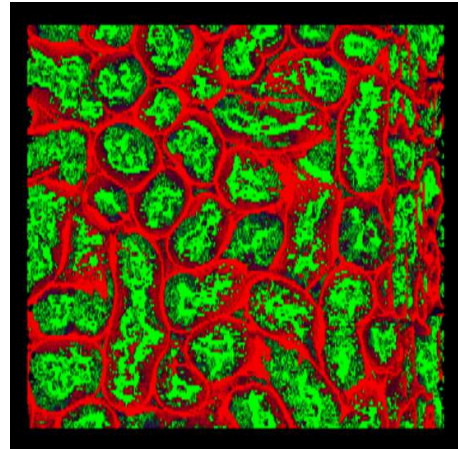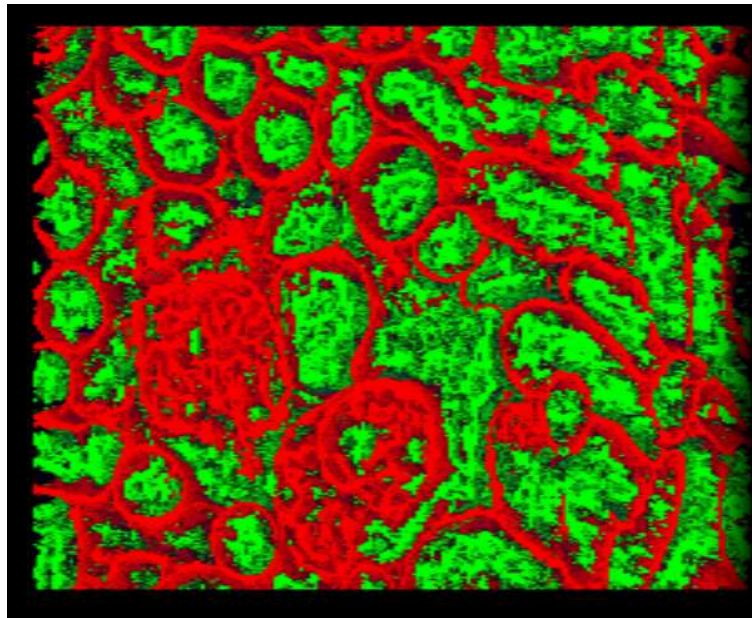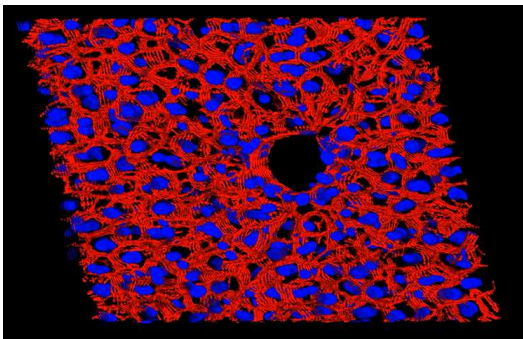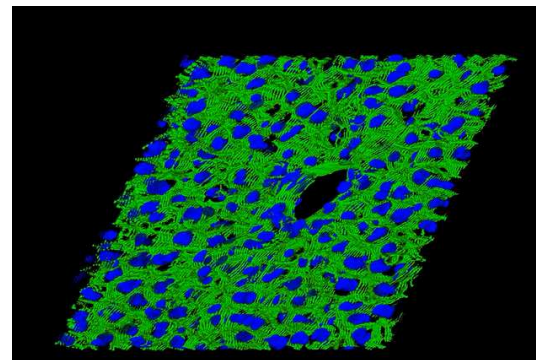
Fig. 4.15.: 3D visualization of different cross-sections of the segmented results for *K-I* and *L-I*.

5. NUCLEI SEGMENTATION USING MIDPOINT ANALYSIS AND MARKED POINT PROCESS

In this chapter, we describe a nuclei segmentation method that makes use of "midpoint analysis" and a 2D marked point process simulation [242]. We first discuss our image analysis goal.

5.1 Image Analysis Goal

Figure 5.1 shows some examples of our image data containing cell nuclei.

(a)	(b)	(c)
(d)	(e)	(f)
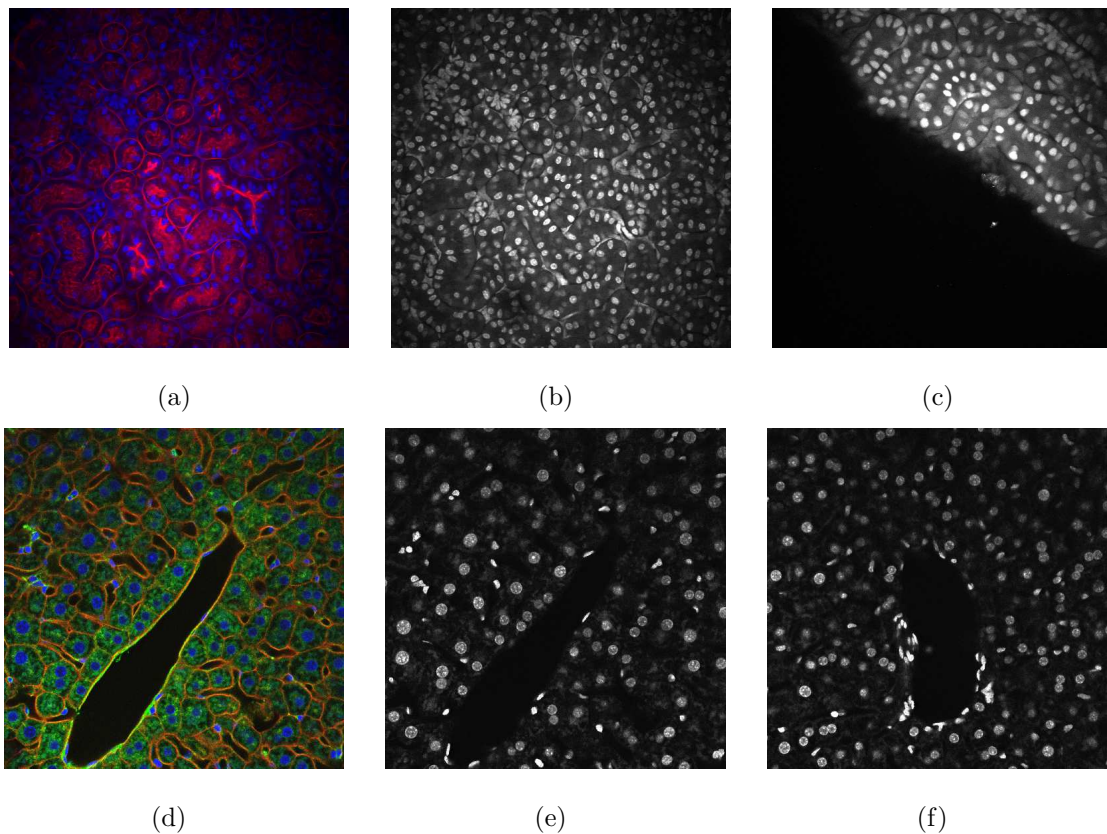
Fig. 5.1.: Examples of our nuclei image data

Figure 5.1 (a), (b) and (c) show images taken from a rat kidney in an *in-vivo* experiment. This data consists of images taken from several hundred focal planes (in the depth dimension) representing a live 3D kidney specimen. Figure 5.1 (a) shows a cross-section of the kidney using red and blue color channels. The blue color channel representing the fluorescence of the dye attached to cell nuclei is shown in Figure 5.1 (b). Another example of kidney images (blue channel) is shown in 5.1 (c). Figure 5.1 (d), (e) and (f) show images taken from a rat liver. The original three channel image is shown in Figure 5.1 (d) and the blue channel representing nuclei is shown in Figure 5.1 (e). Another example of liver images (blue channel) is shown in Figure 5.1 (f).

It is intended to obtain the number of nuclei per unit length, and per unit volume of the specimen. It is also desired to quantify area/volume of each nucleus.

5.2 Overview Of Marked Point Process (MPP) For Image Segmentation

A marked point process (MPP) is a statistical point process in which a "mark" is attached to each event [243–245]. A stochastic simulation of MPP was used as a powerful image analysis approach in which geometric properties of an image are used as the prior distribution and image data are considered at the object level.

Figure 5.2 shows a random configuration of elliptical objects specified by their marks: centers and shape parameters. The parameters of the underlying probability distribution function can be estimated using the application data such that the object configurations are generated at each set of random trials [246]. There are many simulation approaches for MPP. Metropolis-Hastings [247] is a classic one-step birth-death approach in which an object is either born or killed or kept unchanged in each iteration. Reversible jump Markov Chain Monte Carlo algorithm (RJMCMC) [248] is another popular stochastic simulation approach. A subsequent investigation into MPP-based methods has led to the development of a stochastic birth-death approach [249] (along with theoretical analysis) that was effectively used for tree crown extraction
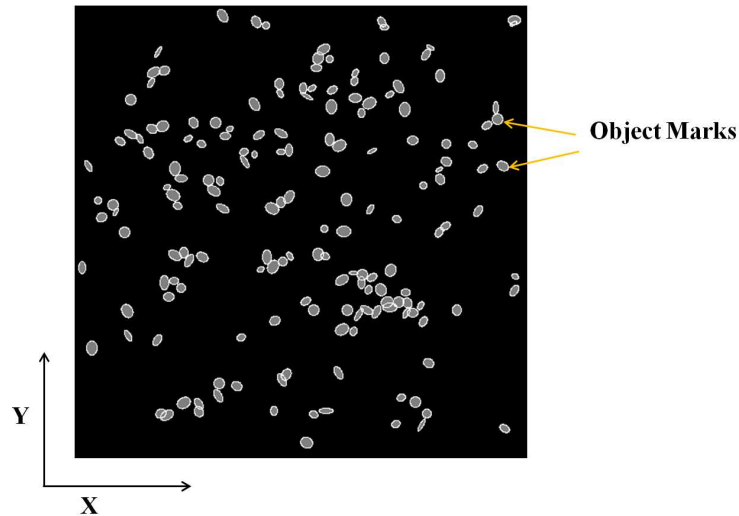
Fig. 5.2.: An example of marked object configuration

from aerial images. A review by Descombes [245] discussed in detail various potential applications of an MPP simulation framework.

Simulating stochastic processes requires a large number of iterations and demands large computation resources [250, 251]. To assign a value that represents likelihood of the object with a given configuration to each possible object orientation at each pixel and search in that high dimensional space is also computationally intensive [252]. Many adaptive approaches have been developed to address these and other challenges. A unified Markov random field (MRF) and MPP based method was developed for micrograph analysis of materials in [253]. An application of MPP to detect small brain lesions using a reversible jump Markov chain Monte Carlo (RJMCMC) algorithm was described in [254]. Many methods have been developed for surveillance applications. MPP based methods are increasingly used in complex image segmentation problems and event simulations, yet its use in biomedical analysis is hardly investigated.

We describe a nuclei segmentation method in which we use adaptive thresholding and midpoint analysis as pre-processing that classifies components such that the computationally expensive MPP is used only for some components and a relatively simpler shape-fitting method for the rest of the components [242]. Our MPP method

is based on the one used in [249]. In our implementation, we use ellipse as the object model and a modified energy function to account for non-uniform brightness typically present in fluorescence microscopy images.

Our proposed approach is intended to provide automatic segmentation of microscopy images with non-uniform brightness, consisting of multiple overlapping objects that can be modeled using specific geometric shapes.

The details of our segmentation method are provided below.

5.3 Our Proposed Method

As indicated above, our images ($\mathcal{I}_{z_p,c_r}$) consists of multiple-channels that reflect the fluorescence of dyes added to the tissue. As shown in Figure 5.3, we first separate
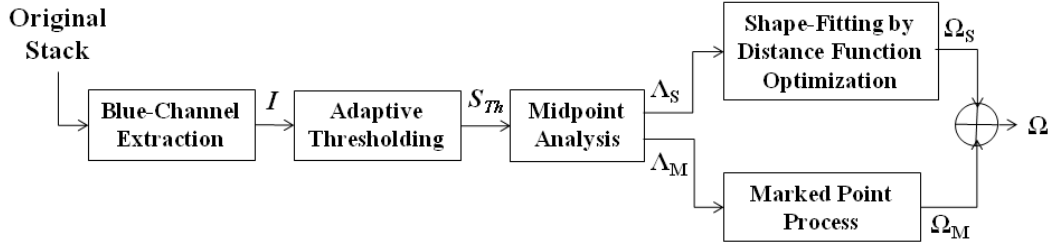
Fig. 5.3.: Our proposed segmentation method.

the blue color channel to obtain grayscale images $\mathcal{I}_{z_p,c_2}$, where c_2 represents the blue-channel data and $p \in \{1, 2, ..., P\}$ denotes the focal planes. We use only the blue color channel throughout this chapter, hence we drop the subscript c_2 to call the grayscale images $\mathcal{I}_{z_p}$, $p \in \{1, 2, ..., P\}$.

A 3D adaptive thresholding is then employed on $\mathcal{I}_{z_p}$ to get segmentation mask $\mathcal{S}_{Th,z_p}$. For each pth image, midpoints analysis is subsequently used to produce two distinct masks: Λ_{S,z_p} and Λ_{M,z_p}. A distance function optimization method is used with the first mask and a MPP based method is used with the second mask. Segmentation results of the two methods, Ω_{S,z_p} and Ω_{M,z_p} respectively, are combined to

produce the final segmented image Ω_{z_p}.

Adaptive Thresholding: Our method employs initially an adaptive thresholding scheme. The objective of this step is to separate the foreground that represents the presence of a biological quantity in an image. This is done using two functions: a thresholding function f_{Th,z_p} that uses a 3D neighborhood information to assign signed and scaled value to each pixel and a voting function f_{v,z_p} that uses a Gaussian filter to aggregate weighted votes, similar to the voting-based distributing function used in [182].

Let $\mathcal{I}_{z_p}(s) \in [0,1]$ be the pixel intensity at pixel s of the original images $\mathcal{I}_{z_p}$, $p \in \{1, 2, ..., P\}$. Let $(w_{Th,x} \times w_{Th,y} \times w_{Th,z})$ be the 3D window centered at pixel s and let $\tau_{z_p}(s)$ be the mean pixel intensity of this window. The thresholding function $f_{Th,z_p} : [0,1] \rightarrow [-1,1]$ is used to assign to each pixel s of pth image a linearly scaled value and a sign, based on its original intensity $\mathcal{I}_{z_p}(s)$ and the local mean $\tau_{z_p}(s)$, as indicated by Eq. 5.1.

$$f_{Th,z_p}(s) = \begin{cases} \frac{\mathcal{I}_{z_p}(s)-(\tau_{z_p}(s)+\tau_c)}{1-(\tau_{z_p}(s)+\tau_c)} & \text{if } \mathcal{I}_{z_p} \geq (\tau_{z_p}(s)+\tau_c) \\ -\frac{(\tau_{z_p}(s)+\tau_c)-\mathcal{I}_{z_p}(s)}{(\tau_{z_p}(s)+\tau_c)} & \text{if } \mathcal{I}_{z_p} < (\tau_{z_p}(s)+\tau_c) \end{cases} \tag{5.1}$$

where τ_c is a positive constant. Let $g_v(x,y,z)$ be a 3D truncated Gaussian function: $g_v(x,y,z) = e^{-\frac{|x|^2+|y|^2+|z|^2}{a^2}}$, where $x = -w_{v,x}, .., 0, .., w_{v,x}$, $y = -w_{v,y}, .., 0, .., w_{v,y}$ and $z = -w_{v,z}, .., 0, .., w_{v,z}$. The voting function $f_{v,z_p} : [-1,1] \rightarrow [-\infty, \infty]$ is used to assign each pixel a value that is the summation of f_{Th,z_p} values from its neighborhood, weighted using $g_v(x,y,z)$:

$$f_{v,z_p}(s) = (f_{Th,z_p} * g_v)(s) \tag{5.2}$$

Now, based on the sign of $f_{v,z_p}(s)$, pixel s is segmented as foreground mask: $\mathcal{S}_{Th,z_p} = \{s : f_{v,z_p}(s) \geq 0\}$, where $\mathcal{S}_{Th,z_p}$ is the set of foreground pixels from images $\mathcal{I}_{z_p}$, $p \in \{1, 2, ..., P\}$. The outcome of this step: initial segmentation mask $\mathcal{S}_{Th,z_p}$ is used in the subsequent steps to do nuclei segmentation.

The constant threshold τ_c is selected empirically for a particular image stack based on the desired brightness of a segmented nuclei. A higher τ_c reflects segmenting fewer pixels with intensities being significantly above the local mean. $\tau_c \geq 0$ is necessary to avoid assigning regions with pixel intensities ≈ 0 as foreground.

Object Model: In order to count the number of nuclei and quantify their size in each pth image, we use an ellipse as the shape model for objects to be segmented. The shape parameters for each elliptical object centered at (c) are the lengths of the semi-major and semi-minor axes (a, b) and the orientation angle of the semi-major axis with the horizontal (θ). Let $\rho = (a, b, \theta)$ be the parameter vector such that $\rho \in \mathcal{P}$, the parameter space. Based on the size of objects to be segmented, we limit the parameter space with $a \in (a_{\min}, a_{\max})$ $b \in (b_{\min}, b_{\max})$. Also, Δ_θ be the stepsize considered for angular orientations θ of an object.

Midpoint Analysis: Let $\mathcal{S}_{Th, z_p}$ be the segmentation mask for pth image. Let λ be a connected component of $\mathcal{S}_{Th, z_p}$, using a 4-point neighborhood. Small components can be safely removed to preserve a high-level structural continuity. Therefore, λ in which the number of pixels is smaller than a threshold ν is not considered for midpoint analysis. The goal of midpoint analysis is to classify λs in $\mathcal{S}_{Th, z_p}$ into two groups: single-object components (Λ_{S, z_p}) and multiple-objects components (Λ_{M, z_p}). We first determine the potential midpoint locations/pixels by horizontally and vertically scanning the rows and column respectively, as shown in Figure 5.4 (a), using a process similar to the one described in [255]. This generates two sets of potential midpoints $\{m_{c,x}\}$ and $\{m_{c,y}\}$ along the rows and columns of the connected component λ, and which are depicted in blue and orange respectively, in Figure 5.4 (a). A pixel that is detected in both $\{m_{c,x}\}$ and $\{m_{c,y}\}$ is called as a midpoint pixel m_c as indicated by the pixel colored in red.
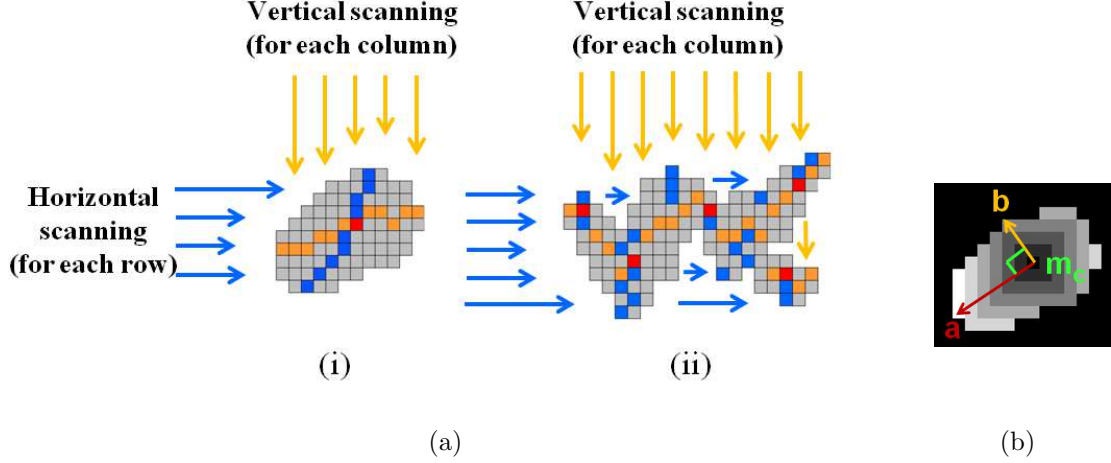
Fig. 5.4.: Examples of midpoint analysis and selecting ellipse parameters for shape-fitting.

We assume that the components with one or no midpoint pixel m_c contain a maximum of one object and hence belong to Λ_{S,z_p}. Components containing more than one m_c may contain multiple objects and belong to Λ_{M,z_p}. Thus,

$$\Lambda_{S,z_p} = \{\lambda : \lambda \text{ contains at most one } m_c\} \qquad (5.3)$$

$$\Lambda_{M,z_p} = \{\lambda : \lambda \text{ contains more than one } m_c\text{'s}\} \qquad (5.4)$$

where $\Lambda_{S,z_p} \cap \Lambda_{M,z_p} = \phi$. An example component shown in Figure 5.4 (a)(i) belongs to Λ_{S,z_p} and that in Figure 5.4 (a)(ii) belongs to Λ_{M,z_p}.

Shape Fitting by Distance Function Optimization: We use a distance function to determine the parameters of the elliptical object for a $\lambda \in \Lambda_{S,z_p}$ in a pth image. Let A_1 be an elliptical disk centered at s with parameters $\rho = (a, b, \theta)$. Let A_2 be the outer elliptical ring with parameters $(a+1, b+1, \theta)$. Using the empirical means and variances of the pixels belonging to A_1 and A_2 at pixel s with parameters ρ:

$$\mu_1(s, \rho) = \frac{\sum_{u \in A_1} I_j(u)}{N_1}, \; \sigma_1^2(s, \rho) = \frac{\sum_{u \in A_1} I_j^2(u)}{N_1} - \mu_1^2(s, \rho) \qquad (5.5)$$

$$\mu_2(s, \rho) = \frac{\sum_{u \in A_2} I_j(u)}{N_2}, \; \sigma_2^2(s, \rho) = \frac{\sum_{u \in A_2} I_j^2(u)}{N_2} - \mu_2^2(s, \rho) \qquad (5.6)$$

we obtain $B(s, \rho)$ the Bhattacharyya distance [249, 253] between the distributions of the pixels contained A_1 and those contained in A_2:

$$
\begin{aligned}
B(s, \rho) = {} & \frac{1}{4}(\mu_1(s, \rho) - \mu_2(s, \rho))^2 \sqrt{\sigma_1{}^2(s, \rho) + \sigma_2{}^2(s, \rho)} \\
& - \frac{1}{2} log(\frac{2\sigma_1(s, \rho)\sigma_2(s, \rho)}{\sigma_1{}^2(s, \rho) + \sigma_2{}^2(s, \rho)})
\end{aligned}
\tag{5.7}
$$

Recall that Λ_{S,z_p} consists of components of $\mathcal{S}_{Th,z_p}$ with at most one m_c. For a λ with no pixel determined as m_c, the X and Y coordinates of m_c are approximated by rounding the means of the X-coordinates of the set of $\{m_{c,x}\}$ and the Y-coordinates of the set of $\{m_{c,y}\}$, respectively. Next, as shown in Figure 5.4 (b), a pixel $s_a \in \lambda$ that is farthest (in Euclidean distance) from m_c is obtained. The vector from m_c to s_a is considered the semi-major axis, and a is considered to be the length of the vector. The orientation θ is now the angle that the vector from m_c to s_a subtends with the horizontal axis. A vector that is perpendicular to the semi-major axis is drawn from m_c within λ and b is determined to be the length to the farthest pixel along that vector. Next, a rectangular pixel window $W_{c,\lambda}$ of size $(w_c \times w_c)$ centered at m_c is further examined for other candidates for the object center, and a corresponding parameter space, $\mathcal{P}_\lambda = [a \pm w_a] \times [b \pm w_b] \times [\theta \pm w_\theta]$ is formed by varying a, b and θ. The center candidate and parameters from $W_c \times (\mathcal{P}_\lambda \cap \mathcal{P})$ that maximize $B(s, \rho)$, are chosen as the ellipse center c_λ with parameters ρ_λ for λ, that is

$$
(c_\lambda, \rho_\lambda) = \underset{s \in W_c, \rho \in \mathcal{P}_\lambda \cap \mathcal{P}}{\arg\max} \; B(s, \rho)
$$

where $\mathcal{P}$ is the parameter space defined for our object model. An object centered at c_λ with parameters ρ_λ is thus generated.

Marked Point Process: We employ a marked point process approach based on the spatial birth-death process described in [249]. In our method, an object can be generated only when its center pixel belongs to Λ_{M,z_p}.

Objects of different parameters can be generated based on their relative probabilities. We also incorporate two additional energy functions to account for non-uniform

brightness and local behavior of the birth rate function. A detailed proof of the convergence of this spatial birth-death process is presented in [249]. We do not provide a theoretical proof of our method. We expect the outline of the proof to be very similar to the base-method described in [249]. However, we would like to point out that our proposed method reports convergence for all images that we used, with a broad range of parameters without the need of fine-tuning.

Let Γ be the configuration of objects with their corresponding parameters. $\Gamma = (\Gamma_s, \Gamma_\rho)$, where Γ_s is a set of pixels that are object centers and Γ_ρ is a set of their respective parameters. Let $H(\Gamma)$ be the energy function for the Gibbs distribution function for the configuration Γ during the spatial birth-death process simulation. Let $H_{\text{Object}}(s, \rho)$ be the term representing how well the object centered at s with parameters ρ fits the image data I_j:

$$H_{\text{Object}}(s, \rho) = \begin{cases} \frac{1 - B(s,\rho)}{T} & \text{if } B(s,\rho) \geq T \\ e^{-\frac{B(s,\rho) - T}{3B(s,\rho)}} - 1 & \text{if } B(s,\rho) < T \end{cases}$$

where $B(s, \rho)$ is distance measure described in Equation 5.7 and T is a threshold. Let $H_{\text{Brightness}}(s)$ be the term accounting for the local brightness in the neighborhood of s in an image. $H_{\text{Brightness}}(s) = \tau_s$, where τ_s is the local mean for pixel s and was used in adaptive thresholding. We define birth energy $H_B(s, \rho)$ and birth rate $b(s, \rho)$ at pixel s for parameter set ρ as:

$$H_B(s, \rho) = H_{\text{Object}}(s, \rho) + H_{\text{Brightness}}(s),$$

$$b(s, \rho) = 1 + 9 \frac{\max(H_B(s, \rho)) - H_B(s, \rho)}{\max(H_B(s, \rho) - \min(H_B(s, \rho)}$$

Let $b_c(s)$ be the cumulative birth rate and $b_n(s)$ be the normalized birth rate at pixel s:

$$b_c(s) = \sum_{\rho \in \mathcal{P}} b(s, \rho) \ , \ b_n(s) = \frac{b_c(s)}{\max_{s \in \Lambda_M} b_c(s)}$$

Let $H_{\text{Inter}}(s_1, s_2)$ be the energy term corresponding to the object interaction model that accounts for the closeness or overlap between the two objects centered at s_1 and s_2. It is determined based on the Euclidean distance between the centers:

$$H_{\text{Overlap}}(s_1, s_2) = \max(0, 1 - \frac{\|s_1, s_2\|}{2r})$$

Let $H_{\text{Peak}}(s)$ be the energy term representing the local maxima of the cumulative birth rate function. In case of object overlap, this term causes objects not centered at the peaks of the birth rate function to be more prone to being eliminated than the ones centered at the peaks. The local maxima pixels are also used for configuration initialization.

$$H_{\text{Peak}}(s) = \begin{cases} -h_P & \text{if } \sum_{\rho \in \mathcal{P}} b_c(s) \text{ has a local maxima at } s. \\ 0 & \text{Otherwise} \end{cases}$$

where h_P is a positive contant. Therefore the energy function is obtained as:

$$H(\Gamma) = \alpha \{ \sum_{(s,\rho) \in \Gamma} H_{\text{Object}}(s, \rho) + \sum_{s \in \Gamma_s} H_{\text{Brightness}}(s) \}$$
$$+ \sum_{s_1, s_2 \in \Gamma_s} H_{\text{Overlap}}(s_1, s_2) + \sum_{s \in \Gamma_s} H_{\text{Peak}}(s)$$

where α is a positive constant. Now, a multiple birth-death process is simulated to optimize the energy function according to [249]:

- Determine $H_{\text{Object}}(s, \rho)$, $H_{\text{Brightness}}(s)$, $H_B(s, \rho)$, $b(s, \rho)$, $b_c(s)$, $b_n(s)$ and $H_{\text{Peak}}(s)$ for all $s \in \Lambda_M$ and $\rho \in \mathcal{P}$.

- *Parameter Initialization:* Set the inverse temperature $\beta = \beta_0$ and the discretization step $\delta = \delta_0$.

- *Configuration Initialization:* Start with $\Gamma = \Gamma^0$ such that $\Gamma_s{}^0$ contains objects centered at s where $b_c(s)$ achieves local maxima and $\Gamma_\rho)$ contains their parameters $argmax_{\rho \in \mathcal{P}} b(s, \rho)$ for each s respectively.

- *Birth Step:* For each $s \in \Lambda_M$, if $s \notin \Gamma_s$ add a point at s with probability $\delta b_n(s)$ and give birth to an object of parameter ρ with probability $= \frac{b(s,\rho)}{\sum_{\rho \in \mathcal{P}} b(s,\rho)}$.

- *Death Step:* Sort the configuration of points Γ from highest to lowest values of $H_B(s, \rho)$. For each sorted point s obtain death rate $d(s, \rho) = \frac{\delta a(s)}{1 + \delta a(s)}$, where $a(s) = e^{-\beta(H(\Gamma/\{s, \rho\}) - H(\Gamma))}$ and kill the object with probability $d(s)$.

- *Convergence Test:* If all the objects born in the *Birth Step* are killed in the *Death Step*, stop. Otherwise, increase β and decrease δ by a geometric scheme using the common ratios Δ_β and Δ_δ respectively and go back to the *Birth Step*.

The proposed nuclei segmentation method is described next:

Our Proposed Nuclei Segmentation Method

Require: Original images $\mathcal{I}_{z_p, c_r}$

Extract blue color-channel from $\mathcal{I}_{z_p, c_r}$ to obtain grayscale images $\mathcal{I}_{z_p}$, $p \in \{1, 2, ..., P\}$

Do **Adaptive Thresholding** to $\mathcal{I}_{z_p}$ to get $\mathcal{S}_{Th, z_p}$

for Each pth image **do**

    Obtain $\mathcal{S}_{Th, z_p}$ as segmentation mask for the pth image from $\mathcal{S}_{Th, z_p}$

    Use **Object Model** as ellipse with $a_{\min}$, $a_{\max}$, $b_{\min}$, $b_{\max}$ and Δ_θ

    Do **Midpoint Analysis** to obtain Λ_{S, z_p} and Λ_{M, z_p}

    for Each component $\lambda \in \Lambda_{S, z_p}$ **do**

        Shape-Fitting by Distance Function Optimization with $m_{c,i}$'s, w_c, w_a, w_b, w_θ to obtain $(c_\lambda, \rho_\lambda)$ and Ω_{S, z_p}

    Do **Marked Point Process** using $\mathcal{I}_{z_p}$, Λ_{M, z_p}, α, β_0, δ_0, Δ_β, Δ_δ, h_P, r and T to obtain Ω_{M, z_p}

    Combine using OR operation Ω_{S, z_p} and Ω_{M, z_p} to obtain the final segmentation result Ω_{z_p}

5.4 Experimental Results

We tested our method using several images taken from rat kidney (*K-I*, *K-V* and *K-VI*) [1] and liver (*L-I*, *L-II*, *L-III*, *L-IV* and *L-V*) [2] samples using fluorescecne microscopy. We used 32 images from each set of kidney images (*K-I*, *K-V* and *K-VI*). *L-I* contains 36 images. *L-II*, *L-III*, *L-IV* and *L-IV* are single images. Each image had three 8-bit color-channels.

The values used for the various parameters were $w_{Th,x} = w_{Th,y} = 15$, $w_{Th,z} = 3$, $w_{v,x} = w_{v,y} = 2$, $w_{v,z} = 1$ for adaptive thresholding, $\nu = 10$ pixels for midpoint analysis, and $w_a = w_b = 2$ and $w_\theta = 30°$ for shape fitting, $\alpha = 0.5$, $\beta_0 = 0.5$, $\delta_0 = 0.5$, $\Delta_\beta = 1.05$, $\Delta_\delta = 0.95$, $h_P = 2$, $r = \frac{1}{2}(a_{\min} + a_{\max})$ and T as 1 percentile of $B(s, \rho)$, for marked point process. All parameters were selected without fine-tuning and kept unchanged for all images. It took between 70 and 110 MPP iterations for one image to converge to the final configuration. The details of our image data with shape parameters are listed in Table 5.1.

Table 5.1: Details of our image data with specific parameters

Image Data	*K-I*	*K-V*	*K-VI*	*L-I, L-II, L-III, L-IV & L-V*
Dimensions	512×512	640×640	512×512	512×512
τ_c	$10/255$	$5/255$	$10/255$	$5/255$
$(a_{\min}, a_{\max})$	$(4, 8)$	$(6, 14)$	$(4, 14)$	$(4, 14)$
$(b_{\min}, b_{\max})$	$(2, 6)$	$(4, 12)$	$(2, 12)$	$(2, 12)$
Δ_θ	$30°$	$20°$	$30°$	$30°$

Figure 5.5 - Figure 5.10 show some examples of our segmentation results for the kidney images. Each figure contains (a): the original blue channel image ($\mathcal{I}_{z_p}$), (b):

---

[1] The kidney data was provided by Malgorzata Kamocka of Indiana University and was collected at the Indiana Center for Biological Microscopy.

[2] The liver data was provided by Sherry Clendenon and James Sluka of the Biocomplexity Institute, Indiana University at Bloomington.

the result of our midpoint analysis, in which the single-object components (Λ_{S,z_p}) are colored in green and multiple-object components (Λ_{M,z_p}) are colored in blue, (c): the final segmentation result (Ω_{z_p}) and (d): the segmentation result overlaid on the original image in (a). The value of n indicates the count of cell nuclei segmented using elliptical disks from the original images.

(a) Original

(b) Result of Midpoint Analysis

(c) Segmentation Result (n = 628)

(d) Result Overlay
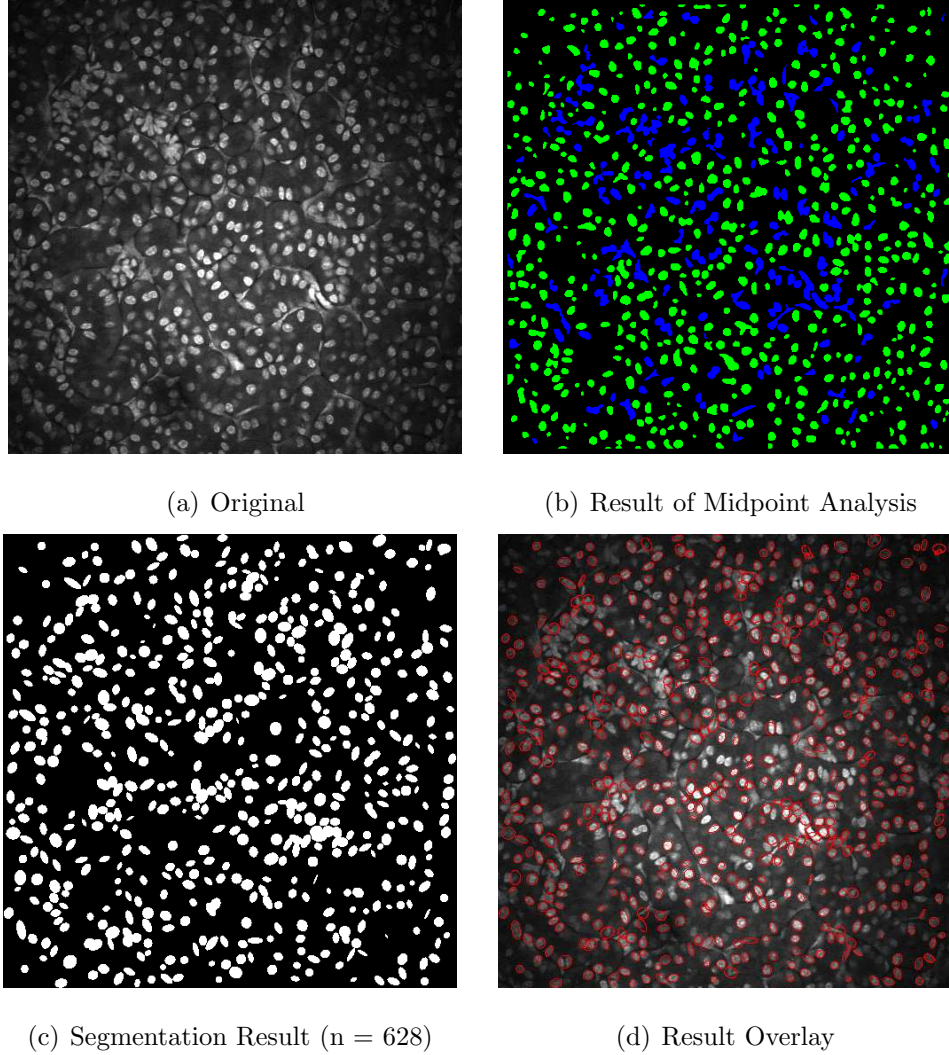
Fig. 5.5.: Segmentation results: *K-I*

It can be observed that the original images from kidney images possess non-uniform brightness. *K-V* contains a large completely dark region, whereas *K-VI* has smaller regions of darkness. Images, especially from *K-I* and *K-V*, contain labeling

(a) Original

(b) Result of Midpoint Analysis

(c) Segmentation Result (n = 594)

(d) Result Overlay
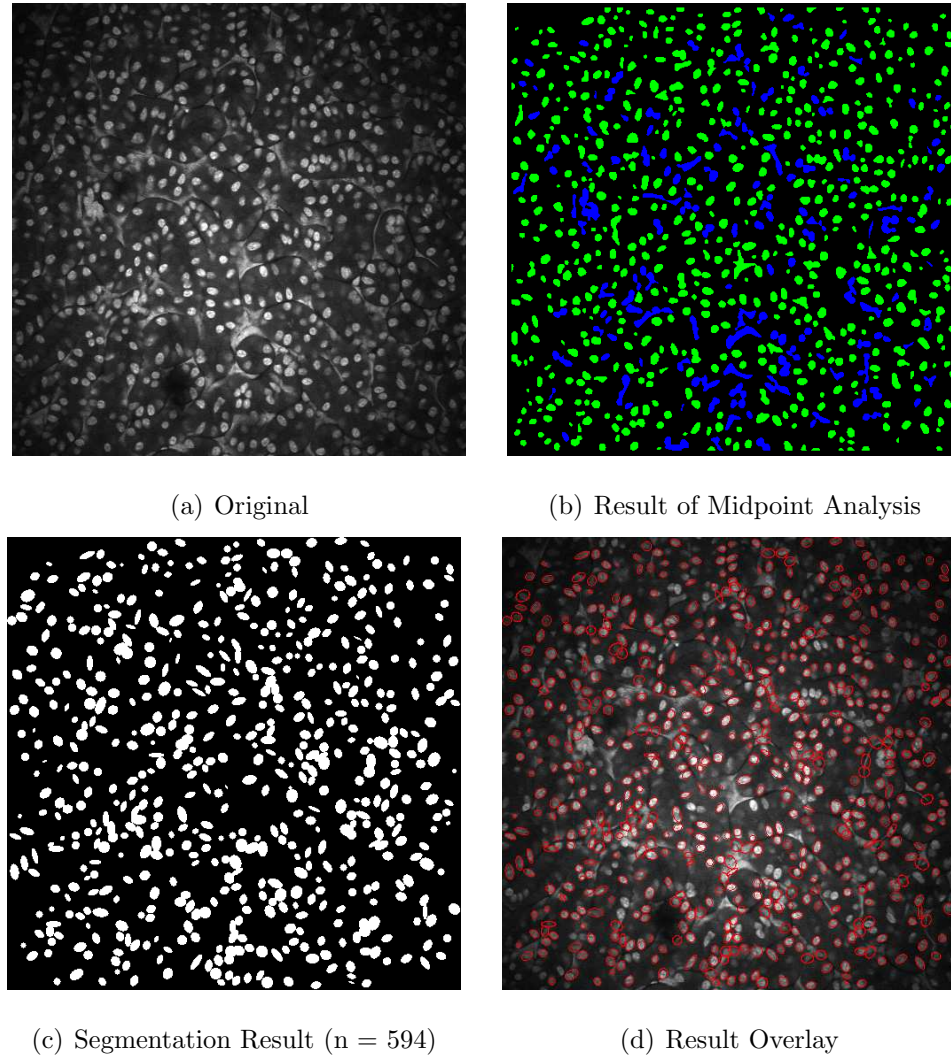
Fig. 5.6.: Segmentation results: *K-I*

errors resulting in frequent appearances of bright regions not representing nuclei. In all cases, our proposed method segments most nuclei present in the bright regions. Also, many nuclei present in the darker regions of images are segmented successfully. A few nuclei are missed as well as detected falsely as the shape parameters were not correctly obtained in those cases. A few red components in Figure 5.7 (b) and Figure 5.8 (b) indicate failure to classify those components into Λ_{S,z_p} or Λ_{M,z_p} because of lack of intersection of vertical and horizontal midpoint loci. This happens because it cannot be theoretically proved that these two midpoint loci would intersect. In

(a) Original

(b) Result of Midpoint Analysis

(c) Segmentation Result (n = 190)

(d) Result Overlay
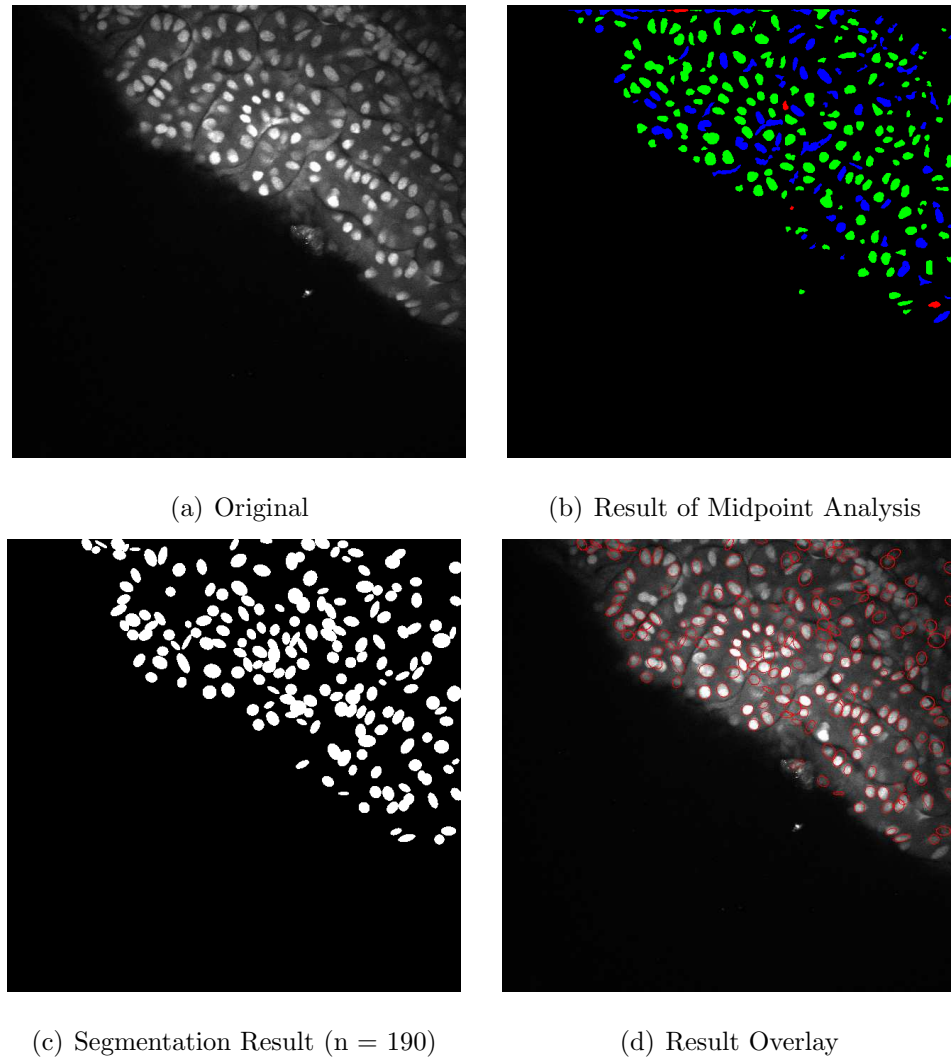
Fig. 5.7.: Segmentation results: K-V

fact, it is possible that an analysis may be able to prove that the midpoint loci, in fact, do not necessarily intersect. For now, we consider these few components that are not classified as either, as failure cases. This should be addressed in a future investigation of midpoint analysis.

Figure 5.11 - Figure 5.15 show segmentation results for our liver images. Each figure contains the original blue channel image at the left, the outcome of our proposed segmentation method at the center and the results overlaid on the original image at

(a) Original

(b) Result of Midpoint Analysis

(c) Segmentation Result (n = 234)

(d) Result Overlay
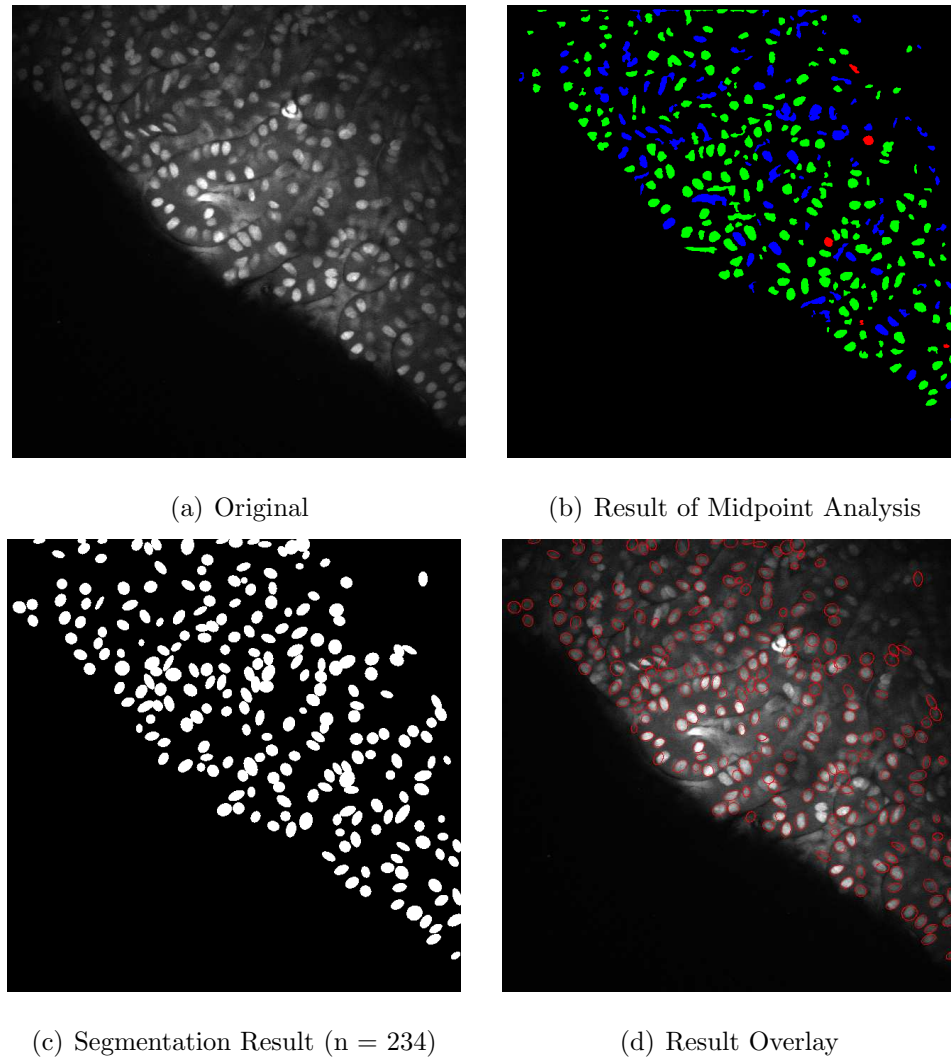
Fig. 5.8.: Segmentation results: *K-V*

the right. The images contains non-uniform brightness and noise, yet our method is able to successfully segment most nuclei.

As discussed in Chapter 4, objective evaluation of our segmentation results proves to be difficult due to the lack of ground truth data, for which the true shape and position of each object in the volume is known [175]. In fact, ground truth is impossible to obtain in fluorescent microscopy, since both the shape and position of an object are fluid in living animals, and are inevitably altered in the process of isolating and

(a) Original

(b) Result of Midpoint Analysis

(c) Segmentation Result (n = 204)

(d) Result Overlay
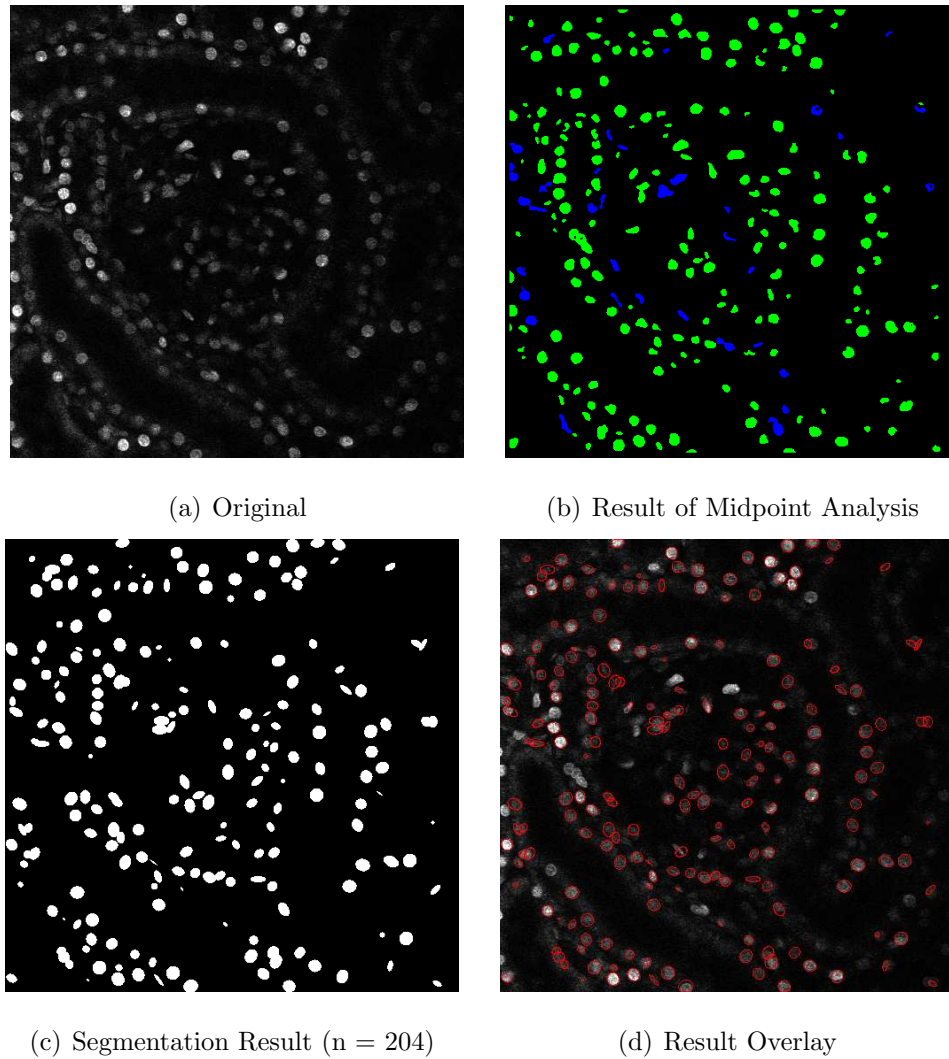
Fig. 5.9.: Segmentation results: *K-VI*

fixing tissues. Hand-segmentation for cell nuclei is extremely tidious in the first place and shape characterization of each of the nuclei is practically impossible.

Figure 5.16 compares our method with the method described in [249] which we denote as $\mathcal{M}_{des}$. We used the same object model and MPP parameters for $\mathcal{M}_{des}$ as that of our method. Note that $\mathcal{M}_{des}$ is applied directly on the original image without any preprocessing such as adaptive thresholding. Also, in $\mathcal{M}_{des}$ the energy term (H) consisted of the sum of only H_{Object} and H_{Inter}. Segmentation results (highlighted in red) from both methods are overlaid on the original images. Method $\mathcal{M}_{des}$ segments

(a) Original

(b) Result of Midpoint Analysis

(c) Segmentation Result (n = 184)

(d) Result Overlay
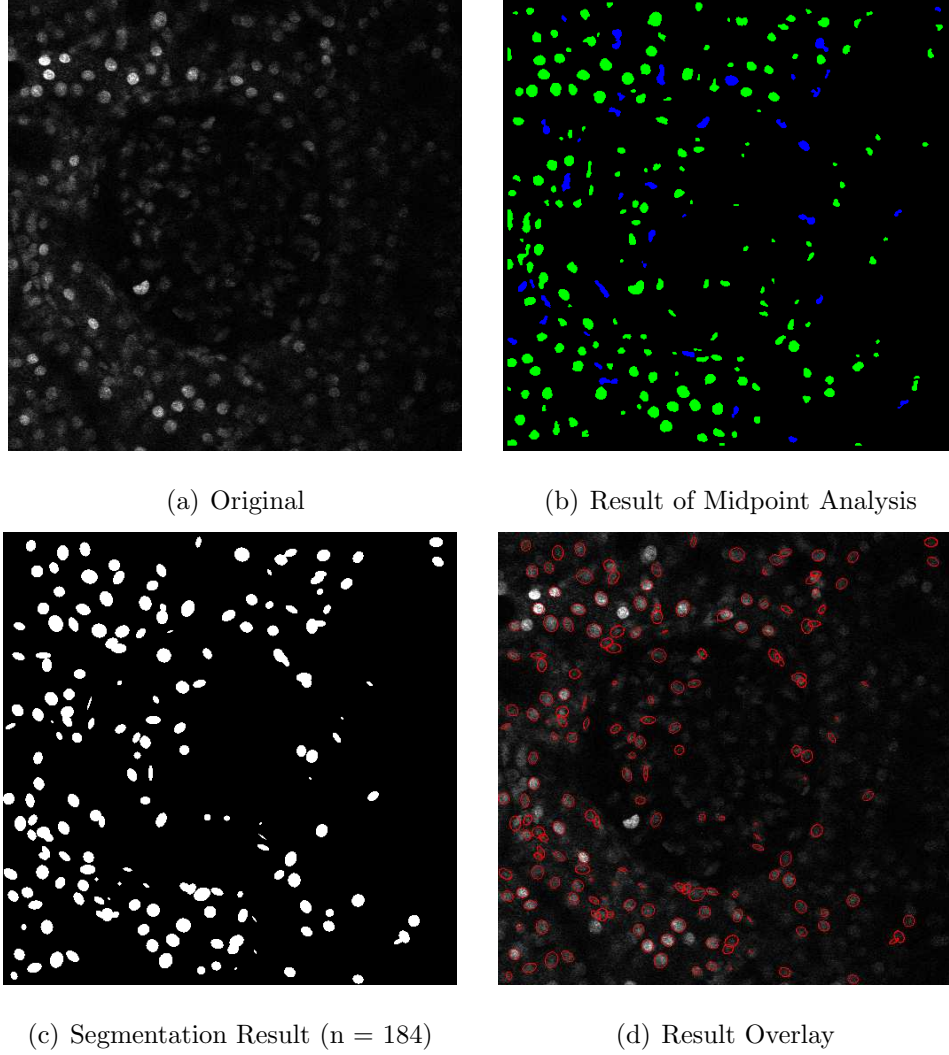
Fig. 5.10.: Segmentation results: *K-VI*

some nuclei at the center correctly. However, it failed to detect many nuclei from the less brighter regions , especially near the boundaries of the image. It also segmented many nuclei at places where there is no nucleus present visually. Our method provided a better segmentation detecting more nuclei correctly. In terms of computational time, our method takes on an average 20 times less than method $\mathcal{M}_{des}$ on the same machine. This is mainly because $\mathcal{M}_{des}$ computes the energy functions at every pixel of the image as against the selective MPP treatment performed in our method.

(a) Original (b) Segmented Result (c) Result Overlay

Fig. 5.11.: Segmentation results: *L-I*

(a) Original (b) Segmented Result (c) Result Overlay
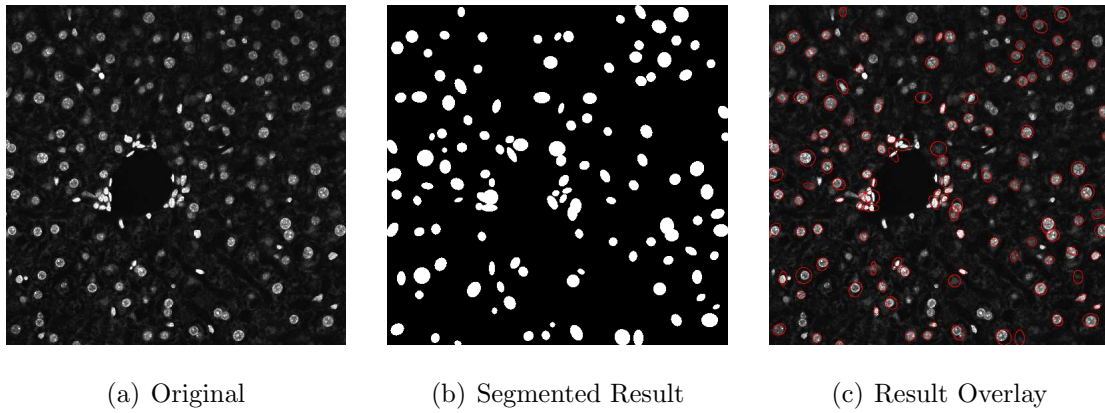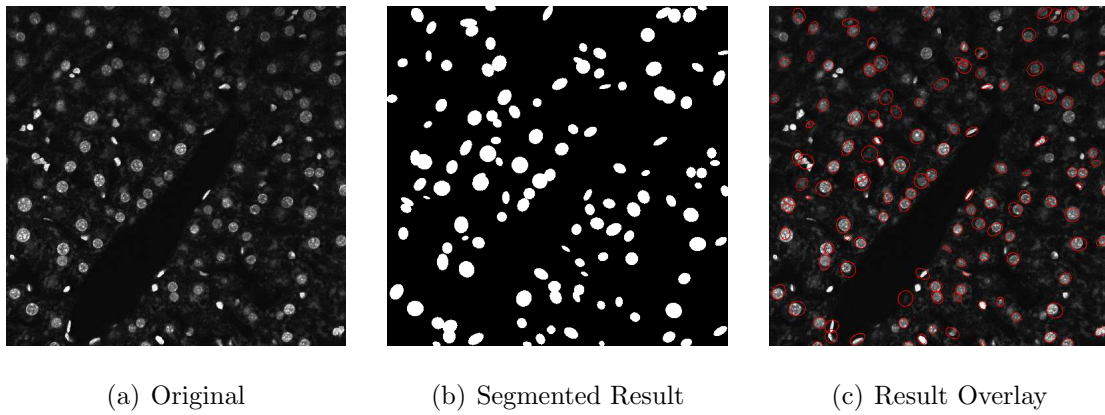
Fig. 5.12.: Segmentation results: *L-II*

In conclusion, our proposed method successfully segments nuclei, enabling their counting and shape characterization.

Next, we show examples of the combined results of our jelly filling method from Chapter 4 and the results of our nuclei segmentation method from this section. Figure 5.17 and Figure 5.18 are kidney images and Figure 5.19 - Figure 5.23 are liver images that we used for jelly filling and nuclei segmentation. Each figure contains (a): the original red channel image, (b): the result of our proposed jelly filling segmentation method, (c): the original blue channel image, (d): the result of our nuclei

(a) Original   (b) Segmented Result   (c) Result Overlay

Fig. 5.13.: Segmentation results: *L-III*

(a) Original   (b) Segmented Result   (c) Result Overlay
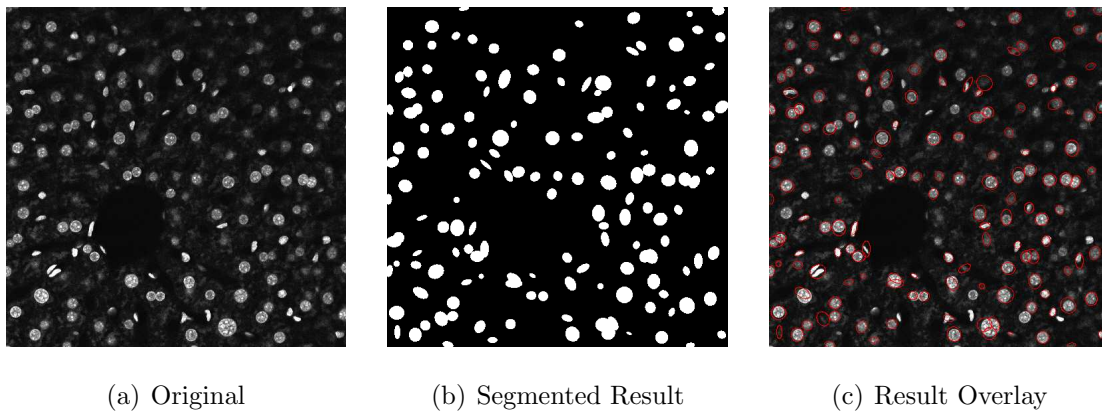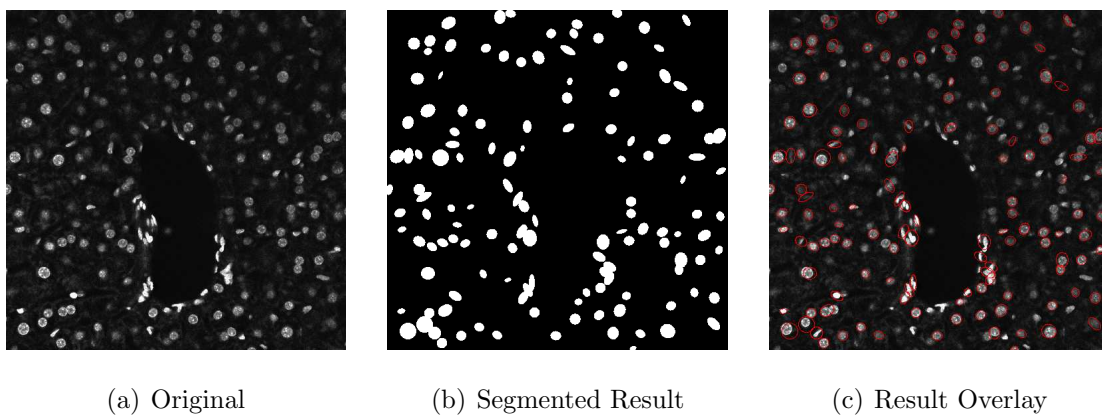
Fig. 5.14.: Segmentation results: *L-IV*

segmentation method, (e): the combined results of (b) and (d), (f): the red and blue channels of (e) representing the boundaries and nuclei respectively.

The results indicate that our automatic image analysis methods can be used as an effective tool in biomedical research.

(a) Original   (b) Segmented Result   (c) Result Overlay
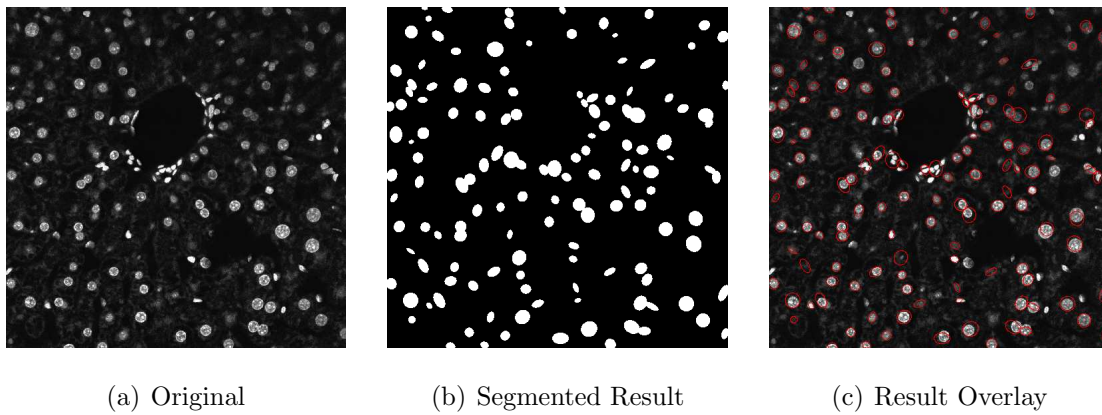
Fig. 5.15.: Segmentation results: *L-V*

(a) Method from [249] ($n = 241$)   (b) Our Method ($n = 628$)
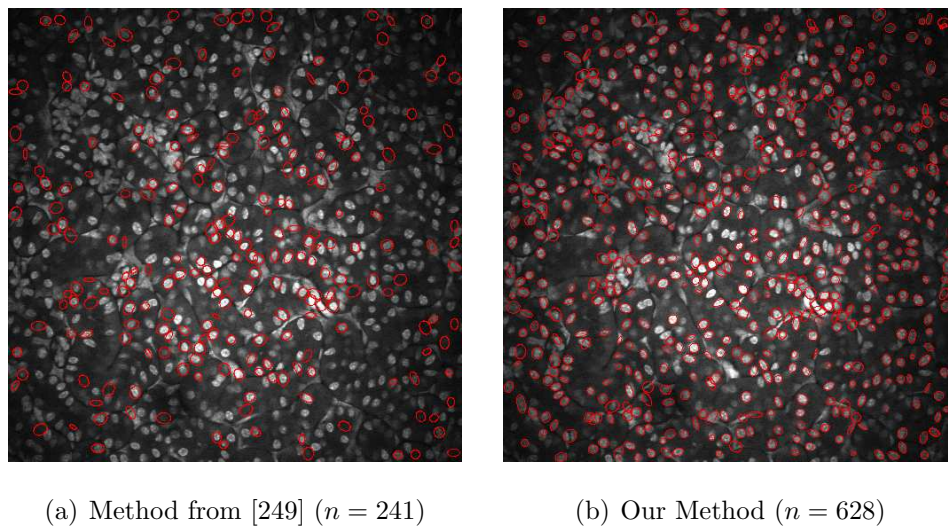
Fig. 5.16.: Comparison of the segmentation results. Outlines of segmented ellipse marks are represented by red and overlaid on the original image.

(a) Original Red Channel

(b) Jelly Filling Segmentation Result

(c) Original Blue Channel

(d) Nuclei Segmentation Result

(e) Combined Segmentation Result

(f) Combined Segmentation Result

Fig. 5.17.: Segmentation results of our proposed methods: (*K-I*)

(a) Original Red Channel

(b) Jelly Filling Segmentation Result

(c) Original Blue Channel

(d) Nuclei Segmentation Result
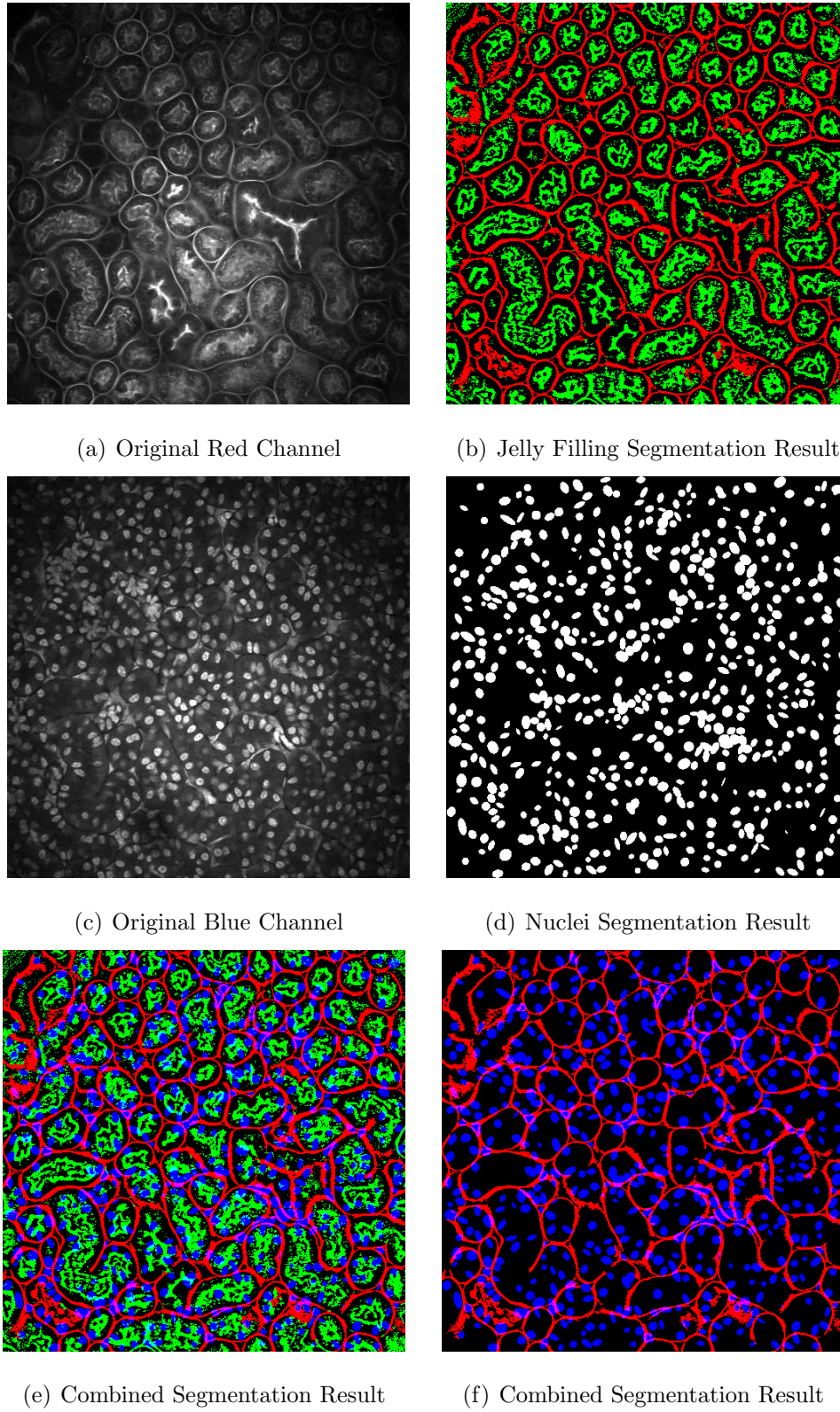
(e) Combined Segmentation Result

(f) Combined Segmentation Result
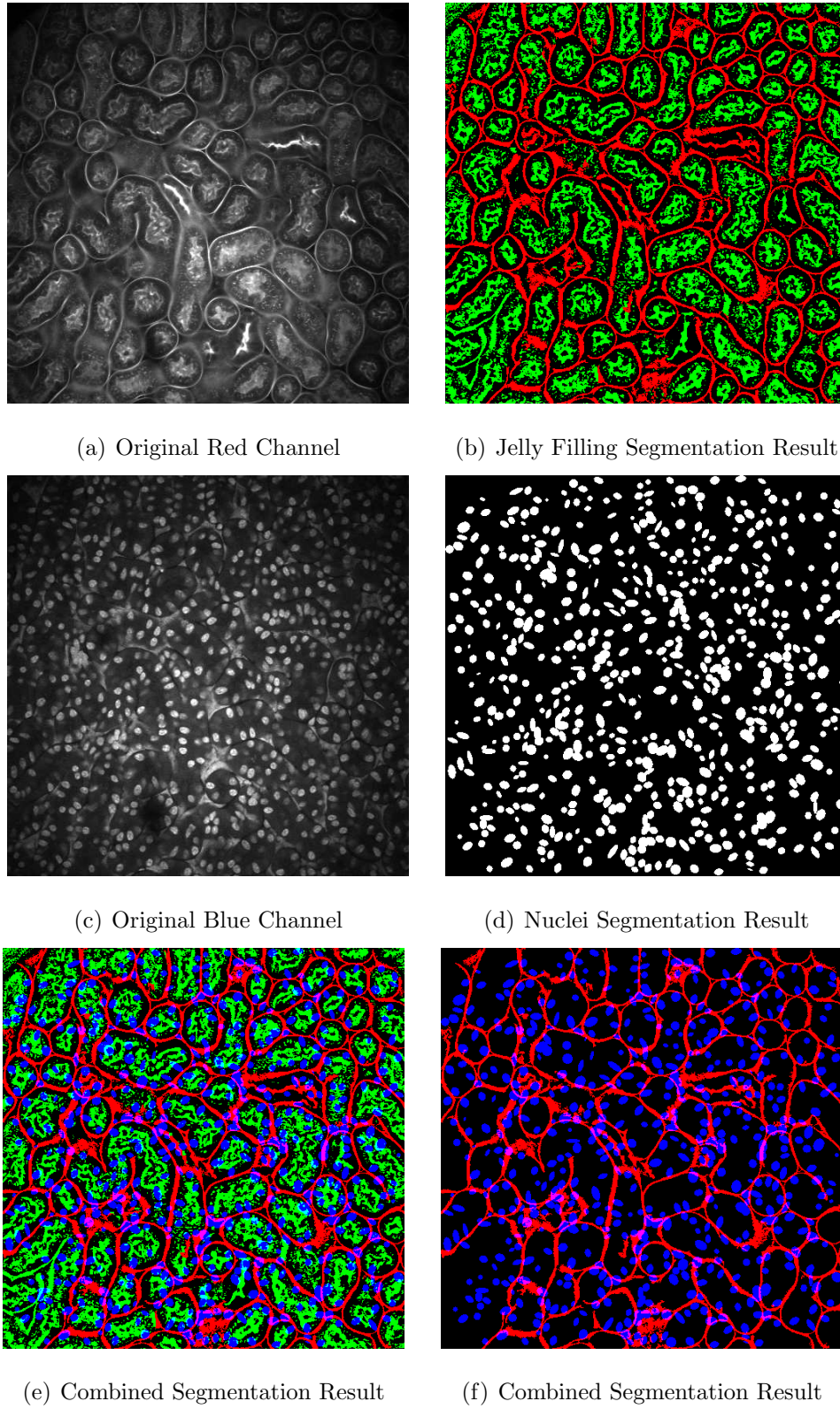
Fig. 5.18.: Segmentation results of our proposed methods: $(K\text{-}I)$
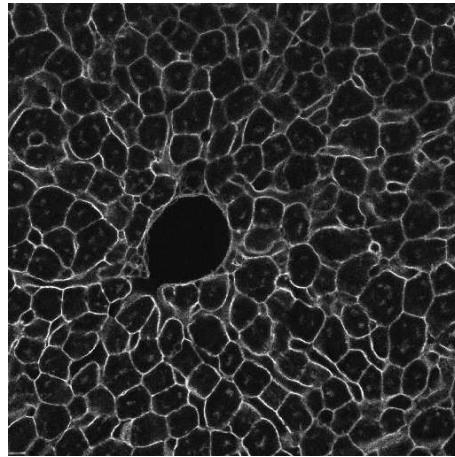
(a) Original Red Channel
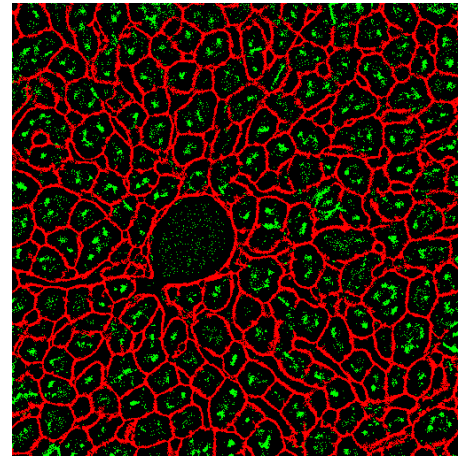
(b) Jelly Filling Segmentation Result
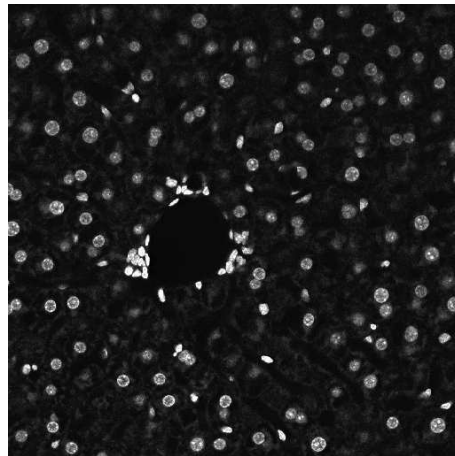
(c) Original Blue Channel
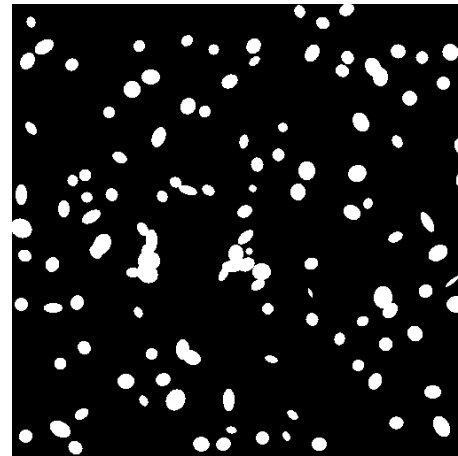
(d) Nuclei Segmentation Result
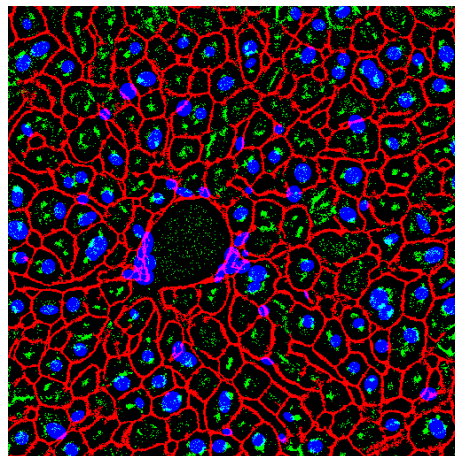
(e) Combined Segmentation Result
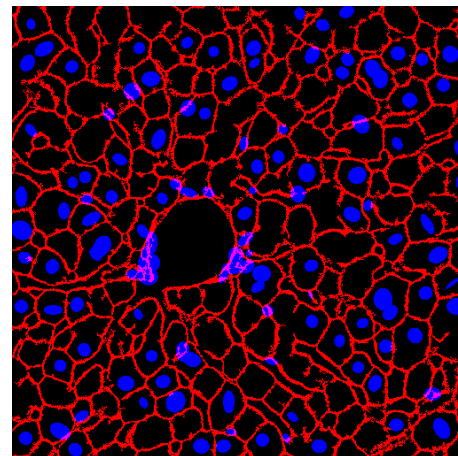
(f) Combined Segmentation Result

Fig. 5.19.: Segmentation results of our proposed methods: (*L-I*)

(a) Original Red Channel

(b) Jelly Filling Segmentation Result

(c) Original Blue Channel

(d) Nuclei Segmentation Result

(e) Combined Segmentation Result

(f) Combined Segmentation Result

Fig. 5.20.: Segmentation results of our proposed methods: (*L-II*)

(a) Original Red Channel

(b) Jelly Filling Segmentation Result

(c) Original Blue Channel

(d) Nuclei Segmentation Result
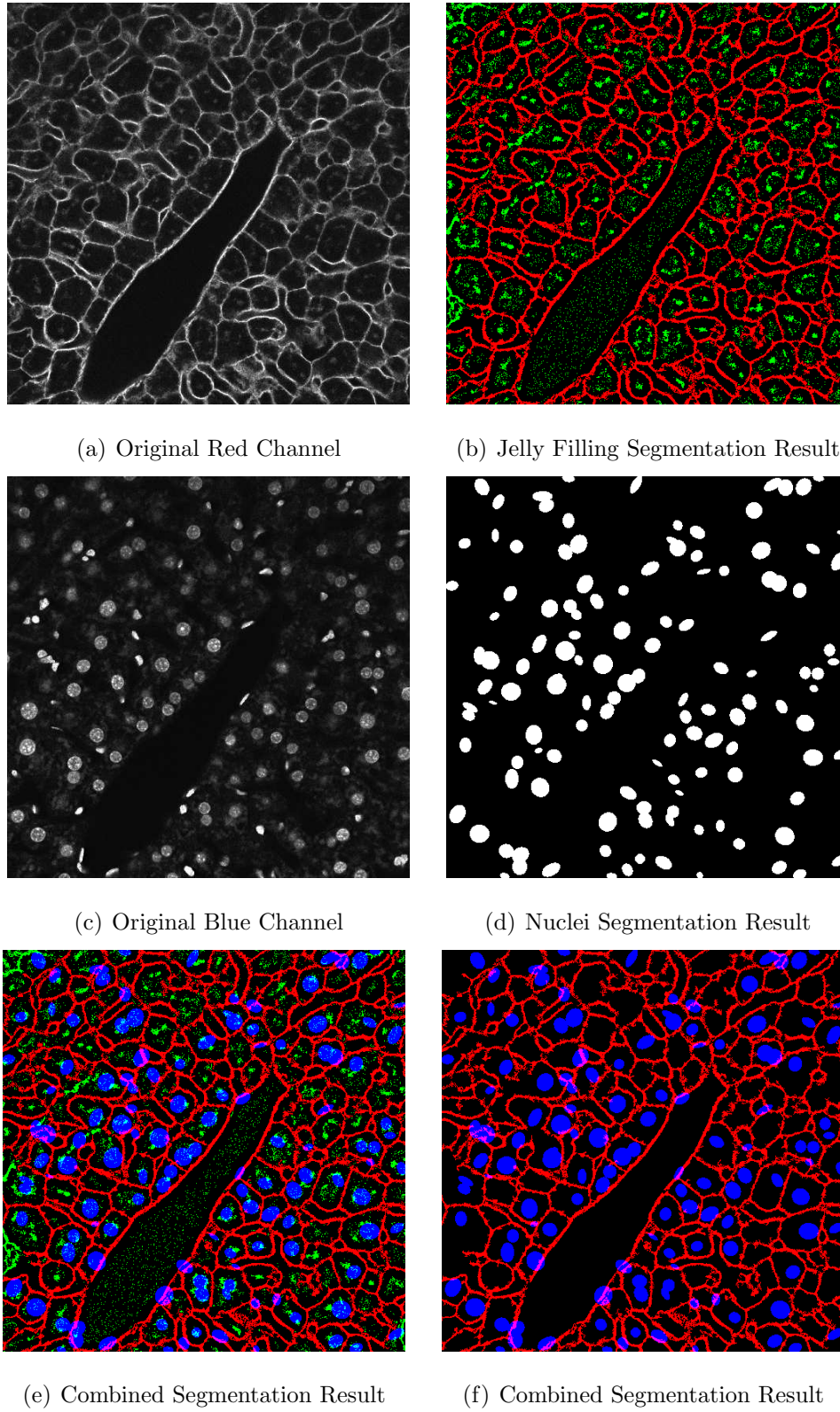
(e) Combined Segmentation Result

(f) Combined Segmentation Result
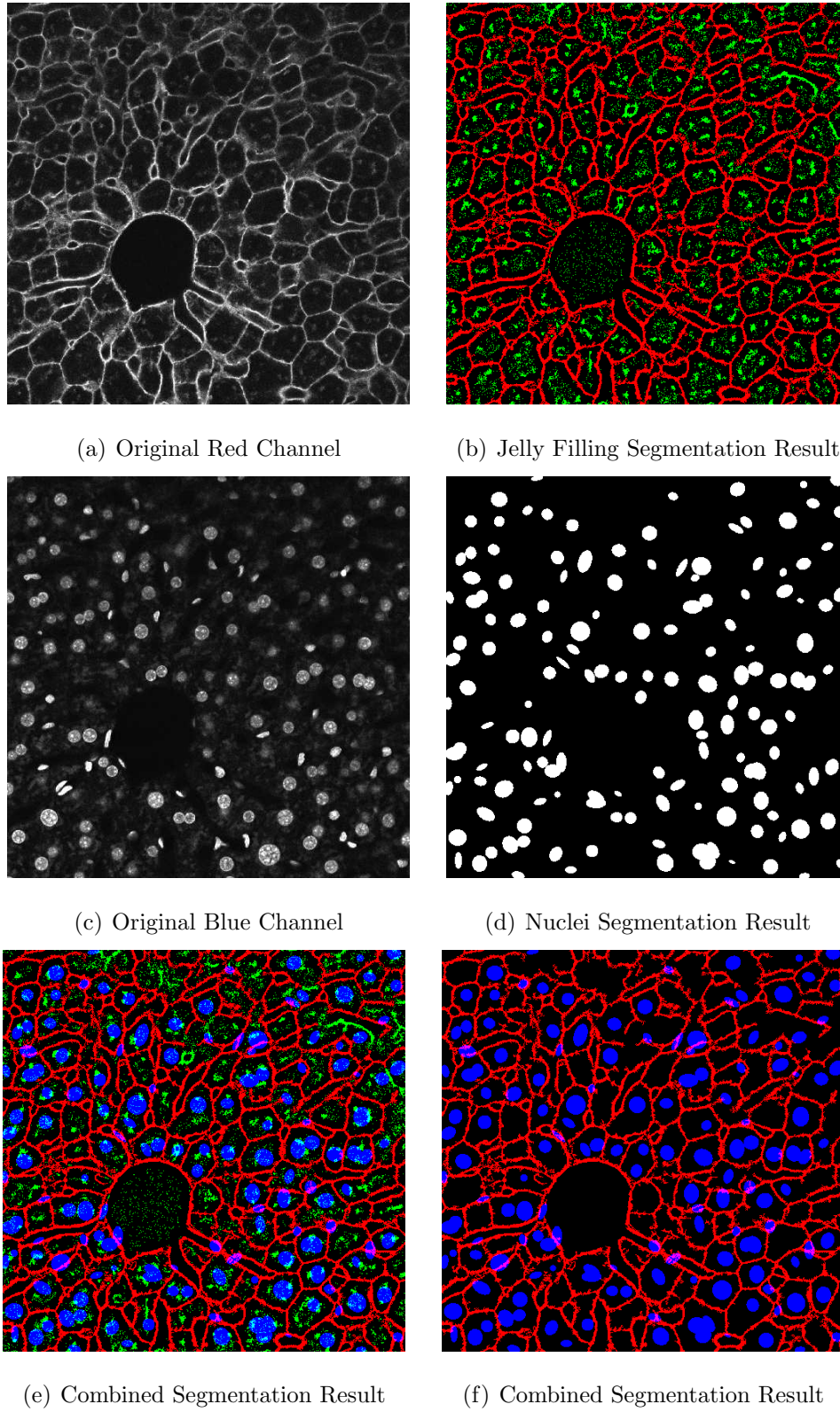
Fig. 5.21.: Segmentation results of our proposed methods: (*L-III*)
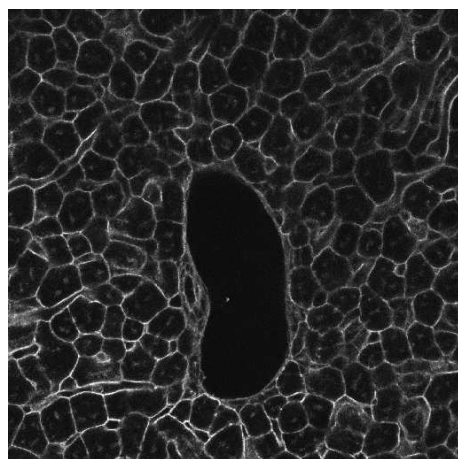
(a) Original Red Channel
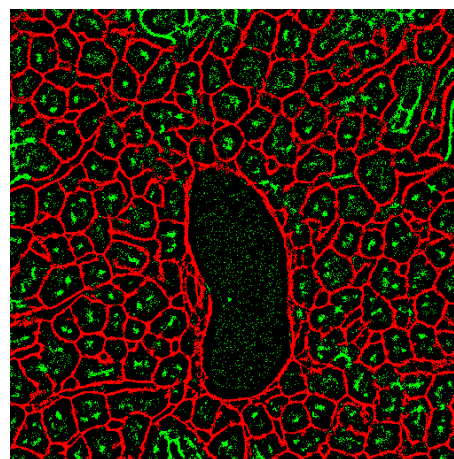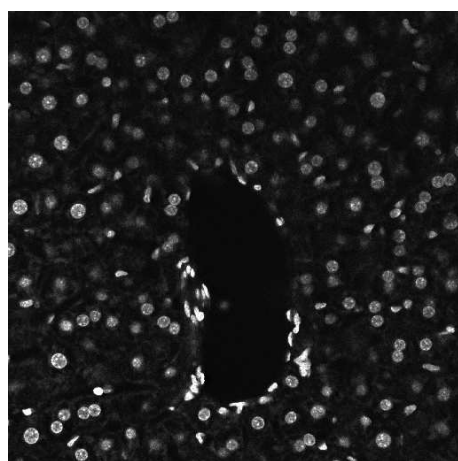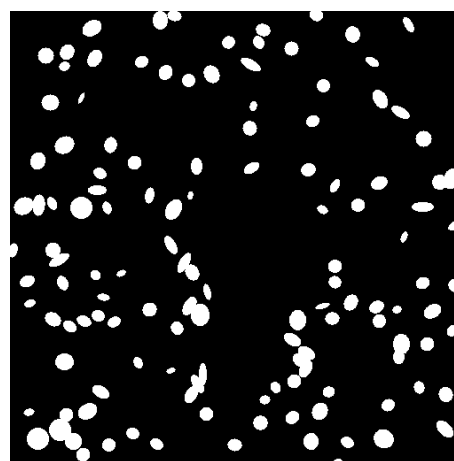
(b) Jelly Filling Segmentation Result

(c) Original Blue Channel

(d) Nuclei Segmentation Result
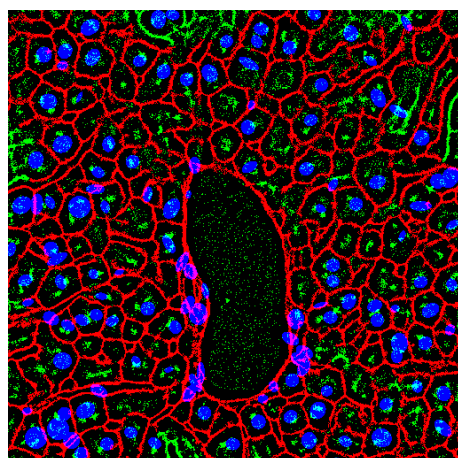
(e) Combined Segmentation Result
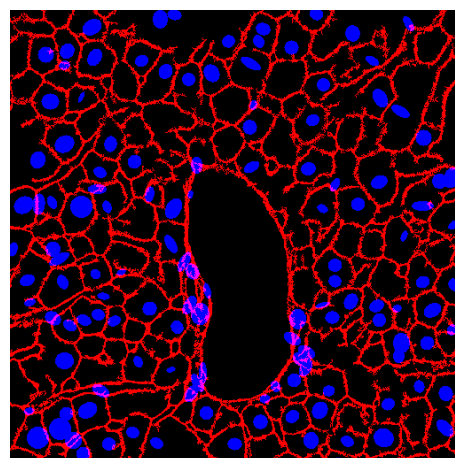
(f) Combined Segmentation Result

Fig. 5.22.: Segmentation results of our proposed methods: (*L-IV*)
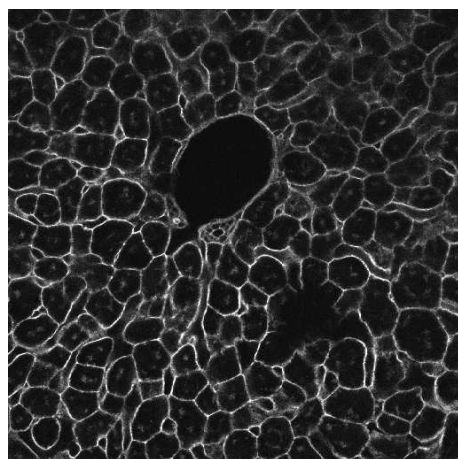
(a) Original Red Channel
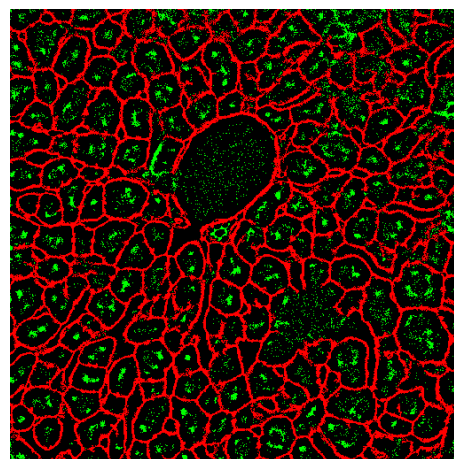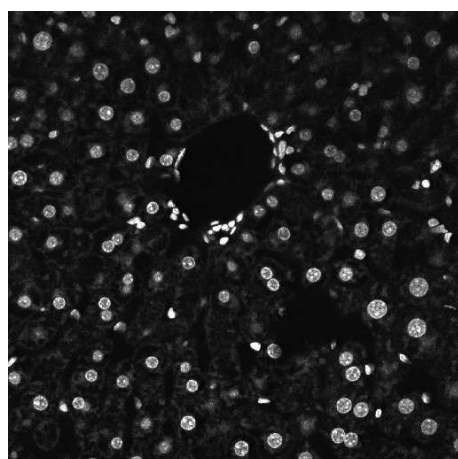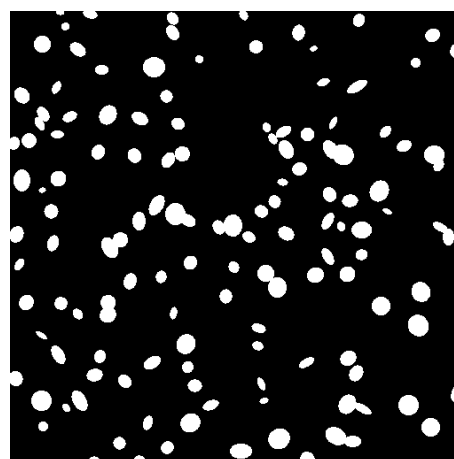
(b) Jelly Filling Segmentation Result

(c) Original Blue Channel

(d) Nuclei Segmentation Result
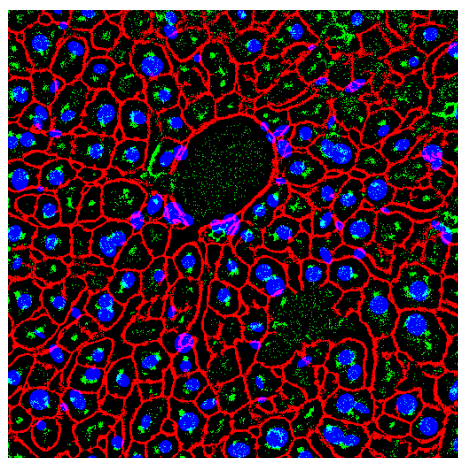
(e) Combined Segmentation Result
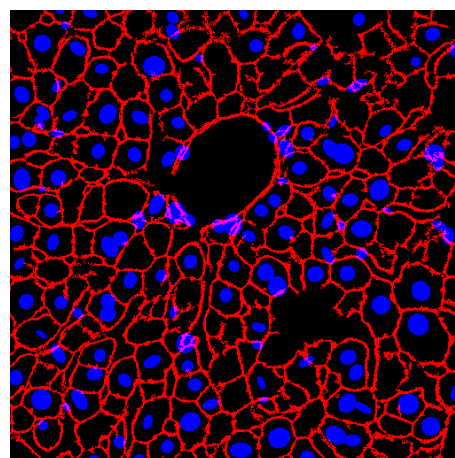
(f) Combined Segmentation Result

Fig. 5.23.: Segmentation results of our proposed methods: (L-V)

6. A WEB-BASED VIDEO ANNOTATION TOOL FOR CROWDSOURCING SURVEILLANCE VIDEOS

Our goal is to provide law enforcement with a web-based video annotation system capable of doing rapid analysis of surveillance video using crowdsourcing methods. This analytic tool should aid in identifying potential threats and help investigate crime. The same platform is also useful for future integration of machine learning techniques where crowdsourced annotations can "help" automatic detection system to improve their performance [256]. [1]

We define a set of requirements to achieve our goal next.

6.1 System Requirements

The requirements are categorized into four main areas.

System Management and Data Security:
Because of the nature and ownership of the video for our law enforcement application, there are requirements on the access and the management of the system. Data and servers are not to be hosted on or integrated with commercial platforms. Access to the system, training modules and tasks are managed by the designated law enforcement entities. The crowd (annotators) must also be properly vetted by law enforcement. Access for workers/annotators should be limited only to the assigned tasks. Members of the crowd can annotate assigned videos with event labels that are assigned to them. Interaction between different members of the crowd should be

---

limited to the crowd model defined by law enforcement.

Video Annotation Tool:

Members of the crowd (annotators) should be able to identify time intervals in which they detect an event. Each event should be categorized using the designated labels. Spatial annotations can be done in the form of rectangular boxes capturing the object(s) of interest. Comments can be added to each annotated event.

Activity Alerting and Result Aggregation:

Workers/annotators should be provided with real-time video streams for real-time alerting or recorded video for "after the event" or forensic analysis. Results of the annotations should be available for law enforcement to examine and track the performance of the annotators. They can be sent to an individual worker or a group of workers for validation, training or other purposes required by the task.

Worker Management:

Workers' profile information should be kept private, protecting their passwords, identity and contact details. The system should be able to keep track of the amount of work done by each worker with some type of performance measure. Their training and actual annotation task status should be visible with the capability of sending emails via the system to the appropriate law enforcement management. The system should be easy-to navigate and organized in terms of the information available.

With this set of broad requirements, our web-based system is described below [256].

6.2 Our Approach

Our video annotation platform is based on a client-server approach where a central authority consisting of designated law enforcement manages the entire system. This ensures that it is the only entity to give access to surveillance video, assign annotation

tasks and manage the workers. We define the two main entities as an essential part of the process:

- Central authority: Full access to manage and architect the crowd (system management authority such as law enforcement)

- Members of the crowd: Similar to clients with limited access to the system (Workers or Annotators)

Apart from the entities defined above, the management authorities can assign members of the crowd various "roles." This capability is essential in managing the crowd and creating a hierarchical model. Roles are used to distinguish members of the crowd into several levels and limit the set of event labels with which they can annotate the video. The capabilities of our system and typical workflow is described in detail below.

Our video annotation system is currently deployed on a server at Purdue University. The server is an Intel Xeon processor (3.20GHz) with 32 GB RAM and 8TB of HDD storage. It runs the Linux operating system (Ubuntu server 12.04), Apache HTTP server (current version: 2.2.22) and PostgreSQL database system(current version: 9.1.11).

Figure 6.1 shows the system architecture. A popular open-source video tool, FFmpeg is used to handle video processing functionalities. Java (with OpenCV library) is used for the implementation of image processing and computer vision functionalities. PHP is used for server scripting and creates dynamic web page content. Web pages generated use HTML5 elements (currently supported by the Google Chrome browser) such as <video> and <canvas>, both are very useful for handling embedded multimedia objects. Embedded in the webpages are JavaScripts that are designed to run on the client side and produce used-friendly interactive features. This enables the crowd to add annotations in a seamless way.

System access is divided into two categories: First-time users and Non-first time users. Administrators can invite users to join the system via an email invitation
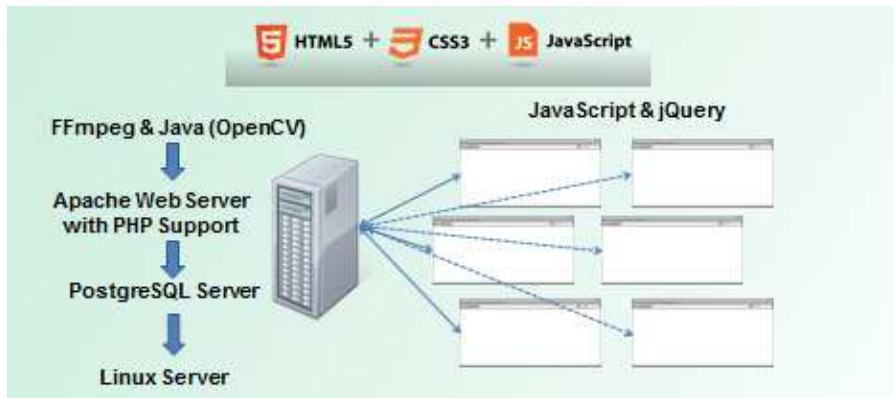
Fig. 6.1.: Our crowdsourcing system architecture

generated by the system. New users can be invited from administrator's portal as described in the Section 6.2.1. A unique 6-digit numeric identifier is generated for each worker whereby they are sent an invitation email along with a one-time password. New members can log in to the system using this identifier and the one-time password. The users are prompted to fill out a form asking for their profile information including new password, address, availability in number of hours/week and a profile picture. Access to any other pages is denied until they fill out this information. Once they submit the form, the new password is is generated and stored in the database.

Returning users can access their account with the unique 6-digit username and password. Usernames (identifiers) and passwords are stored in a PostgreSQL database table sequentially.

6.2.1 Administrator Portal

This portal consists of functionalities developed for law enforcement management of the system.

Users Administration: Administrators can invite new workers by sending an email. They can view users' profile information such as address, profile picture and availability in number of hour/week. They can send emails to workers and block a worker's access to the platform.

Administrators can assign a "role" to a user. A role is useful in allowing or denying access to certain types of annotations. A role is defined by a set of labels corresponding to various events representing specific criminal activities or potential threats such as battery, abandoned baggage, assault. Administrators can create a new role based on a set of associated labels. They can also define a new label with its description and training attributes (described below). When an Administrator assigns a specific role to a set of workers, they can provide annotations only for the labels associated to it. An example of a role is an "Baggage Expert" which consists of labels: Abandoned baggage, suspicious swapping of a bag and visibly suspicious baggage contents. This role allows workers to annotate a suspicious activity of that type by using the associated labels.

Figure 6.2(a) and 6.2(b) represent snapshots of the system interface for managing users and roles. Figure 6.4(b) displays the work flow associated with creating a label or a role.

Training Management: Training is an important approach to improve the quality of annotation [257]. A trained crowd has been shown to perform better than an untrained crowd [258]. In our system, training is associated with an annotation label. Creating training modules specific to an annotation label ensures that workers are trained to annotate videos using that label. When the workers pass training for a label, they are qualified to do actual annotations corresponding to that event.

At the time of creating a new label, Administrators can upload teaching (demo) and training videos. They are prompted to provide their annotations as "ground truth answers" so that the training done by workers can be assessed to make a pass/fail decision. Training videos can be categorized using options such as "easy," "medium" and "hard" based on their level of difficulty. Administrators have a choice to make

(a) *User administration.*

(b) *Role administration.*

Fig. 6.2.: Users and roles administration

the training optional or mandatory for a label. A teaching video includes explanation and instructions on when to annotate a video with this specific label. Figure 6.4(a)

displays the work flow associated with creating a training module and user management.

Administrators can assign specific training modules to workers and review the results. If the results are satisfactory, they are qualified for doing the actual tasks, otherwise a new module is assigned to them. After many unsuccessful attempts, a worker may be denied access to that label or the entire system. Creating training modules for a specific label ensures that chosen members of the crowd possess the required understanding of this type of events. It also helps in creating a hierarchal model. The officer uploads a training video and annotates it to provide the ground truth results. A teaching video can also be uploaded and includes explanation and instructions on when to annotate a video with this specific label. Figure 6.4(a) displays the work flow associated with creating a training module and users management.

Task Management: Managing annotation tasks is the core of our system in which administrators can assign real-time streams or recorded content to specific workers. For real-time streams, administrators needs to specify the details of the camera, such as its make, type and IP-address. Currently the system supports streaming of unprotected streams, but in future more security features will be added. Administrators can assign a particular camera feed to a set of workers. For recorded content, administrators first upload the video (.mp4 format) to the server via their task management portal. Then they have an option of editing the video by cutting it into segments of smaller durations, adding a brief description of the contents and the goal of overall analysis. Then these segments can be assigned to specific workers from a list of available workers. A timestamp of the task assignment is recorded. A list of videos is displayed along with their respective status such as: "Segments created," "Tasks assigned," and "Tasks completed" with the timestamps of the corresponding actions. A typical workflow of task management is illustrated in Figure 6.4(c).

Result Aggregation and Alert Reporting: When workers submit their annotations, they appear as alerts on the administrator's portal. Administrators can also view all the results by checking completed tasks from a drop-down menu in the task

management. The temporal annotations, corresponding to a suspicious event along with specific textual comments, are aggregated in the form of the entire video with annotations and also with cut segments of the actual annotation time duration. For quick viewing, the administrator can only view the cut segments if desired to save significant time by not watching the entire video.

6.2.2 Annotator Portal

The annotator portal is for workers to log in and do the annotation tasks. They can view and modify their profile information. They can the check status of the training tasks assigned to them. Status of each task is displayed e.g., "Assigned but not started," "Started but not complete," "Complete and under review" and "Pass" or "Fail." Similarly, they can view the actual annotation tasks assigned to them where the status can be: "Assigned but not started," "Started but not complete" or "Submitted." Workers are not notified concerning how the results of their submitted tasks are used by the administrators.

Our system is designed to provide easy-to-use tools with click-able interfaces for doing annotation tasks. When workers start a new task, the corresponding video content is displayed with usual video player functions such as play, pause, forward, reverse and volume control. Displayed below the video contents is a list of sliders representing specific labels corresponding to their roles. The workers can create a new annotation by selecting a slider with a label and clicking on the video content. This creates a highlighted time interval on the slider. Both ends of the interval can be dragged to change the duration. Spatial annotations and textual comments can be added by simple click functions. The workers have an option to either "Save and continue later" or "Save and submit" a task. In the prior case, the annotations are saved and displayed as they were the next time the worker resumes doing annotations. In the later case, annotations are sent to system. Figure 6.3 shows an example of the annotation interface for an annotator with 7 different labels.

After doing annotations, a worker can specify number of hours spent on that particular task. This is added to the total number of hours spent by that worker. A typical annotator's workflow is illustrated in Figure 6.4(d). Members of the crowd can log in the system using their ID and check assigned tasks and training. For both of those, the main component is the annotation platform using which they can annotate videos with the labels defined in their corresponding role. For each label in their role, a slider appears below the video player. Annotators can add highlighted time intervals on sliders independently. Each highlighted time interval represents an annotation using the corresponding label. Textual comments and spatial annotation are also possible for each annotation.
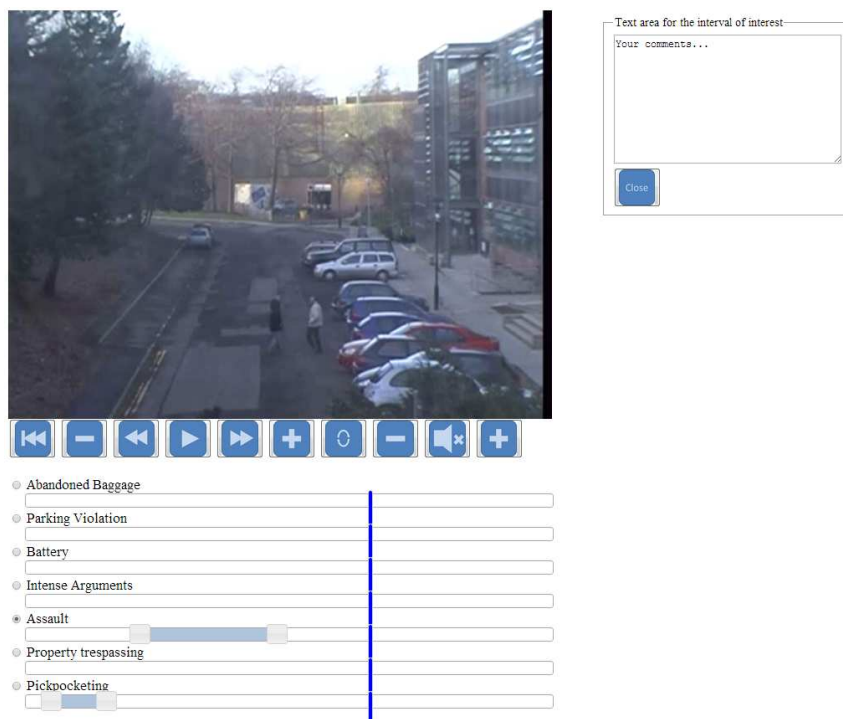
Fig. 6.3.: The annotation interface
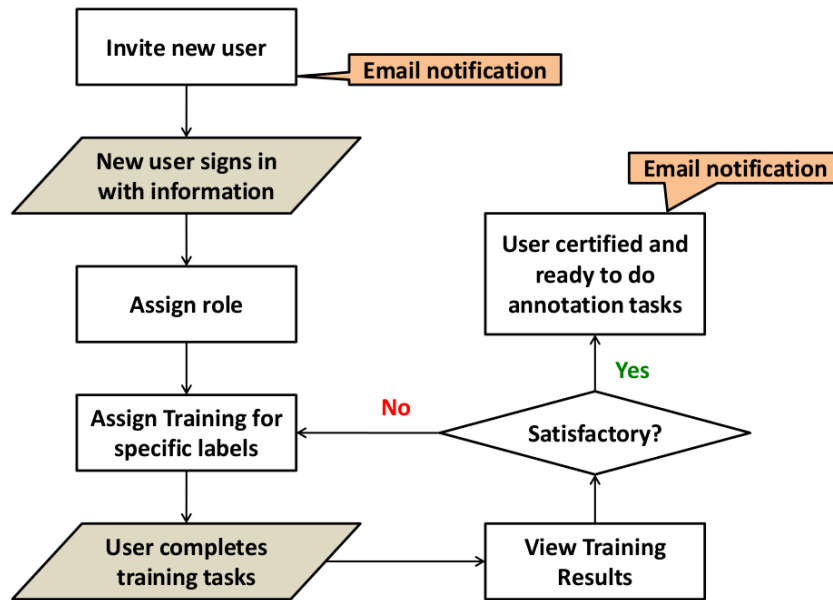
(a) *User management*
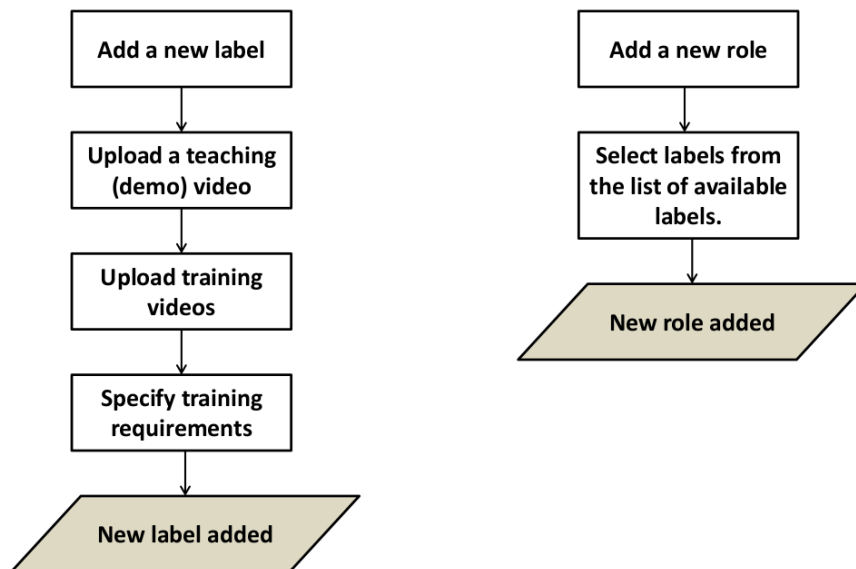
(b) *Labels and roles*
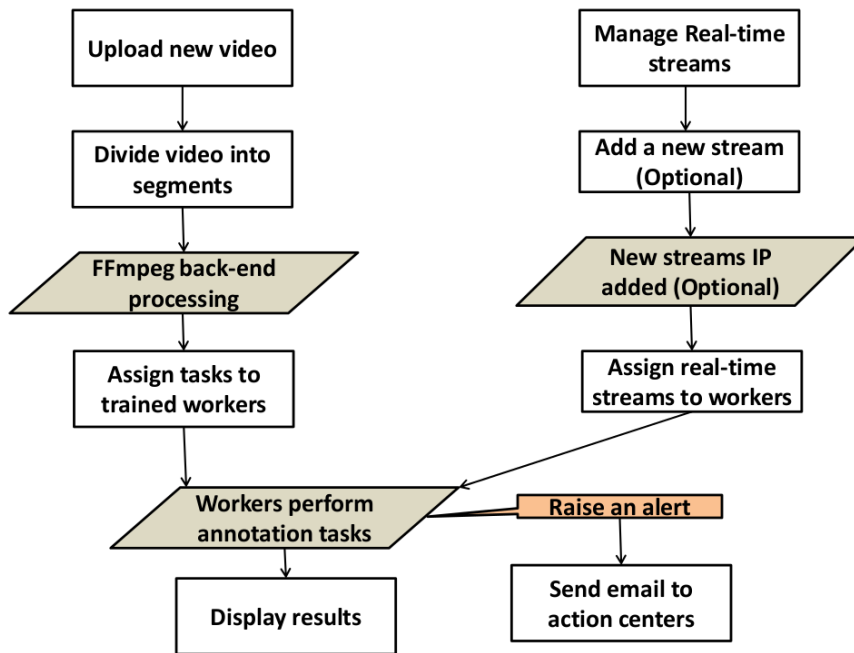
(c) *Task management*
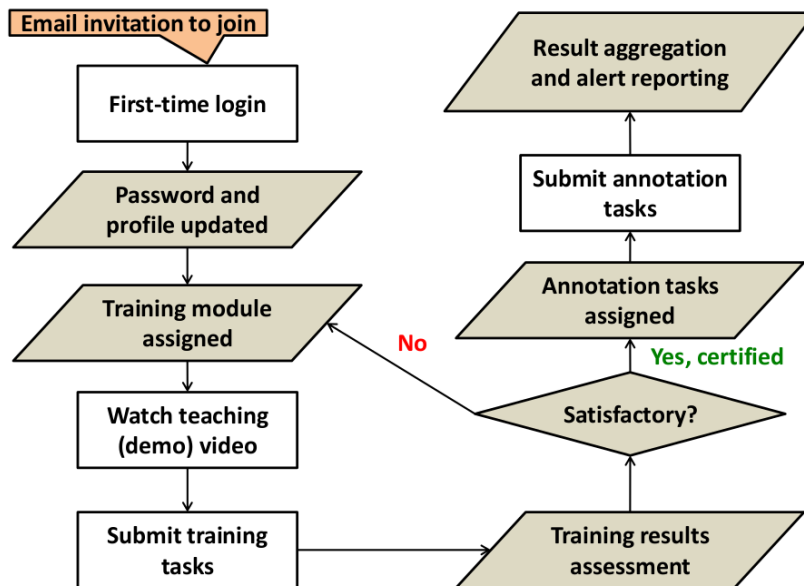
(d) *Annotator workflow*

Fig. 6.3.: Typical system workflow

6.3   Experimental Results

In this section, we present experimental results of our system implementation. For our experiments we used the following publicly available surveillance video datasets: BEHAVE [259], Lunds University traffic dataset[2], UCR-VW [260], PETS 2006, PETS 2007 series datasets and i-Lids dataset for AVSS 2007. We transcoded the videos using FFmpeg into .mp4 format to make it compatible with HTML5.

6.3.1   System Performance

We first test the overall performance of the system, we present precision and recall values for various event types. An "event" is a potential threat or a crime. In this context, precision is the ratio of the number of detected events to the total number of annotations. Recall is the ratio of the number of detected events to the total number of events. Ideally, we would like precision and recall both to be 1. However, this may not be always possible considering ambiguities, occlusions and other difficulties. More importantly, we want recall to as close to 1 as possible, because it represents the number of correctly detected events in the context of true events. Larger recall means fewer missed detections. An event being undetected is worse than having a false indication.

Our experiments used 15 annotators that were assigned surveillance videos containing 41 events. Table 6.1 summarizes these results.

One primary observation is that the performance varies according to the labels. Based on the table, for suspicious bag activity, both precision and recall performance is very good. For intense arguments and battery, recall is 1. For traffic violation, precision and recall are close to 0.5. This indicates that traffic violation events are harder to detect than other labels. Battery and intense arguments events seem more obvious

---

[2]Dataset available from the Video analysis in traffic research project, Lunds University, funded by The Swedish Governmental Agency for Innovation Systems (VINNOVA).

Table 6.1: Training video results

Event Type/Label	Precision	Recall
Suspicious bag activity	0.93	0.78
Traffic violation	0.45	0.56
Intense argument	0.67	1
Battery	1	1

to the crowd and hence more accurately detected. Overall, our system demonstrates a good performance in terms of detecting events for. In the next section, we investigate how training can improve the detection performance.

6.3.2 Trained Vs. Untrained Crowds

The basic intuition behind this comparison is to see the impact of the training process on the annotation performance. A guided training process enhances one's ability to perform tasks. Another important aspect of this process is the ability to identify low performing annotators and avoid assigning them tasks in the first place. Our training module is based on actual surveillance video and it is very likely that annotators who performed poorly would also perform poorly in the assigned tasks. In [228], it has been shown that annotating video is one of the tasks that require high cognitive demand, it was also recommended to identify high performing annotators, which we believe our training module achieves.

We compare the performance of trained annotators against the untrained annotators. Eight annotators are divided randomly into two groups, the training group: the one to be trained first and then assigned tasks, and the non-training group: the others to be assigned the same tasks without undergoing the training. The training process is started by assigning a video to each annotator from the training group.

The training video is accompanied by an audio track that contains a definition of the event and an explanation on how to recognize it, without revealing the actual event.

Two labels are used for training purposes: Suspicious bag activity which includes baggage swap and abandoned baggage, and traffic violation which includes a dangerous turn. Two sets of training videos are created i.e. two for each label. First, *Set 1* was assigned to the training group. Administrators evaluated the annotation results of the training videos submitted by the training group and compared them to "ground truth." They made a decision and passed or failed the workers. Only the workers who failed the training with *Set 1* were assigned *Set 2*. The results of that training were recorded. The annotators who passed the training in either *Set 1* or *Set 2* were assigned the actual tasks. The ones who failed both training sets were not assigned any tasks for that particular label.

Table 6.2 summarizes the training results. All of the 4 annotators passed the bag suspicious activity training module. Out of them, 3 passed the training in the first attempt and 1 passed in the second. For traffic violation, 3 passed in the first attempt, while 1 failed the training.

Table 6.2: Training performance

Label	Attempt I (*Set 1*)		Attempt II (*Set 2*)		Total Passed	Pass Percentage
	Assigned	Passed	Assigned	Passed		
Suspicious bag activity	4	3	1	1	4	100%
Traffic violation	4	3	1	0	3	75%

After the training process was completed, we assigned trained and untrained annotators to the actual tasks to detect events from three videos. Table 6.3 summarizes the videos and events information.

Table 6.3: Task performance

Video	Duration (seconds)	Label	Event Description
Video 1	01:56	Suspicious bag activity	Bag swap in a busy airport hall
Video 2	02:59	Suspicious bag activity	Abandoned bag in a busy airport hall
Video 3	01:53	Traffic violation	Failing to yield to other vehicles when turning

Table 6.4: Final results

Event type	Trained group			Untrained group		
	# Annotators	Precision	Recall	# Annotators	Precision	Recall
Suspicious bag activity	4	0.89	1	4	1	0.5
Traffic violation	3	0.5	0.67	4	0.2	0.25

Table 6.4 is a performance comparison between the two groups. For the trained group, the recall for suspicious bag activity was 1, whereas for untrained crowd it was 0.5. In case of traffic violation, the trained group had a recall of 0.67, significantly higher than that of untrained crowd (0.25). Precision for traffic violation is better in case of trained group than that of untrained group. However, for bag suspicious activity, trained group precision was slightly worse than untrained group.

The results indicate that training process has significantly improved recall values. In other words, the training process decreased the number of missed events (false negatives). Members of the crowd having a better understanding of suspicious events are better able to identify them. Missed events are possible threats to public safety and it's very important to minimize those as much as possible. We expect that when our system is deployed on a larger scale, training will help avoid missed events to a large extent.

7. CONCLUSIONS

7.1 Summary

In this thesis, we developed new methods for error resilient video coding, microscopy image segmentation and an implementation of a video annotation platform. The main contributions of the thesis are:

- Adaptive Error Concealment for Multiple Description Video Coding
  We propose two adaptive error concealment methods for a temporal-spatial four description multiple description video coding architecture. Our adaptive methods are motion vector analysis and error estimation using the H.264-coded MDC bitstreams. We propose another adaptive concealment method for a spatial-subsampling based MDC architecture. This method uses motion information and prediction mode extracted from HEVC-coded MDC bitstreams. Experimental results show that our proposed methods are effective under packet loss conditions during video transmission.

- Error Resilient Video Coding using Duplicated Prediction Information for VPx Bitstreams
  We describe an error resilient coding method for VPx-coded bitstreams using duplication of prediction information. Experiments indicate that our method provides a graceful quality degradation under packet loss conditions.

- Jelly Filling Segmentation of Biological Images Containing Incomplete Labeling
  We propose an iterative 3D segmentation method mainly for fluorescence microscopy images containing the incomplete labeling artifact. Intuitively, our method is based on filling the disjoint background regions of an image with "jelly-like" fluid such that the interactions between the "jellys" and the seg-

mented foreground can be used to separate different biological entities in 3D. Experiments with our images exhibit the effectiveness of our proposed method as against some existing methods.

- Nuclei Segmentation of Microscopy Images using Midpoint Analysis and Marked Point Process
  We present a cell-nuclei segmentation method based on midpoint analysis and a random process simulation. Midpoint analysis is used to classify the segmented regions into single-/multi-centered objects based on their shape properties. A 2D spatial point process simulation is then used to quantify cell-nuclei by their location and size.

- A Video Annotation Tool for Crowdsourcing Surveillance Videos
  We describe our implementation of a web-based video annotation tool built for the use of the law enforcement authorities for rapid analysis of surveillance videos. The tool makes use of crowdsourcing in a controlled manner to distribute annotation tasks to a set of trained "crowds" and aggregates the results for the law enforcement authorities.

7.2 Future Work

Our methods can be improved and extended in the following ways:

- A Four Description MDC Architecture using HEVC/VPx Bitstreams:
  We investigated a four description temporal-spatial four description MDC with its adaptive concealment methods for H.264 bitstreams. In future, a comparative study can be done by using these methods for HEVC/VPx bitstreams. Error resilience capabilities of these methods can then be compared at the system level.

- MDC using adaptive subsampling-based architectures:
  In this thesis, we proposed error concealment methods only for fixed MDC ar-

chitectures, where the number of descriptions and the partition methods were hardwired. However, an MDC architecture could be selected adaptively from a set of available architectures, based on current network conditions, desired processing complexity and target latency requirements This will make the coded bitstreams to contain a desired level of error resilience. Redundancy-rate distortion analysis [54] can be useful in terms of quantifying the amount of redundancy in an encoded bitstream.

- Error Resilience using Additional Keyframe Information in VPx

  We investigated VPx error resilience using duplicated prediction information. However, without error-free keyframes, the decoder can produce and propagate errors. To provide resilience to keyframes, we could encode a downsampled version of keyframes or original keyframes with a lower bitrate. This additional keyframe information can then be sent as side information to assist the decoder in error concealment. A part of this idea is currently under investigation that can be further enriched using adaptive keyframe downsampling ratio that can achieve a required amount of redundancy to provide error resilience.

- Improved Jelly Filling Segmentation using Differential Geometry

  Our proposed jelly filling segmentation can provide acceptable results for most of our image data. It can be improved by using the principles of differential geometry to discriminate objects based on mathematically modeled shape priors. A statistical distance measure such as Bhattacharyya distance could be used to provide likelihood of a particular object configuration. This likelihood can then be used as an influence factor of our jelly filling framework. This will allow segmenting a generic biological structure using the jelly filling concepts.

- 3D MPP for Nuclei Segmentation

  In our work, we proposed a midpoint analysis and 2D MPP based nuclei segmentation method that can be further improved using a 3D MPP simulation. This could help quantification of nuclei in 3D that can also help visualization.

- Segmentation using "Negative" Shape Priors

  A typical MPP for image segmentation uses shape priors to estimate the likelihood of a particular shape that needs to be segmented. Our general observation with microscopy images indicate that due to various reasons, bright regions that do not represent biological quantities appear in typical shapes. Detection and estimation of such regions can be done using geometric shape priors for the purpose of eliminating them from the segmentation outcome.

- A Generic Secure Crowdsourcing Platform with Hierarchical Crowd Model

  We could generalize and improve our crowdsourcing platform to suit a specific target application. An advanced tool offering more features for forensic analysis is developed in [261]. A hierarchical crowd model has been under development and has a potential to also investigate social behavior, mob mentality and other features of crowdsourcing. Another web-based tool that crowdsources classification and validation of food images with their labels, is developed using our platform. A version of our platform is also developed to detect pharmaceutical pills in images.

7.3 Publications Resulting From The Thesis

Book Chapters

- **N. Gadgil**, M. Yang, M. L. Comer, and E. J. Delp, "Multiple description coding," *Academic Press Library in Signal Processing: Image and Video Compression and Multimedia*, S. Theodoridis and R. Chellappa, Eds. Oxford, UK: Elsevier, 2014, vol. 5, no. 8, pp. 251-294.

Journal Articles

- **N. Gadgil**, P. Salama, K. Dunn, and E. J. Delp, "Jelly filling image segmentation of biological structures," *To be submitted to the IEEE Transactions on Medical Imaging.*

- **N. Gadgil** and P. Salama, K. Dunn, and E. J. Delp, Nuclei segmentation of microscopy images using marked point process, *To be submitted to the SPIE Journal of Medical Imaging.*

- M. Yang, **N. Gadgil**, M. L. Comer, and E. J. Delp, "Adaptive error concealment for multiple description video coding," *Signal Processing: Image Communication*, 2016.

- J. Duda, P. Korus, **N. Gadgil**, K. Tahboub, and E. J. Delp, "Image-like 2D barcodes using generalizations of the Kuznetsov-Tsybakov problem," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 4, pp. 691-703, April 2016.

Conference Papers

- C. Fu, **N. Gadgil**, K. Tahboub, P. Salama, K. Dunn and E. J. Delp, "Four dimensional image registration for intravital microscopy," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, July 2016, Las Vegas, NV.

- **N. Gadgil**, P. Salama, K. Dunn and E. J. Delp, "Jelly filling segmentation of fluorescence microscopy images containing incomplete labeling," *Proceedings of the IEEE International Symposium on Biomedical Imaging*, April 2016, Prague, Czech Republic.

- **N. Gadgil**, P. Salama, K. Dunn and E. J. Delp, "Nuclei segmentation of fluorescence microscopy images based on midpoint analysis and marked point process," *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 37-40, March 2016, Santa Fe, NM.

- **N. Gadgil** and E. J. Delp, "VPx error resilient video coding using duplicated prediction information", *Proceedings of the IS&T Electronic Imaging: Conference on Visual Information Processing and Communication VII*, February 2016, San Francisco, CA.

- K. Tahboub, **N. Gadgil**, and E. J. Delp, "Content-based video retrieval on mobile devices: How much content is enough?" *Proceedings of the IEEE International Conference on Image Processing*, pp. 1603-1607, September 2015, Quebec City, Canada.

- **N. Gadgil**, H. Li, and E. J. Delp, "Spatial subsampling-based multiple description video coding with adaptive temporal-spatial error concealment," *Proceedings of the Picture Coding Symposium*, pp. 90-94, May 2015, Cairns, Australia.

- J. Duda, K. Tahboub, **N. Gadgil**, and E. J. Delp, "The use of asymmetric numeral systems as an accurate replacement for Huffman coding," *Proceedings of the Picture Coding Symposium*, pp. 65-69, May 2015, Cairns, Australia.

- K. Tahboub, **N. Gadgil**, J. Ribera, B. Delgado, and E. J. Delp, "An intelligent crowdsourcing system for forensic analysis of surveillance video," *Proceedings of the IS&T/SPIE Electronic Imaging: Video Surveillance and Transportation Imaging Applications*, vol. 9407, pp. 94070I: 1-9, February 2015, San Francisco, CA.

- J. Duda, **N. Gadgil**, K. Tahboub, and E. J. Delp, "Generalizations of the Kuznetsov-Tsybakov problem for generating image-like 2D barcodes," *Proceedings of the IEEE International Conference on Image Processing*, pp. 4221-4225, October 2015, Paris, France.

- **N. Gadgil**, K. Tahboub, D. Kirsh, and E. J. Delp, "A web-based video annotation system for crowdsourcing surveillance videos," *Proceedings of the IS&T/SPIE Electronic Imaging: Imaging and Multimedia Analytics in a Web and Mobile World*, vol. 9027, pp. 90270A: 1-12, February 2014, San Francisco, CA.

- K. Tahboub, **N. Gadgil**, M. L. Comer, and E. J. Delp, "An HEVC compressed domain content-based video signature for copy detection and video retrieval," *Proceedings of the IS&T/SPIE Electronic Imaging: Imaging and Multimedia*

Analytics in a Web and Mobile World, vol. 9027, pp. 90270E: 1-13, February 2014, San Francisco, CA.

- **N. Gadgil**, M. L. Comer, and E. J. Delp, "Adaptive error concealment for multiple description video coding using error estimation," *Proceedings of the Picture Coding Symposium*, pp. 97-100, December 2013, San Jose, CA.

- **N. Gadgil**, M. Yang, M. L. Comer, and E. J. Delp, "Adaptive error concealment for multiple description video coding using motion vector analysis," *Proceedings of the IEEE International Conference on Image Processing*, pp. 1637-1640, October 2012, Orlando, FL.

LIST OF REFERENCES

LIST OF REFERENCES

[1] "Cisco visual networking index: Forecast and methodology, 2014-2019," *White Paper*, May 2015, Cisco Systems Inc., San Jose, CA.

[2] D. Tse and P. Viswanath, *Fundamentals of wireless communication.* Cambridge University Press, 2005.

[3] C. Shannon, "A mathematical theory of communications," *The Bell System Technical Journal*, vol. 27:379-423, pp. 623–656, October 1948.

[4] C. Shannon, "Coding theorems for a discrete source with a fidelity criterion," *Institute of Radio Engineers, International Convention Record*, vol. 7, pp. 325–350, 1959.

[5] T. Cover and J. Thomas, *Elements of Information Theory.* Wiley, 1991, New York, NY.

[6] A. Tekalp, *Digital video processing.* Prentice Hall, 1995, Westford, MA.

[7] P. Tudor, "MPEG-2 video compression," *Electronics Communication Engineering Journal*, vol. 7, no. 6, pp. 257–264, December 1995.

[8] T. Weigand, G. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, July 2003.

[9] G. Sullivan, J. Ohm, W. Han, and T. Weigand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, December 2012.

[10] J. Bankoski, J. Koleszar, L. Quillio, J. Salonen, P. Wilkins, and Y. Xu, "VP8 Data Format and Decoding Guide: RFC 6386," 2013, URL: http://datatracker.ietf.org/doc/rfc6386/, Last accessed: 01/14/2016.

[11] J. Bankoski, R. Bultje, A. Grange, Q. Gu, J. Han, J. Koleszar, D. Mukherjee, P. Wilkins, and Y. Xu, "Towards a next generation open-source video codec," *Proceedings of the SPIE/IS&T Electronic Imaging: Visual Information Processing and Communication IV*, vol. 8666, pp. 1–13, February 2013, Burlingame, CA.

[12] D. Mukherjee, H. Su, J. Bankoski, A. Converse, J. Han, Z. Liu, and Y. Xu, "An overview of new video coding tools under consideration for VP10: The successor to VP9," *Proceedings of SPIE Applications of Digital Image Processing XXXVIII*, vol. 9599, pp. 1E:1–12, August 2015, San Diego, CA.

[13] I. Richardson, *H.264 and MPEG-4 video compression: Video Coding for Next-generation Multimedia.* Wiley Online Library, 2004, Chichester, UK.

[14] Y. Wang, J. Ostermann, and Y. Zhang, *Video processing and communications.* Prentice Hall, 2002, vol. 5.

[15] Y. Wang and Q.-F. Zhu, "Error control and concealment for video communication: A review," *Proceedings of the IEEE*, vol. 86, no. 5, pp. 974–997, May 1998.

[16] "The WebM Project," URL: http://www.webmproject.org/, Last accessed: 01/14/2016.

[17] T. Wiegand, G. Sullivan, and A. Luthra, "Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-T Rec. H.264—ISO/IEC 14496-10 AVC)," *Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, JVTG050*, May 2003, Geneva, Switzerland.

[18] B. Bross, W.-J. Han, J.-R. Ohm, G. Sullivan, Y.-K. Wang, and T. Wiegand, "High efficiency video coding (HEVC) text specification draft 10 (for FDIS & last call)," *Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG, JCTVC-L1003*, January 2013, Geneva, Switzerland.

[19] K. Suhring, "H.264/AVC software coordination," http://iphome.hhi.de/suehring/tml/, Last accessed: 05/06/2016.

[20] K. Suehring and K. Sharman, "HM software repository," https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/, Last accessed: 05/06/2016.

[21] S. Wenger, M. Hannuksela, T. Stockhammer, M. Westerlund, and D. Singer, "RTP payload format for H.264 video," RFC 3984, Tech. Rep., 2005.

[22] "Narrow-band visual telephone systems and terminal equipment," ITU-T Recommendation H.320, 1999.

[23] K. Brandenburg and G. Stoll, "The ISO-MPEG-1 audio: A generic standard for coding of high-quality digital audio," *Journal of the Audio Engineering Society*, vol. 42, no. 10, pp. 780–792, 1994.

[24] T. Stockhammer, "Dynamic adaptive streaming over HTTP: Standards and design principles," *Proceedings of the second annual ACM conference on Multimedia systems*, pp. 133–144, February 2011, San Jose, CA.

[25] W. Lam, A. Reibman, and B. Liu, "Recovery of lost or erroneously received motion vectors," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, no. 12, pp. 417–420, April 1993, Minneapolis, MN.

[26] S. Shirani, F. Kossentini, and R. Ward, "A concealment method for video communications in an error-prone environment," *IEEE Journal on Selected Areas in Communication*, vol. 18, no. 6, pp. 1822–1833, June 2000.

[27] J. Seiler and A. Kaup, "Adaptive joint spatio-temporal error concealment for video communication," *Proceedings of the IEEE Workshop on Multimedia Signal Processing*, pp. 229–234, October 2008, Cairns, Australia.

[28] W-Y.Kung, C.-S. Kim, and C.-C. Kuo, "Spatial and temporal error concealment techniques for video transmission over noisy channels," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 7, pp. 789–802, July 2006.

[29] M. Yang, "Multiple description video coding with adaptive error concealment," Ph.D. dissertation, Purdue University, West Lafayette, May 2012.

[30] T. Stockhammer, M. Hannuksela, and T. Wiegand, "H.264/AVC in wireless environments," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 657–673, July 2003.

[31] M. Karczewicz and R. Kurceren, "The SP-and SI-frames design for H.264/AVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 637–644, July 2003.

[32] G. Sullivan, "Seven steps toward a more robust codec design," JVT-C117, May 2002.

[33] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1103–1120, September 2007.

[34] V. Goyal, "Multiple description coding: Compression meets the network," *IEEE Signal Processing Magazine*, vol. 18, no. 5, pp. 74–93, September 2001.

[35] N. Gadgil, M. Yang, M. Comer, and E. Delp, "Multiple description coding," S. Theodoridis and R. Chellappa, Eds. Elsevier, 2014, vol. 5, no. 8, pp. 251–294, Oxford, UK.

[36] J. Wolf, A. Wyner, and J. Ziv, "Source coding for multiple descriptions," *The Bell System Technical Journal*, vol. 59, no. 8, pp. 1417–1426, October 1980.

[37] Z. Zhang and T. Berger, "New results in binary multiple descriptions," *IEEE Transactions on Information Theory*, vol. 33, no. 4, pp. 502 – 521, July 1987.

[38] H. Witsenhausen, "On source networks with minimal breakdown degradation," *The Bell System Technical Journal*, vol. 59, no. 6, pp. 1083–1087, July-August 1980.

[39] H. Witsenhausen and A. Wyner, "Source coding for multiple descriptions II: A binary source," *The Bell System Technical Journal*, vol. 60, no. 10, pp. 2281–2292, December 1981.

[40] L. Ozarow, "On a source-coding problem with two channels and three receivers," *The Bell System Technical Journal*, vol. 59, no. 10, pp. 1909–1921, December 1980.

[41] N. Jayant and S. Christensen, "Effects of packet losses in waveform coded speech and improvements due to an odd-even sample-interpolation procedure," *IEEE Transactions on Communications*, vol. 29, no. 2, pp. 101 – 109, February 1981.

[42] N. Jayant, "Subsampling of a DPCM speech channel to provide two "Self-Contained" half-rate channels," *The Bell System Technical Journal*, vol. 60, no. 4, pp. 501–509, April 1981.

[43] A. Gamal and T. Cover, "Achievable rates for multiple descriptions," *IEEE Transactions on Information Theory*, vol. 28, no. 6, pp. 851 – 857, November 1982.

[44] A. Gamal and T. Cover, "Information theory of multiple descriptions," Department of Statistics, Stanford University, Stanford, CA, Tech. Rep. 43, December 1980.

[45] T. Berger and Z. Zhang, "Minimum source degradation in binary source encoding," *IEEE Transactions on Information Theory*, vol. 29, no. 6, pp. 807 – 814, November 1983.

[46] R. Ahlswede, "The rate-distortion region for multiple descriptions without excess rate," *IEEE Transactions on Information Theory*, vol. 31, no. 6, pp. 721–726, November 1985.

[47] R. Ahlswede, "On multiple descriptions and team guessing," *IEEE Transactions on Information Theory*, vol. 32, no. 4, pp. 543–549, July 1986.

[48] V. Vaishampayan, "Design of multiple description scalar quantizers," *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 821 –834, May 1993.

[49] V. Vaishampayan and J. Domaszewicz, "Design of entropy-constrained multiple-description scalar quantizers," *IEEE Transactions on Information Theory*, vol. 40, no. 1, pp. 245–250, January 1994.

[50] V. Vaishampayan and J. Batllo, "Multiple description transform codes with an application to packetized speech," *Proceedings of the International Symposium on Information Theory*, p. 458, July 1994, Trondheim, Norway.

[51] V. Vaishampayan, "Application of multiple description codes to image and video transmission over lossy networks," *Proceedings of the 7th International Packet Video Workshop*, pp. 55–60, 1996, Brisbane, Australia.

[52] J. Batllo and V. Vaishampayan, "Asymptotic performance of multiple description transform codes," *IEEE Transactions on Information Theory*, vol. 43, no. 2, pp. 703–707, March 1997.

[53] Y. Wang, M. Orchard, and A. Reibman, "Multiple description image coding for noisy channels by pairing transform coefficients," *Proceedings of the IEEE First Workshop on Multimedia Signal Processing*, pp. 419–424, June 1997, Princeton, NJ.

[54] M. Orchard, Y. Wang, V. Vaishampayan, and A. Reibman, "Redundancy rate-distortion analysis of multiple description coding using pairwise correlating transforms," *Proceedings of the IEEE International Conference on Image Processing*, vol. 1, pp. 608–611, October 1997, Santa Barbara, CA.

[55] Y. Wang, M. Orchard, V. Vaishampayan, and A. Reibman, "Multiple description coding using pairwise correlating transforms," *IEEE Transactions on Image Processing*, vol. 10, no. 3, pp. 351 –366, March 2001.

[56] S. Hemami, "Reconstruction-optimized lapped orthogonal transforms for robust image transmission," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, no. 2, pp. 168–181, April 1996.

[57] H. Malvar and D. Staelin, "The LOT: Transform coding without blocking effects," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, no. 4, pp. 553–559, April 1989.

[58] A. Reibman, H. Jafarkhani, Y. Wang, M. Orchard, and R. Puri, "Multiple description coding for video using motion compensated prediction," *Proceedings of the IEEE International Conference on Image Processing*, vol. 3, pp. 837–841, October 1999, Kobe, Japan.

[59] A. Reibman and H. Jafarkhani and Y. Wang and M. Orchard and R. Puri, "Multiple description video coding using motion compensated temporal prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 3, pp. 193–204, March 2002.

[60] Y. Wang, A. Reibman, and S. Lin, "Multiple description coding for video delivery," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 57 –70, January 2005.

[61] J. Apostolopoulos, "Error-resilient video compression through the use of multiple states," *Proceedings of the IEEE International Conference on Image Processing*, vol. 3, pp. 352 –355, September 2000, Vancouver, Canada.

[62] N. Franchi, M. Fumagalli, R. Lancini, and S. Tubaro, "Multiple description video coding for scalable and robust transmission over IP," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 3, pp. 321 – 334, March 2005.

[63] N. Boulgouris, K. Zachariadis, A. Leontaris, and M. Strintzis, "Drift-free multiple description coding of video," *Proceedings of the IEEE Fourth Workshop on Multimedia Signal Processing*, pp. 105–110, October 2001, Cannes, France.

[64] M. Yang, M. Comer, and E. Delp, "A four-description MDC for high loss-rate channels," *Proceedings of the Picture Coding Symposium*, pp. 418–421, December 2010, Nagoya, Japan.

[65] A. Reibman, H. Jafarkhani, Y. Wang, and M. Orchard, "Multiple description video using rate-distortion splitting," *Proceedings of the IEEE International Conference on Image Processing*, vol. 1, pp. 978–981, October 2001, Thessaloniki , Greece.

[66] C. Kim and S. Lee, "Multiple description coding of motion fields for robust video transmission," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 9, pp. 999 –1010, September 2001.

[67] Y. Wang and S. Lin, "Error-resilient video coding using multiple description motion compensation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 6, pp. 438–452, June 2002.

[68] C. Zhu and M. Liu, "Multiple description video coding based on hierarchical B pictures," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 4, pp. 511–521, April 2009.

[69] S. Regunathan and K. Rose, "Efficient prediction in multiple description video coding," *Proceedings of the IEEE International Conference on Image Processing*, vol. 1, pp. 1020–1023, September 2000, Vancouver, Canada.

[70] X. Tang and A. Zakhor, "Matching pursuits multiple description coding for wireless video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 6, pp. 566–575, June 2002.

[71] K. Matty and L. Kondi, "Balanced multiple description video coding using optimal partitioning of the DCT coefficients," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 7, pp. 928–934, July 2005.

[72] R. Puri and K. Ramchandran, "Multiple description source coding using forward error correction codes," *Conference Record of the Thirty-Third Asilomar Conference on Signals, Systems, and Computers*, vol. 1, pp. 342–346, October 1999, Pacific Grove, CA.

[73] R. Bernardini, M. Durigon, R. Rinaldo, L. Celetto, and A. Vitali, "Polyphase spatial subsampling multiple description coding of video streams with H264," *Proceedings of the IEEE International Conference on Image Processing*, vol. 5, pp. 3213–3216, October 2004, Singapore.

[74] M. Gallant, S. Shirani, and F. Kossentini, "Standard-compliant multiple description video coding," *Proceedings of the IEEE International Conference on Image Processing*, vol. 1, pp. 946–949, October 2001, Thessaloniki, Greece.

[75] I. Bajic and J. Woods, "Domain-based multiple description coding of images and video," *IEEE Transactions on Image Processing*, vol. 12, no. 10, pp. 1211–1225, September 2003.

[76] J. R. Ohm, "Three-dimensional subband coding with motion compensation," *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 559–571, September 1994.

[77] M. V. der Schaar and D. Turaga, "Multiple description scalable coding using wavelet-based motion compensated temporal filtering," *Proceedings of the IEEE International Conference on Image Processing*, vol. 3, pp. 489–492, September 2003, Barcelona, Spain.

[78] E. Akyol, A. Tekalp, and M. Civanlar, "Scalable multiple description video coding with flexible number of descriptions," *Proceedings of the IEEE International Conference on Image Processing*, vol. 3, pp. 712–715, September 2005, Genoa, Italy.

[79] N. Kamnoonwatana, D. Agrafiotis, and C. Canagarajah, "Flexible adaptive multiple description coding for video transmission," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 1, pp. 1–11, January 2012.

[80] D. Wang, N. Canagarajah, and D. Bull, "Slice group based multiple description video coding with three motion compensation loops," *Proceedings of the IEEE International Symposium on Circuits and Systems*, vol. 2, pp. 960–963, May 2005, Kobe, Japan.

[81] Z. Wei, K. Ma, and C. Cai, "Prediction-compensated polyphase multiple description image coding with adaptive redundancy control," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 3, pp. 465–478, March 2012.

[82] C. Lin, T. Tillo, Y. Zhao, and B. Jeon, "Multiple description coding for H.264/AVC with redundancy allocation at macro block level," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 5, pp. 589 –600, May 2011.

[83] T. Tillo, M. Grangetto, and G. Olmo, "Redundant slice optimal allocation for H.264 multiple description coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 1, pp. 59–70, January 2008.

[84] T. Tillo and G. Olmo, "Data-dependent pre-and postprocessing multiple description coding of images," *IEEE Transactions on Image Processing*, vol. 16, no. 5, pp. 1269–1280, May 2007.

[85] S. Shirani, M. Gallant, and F. Kossentini, "Multiple description image coding using pre-and post-processing," *Proceedings of the International Conference on Information Technology: Coding and Computing*, pp. 35–39, April 2001, Las Vegas, NV.

[86] N. Franchi, M. Fumagalli, and R. Lancini, "Flexible redundancy insertion in a polyphase down sampling multiple description image coding," *Proceedings of the IEEE International Conference on Multimedia and Expo*, vol. 2, pp. 605 – 608, August 2002, Lausanne, Switzerland.

[87] Z. Wei, C. Cai, and K.-K. Ma, "A novel H.264-based multiple description video coding via polyphase transform and partial prediction," *Proceedings of the International Symposium on Intelligent Signal Processing and Communications*, vol. 1, pp. 151 –154, December 2006, Yonago, Japan.

[88] D. Wang, N. Canagarajah, D. Redmill, and D. Bull, "Multiple description video coding based on zero padding," *Proceedings of the 2004 International Symposium on Circuits and Systems*, vol. 2, pp. 205–208, May 2004, Vancouver, Canada.

[89] G. Zhang and R. Stevenson, "Efficient error recovery for multiple description video coding," *Proceedings of the IEEE International Conference on Image Processing*, vol. 2, pp. 829–832, October 2004, Singapore.

[90] E. Akyol and A. Tekalp and M. Civanlar, "A flexible multiple description coding framework for adaptive peer-to-peer video streaming," *IEEE Journal of Selected Topics in Signal Processing,*, vol. 1, no. 2, pp. 231 –245, August 2007.

[91] M. Yang, M. Comer, and E. Delp, "An adaptable spatial-temporal error concealment method for multiple description coding based on error tracking," *Proceedings of the IEEE International Conference on Image Processing*, September 2011, Brussels, Belgium.

[92] M. Yang, M. Comer, and E. Delp, "Macroblock level adaptive error concealment methods for MDC," *Proceedings of the Picture Coding Symposium*, pp. 485–488, May 2012, Krakow, Poland.

[93] W.-J. Tsai and J.-Y. Chen, "Joint temporal and spatial error concealment for multiple description video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 12, pp. 1822–1833, December 2010.

[94] N. Kamnoonwatana, D. Agrafiotis, and C. Canagarajah, "Flexible adaptive multiple description coding for video transmission," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 1, pp. 1–11, January 2012.

[95] C. Hsiao and W. Tsai, "Hybrid multiple description coding based on H.264," *IEEE Transactions on Circuits Systems Video Technology*, vol. 20, no. 1, pp. 76–87, January 2010.

[96] N. Gadgil, M. Yang, M. Comer, and E. Delp, "Adaptive error concealment for multiple description video coding using motion vector analysis," *Proceedings of the IEEE International Conference on Image Processing*, pp. 1637–1640, October 2012, Orlando, FL.

[97] N. Gadgil, M. Comer, and E. Delp, "Adaptive error concealment for multiple description video coding using error estimation," *Proceedings of the Picture Coding Symposium*, pp. 97–100, December 2013, San Jose, CA.

[98] N. Gadgil, H. Li, and E. Delp, "Spatial subsampling-based multiple description video coding with adaptive temporal-spatial error concealment," *Proceedings of the Picture Coding Symposium*, pp. 90–94, May 2015, Cairns, Australia.

[99] R. Aeron, V. Goyal, and J. Kovacevic, "Multiple description transform coding of audio using optimal transforms of arbitrary dimension," Patent US 6,253,185 B1, June 26, 2001, Murray Hill, NJ.

[100] V. Goyal, J. Kovacevic, and M. Vetterli, "Multiple description transform coding of images using optimal transforms of arbitrary dimension," Patent US 6,330,370 B2, December 11, 2001, Murray Hill, NJ.

[101] M. Orchard, H. Jafarkhani, A. Reibman, and Y. Wang, "Method and apparatus for accomplishing multiple description coding for video," Patent US 6,556,624 B1, April 29, 2003, New York, NY.

[102] H. Jafarkhani, M. Orchard, A. Reibman, and Y. Wang, "Multiple description coding communication system," Patent US 6,823,018 B1, November 23, 2004, New York, NY.

[103] V. Goyal, J. Kovacevic, and F. Masson, "Method and apparatus for wireless transmission using multiple description coding," Patent US 6,983,243 B1, January 3, 2006, Murray Hill, NJ.

[104] J. Apostolopoulos, S. Basu, G. Cheung, R. Kumar, S. Roy, W. tan Tan, S. Wee, T. Wong, and B. Shen, "Method for handling off multiple description streaming media sessions between servers in fixed and mobile streaming media systems," Patent US 6,996,618 B2, February 7, 2006, Houston, TX.

[105] S. Lin, A. Vetro, and Y. Wang, "Adaptive error-resilient video encoding using multiple description motion compensation," Patent US 7,106,907 B2, September 12, 2006, Cambridge, MA.

[106] P. Chou, V. Padmanabhan, and H. Wang, "Layered multiple description coding," Patent US 7,426,677 B2, September 16, 2008, Redmond, WA.

[107] C. Cheng and C. Bauer, "Correlating and decorrelating transforms for multiple description coding system," Patent US 7,536,299 B2, May 19, 2009, San Francisco, CA.

[108] A. Irvine and V. Raveendran, "Apparatus and method for multiple description encoding," Patent US 7,561,073 B2, July 14, 2009, San Diego, CA.

[109] S. Panwar, K. Ross, and Y. Wang, "On demand peer-to-peer video streaming with multiple description coding," Patent US 7,633,887 B2, December 15, 2009.

[110] M. Paniconi, J. Carrig, and Z. Miao, "Variable support robust transform for multiple description coding," Patent US 7,817,869 B2, October 19, 2010.

[111] M. Cancemi and A. Vitali, "Method and system for multiple description coding and computer program product therefor," Patent US 7,991,055 B2, August 2, 2011.

[112] W. Zhan, "Method, apparatus and system for multiple-description coding and decoding," Patent US 8,279,947 B2, October 2, 2012.

[113] Y.-K. Wang, M. Hannuksela, and I. Bouazizi, "Apparatus and method for indicating track relationships in media files," Patent US 8,365,060 B2, January 29, 2013.

[114] M. Kyan, L. Guan, M. Arnison, and C. Cogswell, "Feature extraction of chromosomes from 3-D confocal microscope images," *IEEE Transactions on Biomedical Engineering*, vol. 48, no. 11, pp. 1306–1318, November 2001.

[115] K. Dunn, R. Sandoval, K. Kelly, P. Dagher, G. Tanner, S. Atkinson, R. Bacallao, and B. Molitoris, "Functional studies of the kidney of living animals using multicolor two-photon microscopy," *American Journal of Physiology-Cell Physiology*, vol. 283, no. 3, pp. C905–C916, September 2002.

[116] J. Lichtman and J.-A. Conchello, "Fluorescence microscopy," *Nature methods*, vol. 2, no. 12, pp. 910–919, November 2005.

[117] K. Dunn and T. Sutton, "Functional studies in living animals using multiphoton microscopy," *ILAR journal*, vol. 49, no. 1, pp. 66–77, 2008.

[118] P. So, C. Dong, B. Masters, and K. Berland, "Two-photon excitation fluorescence microscopy," *Annual review of biomedical engineering*, vol. 2, no. 1, pp. 399–429, 2000.

[119] J. Irwin and J. Macdonald, "Microscopic observations of the intrahepatic circulation of living guinea pigs," *The Anatomical Record*, vol. 117, no. 1, pp. 1–15, September 1953.

[120] M. Clemens, P. McDonagh, I. Chaudry, and A. Baue, "Hepatic microcirculatory failure after ischemia and reperfusion: Improvement with ATP-MgCl2 treatment," *American Journal of Physiology-Heart and Circulatory Physiology*, vol. 248, no. 6, pp. H804–H811, June 1985.

[121] M. Ghiron, "Über eine neue methode mikroskopischer untersuchung em lebenden organismus," *Zbl Physiol*, vol. 26, pp. 613–617, 1912.

[122] A. Maunsbach, "Observations on the segmentation of the proximal tubule in the rat kidney: Comparison of results from phase contrast, fluorescence and electron microscopy," *Journal of ultrastructure research*, vol. 16, no. 3, pp. 239–258, 1966.

[123] M. Steinhausen and G. A. Tanner, *Microcirculation and tubular urine flow in the mammalian kidney cortex (in vivo microscopy).* Springer-Verlag, Berlin-Heidelberg, Germany, 1976.

[124] K. Svoboda and R. Yasuda, "Principles of two-photon excitation microscopy and its applications to neuroscience," *Neuron*, vol. 50, no. 6, pp. 823–839, June 2006.

[125] J. Skoch, G. A. Hickey, S. T. Kajdasz, B. T. Hyman, and B. J. Bacskai, "In vivo imaging of amyloid-beta deposits in mouse brain with multiphoton microscopy," *Amyloid Proteins*, ser. Methods in Molecular Biology, E. M. Sigurdsson and J. M. Walker, Eds. Humana Press, 2005, vol. 299, pp. 349–363.

[126] A. Zarbock and K. Ley, "New insights into leukocyte recruitment by intravital microscopy," *Visualizing Immunity*, ser. Current Topics in Microbiology and Immunology, M. Dustin and D. McGavern, Eds. Springer Berlin Heidelberg, 2009, vol. 334, pp. 129–152.

[127] C. Sumen, T. Mempel, I. Mazo, and U. von Andrian, "Intravital microscopy: Visualizing immunity in context," *Immunity*, vol. 21, no. 3, pp. 315–329, September 2004.

[128] M. J. Hickey and P. Kubes, "Intravascular immunity: The host-pathogen encounter in blood vessels," *Nature Reviews - Immunology*, vol. 9, no. 5, pp. 364–375, May 2009.

[129] S. Lunt, C. Gray, C. Reyes-Aldasoro, S. Matcher, and G. Tozer, "Application of intravital microscopy in studies of tumor microcirculation," *Journal of Biomedical Optics*, vol. 15, no. 1, pp. 1–14, February 2010.

[130] J. Condeelis and J. E. Segall, "Intravital imaging of cell movement in tumours," *Nature Reviews - Cancer*, vol. 3, pp. 921–930, December 2003.

[131] D. Fukumura, D. G. Duda, L. L. Munn, and R. K. Jain, "Tumor microvasculature and microenvironment: Novel insights through intravital imaging in pre-clinical models," *Microcirculation*, vol. 17, no. 3, pp. 206–225, April 2003.

[132] Nobel Media AB 2014, "The Nobel prize in Chemistry 2008," Last accessed: 11 May 2016. [Online]. Available: http://www.nobelprize.org/nobel_prizes/chemistry/laureates/2008/

[133] Nobel Media AB 2014, "The Nobel prize in Chemistry 2014 - Advanced information," Last accessed: 11 May 2016. [Online]. Available: http://www.nobelprize.org/nobel_prizes/chemistry/laureates/2014/advanced.html

[134] D. Murphy, *Fundamentals of light microscopy and electronic imaging.* John Wiley & Sons, 2002, New York, NY.

[135] U. Kubitscheck, *Fluorescence microscopy: From principles to biological applications.* John Wiley & Sons, 2013, Weinheim, Germany.

[136] J. Lakowicz, *Principles of fluorescence spectroscopy.* Springer Science & Business Media, 2013, Singapore.

[137] S. Gage, "Modern dark-field microscopy and the history of its development," *Transactions of the American Microscopical Society*, vol. 39, no. 2, pp. 95–141, April 1920.

[138] F. Zernike, "How I discovered phase contrast," *Science*, vol. 121, no. 3141, pp. 345–349, May 1955.

[139] M. Pluta, "Nomarski's DIC microscopy: A review," *Proceedings of the Phase Contrast and Differential Interference Contrast Imaging Techniques and Applications, International Society for Optics and Photonics*, vol. 1846, pp. 10–25, October 1992, Warsaw, Poland.

[140] H. Verschueren, "Interference reflection microscopy in cell biology: Methodology and applications," *Journal of Cell Science*, vol. 75, no. 1, pp. 279–301, April 1985.

[141] R. Hoffman, "The modulation contrast microscope: Principles and performance," *Journal of Microscopy*, vol. 110, no. 3, pp. 205–222, July 1977.

[142] S. Hell and J. Wichmann, "Breaking the diffraction resolution limit by stimulated emission: Stimulated-emission-depletion fluorescence microscopy," *Optics letters*, vol. 19, no. 11, pp. 780–782, June 1994.

[143] S. Hess, T. Girirajan, and M. Mason, "Ultra-high resolution imaging by fluorescence photoactivation localization microscopy," *Biophysical journal*, vol. 91, no. 11, pp. 4258–4272, December 2006.

[144] M. Rust, M. Bates, and X. Zhuang, "Sub-diffraction-limit imaging by stochastic optical reconstruction microscopy (STORM)," *Nature methods*, vol. 3, no. 10, pp. 793–796, August 2006.

[145] A. Jablonski, "Efficiency of anti-Stokes fluorescence in dyes," *Nature*, vol. 131, no. 839-840, p. 21, April 1933.

[146] D. Piston, "Imaging living cells and tissues by two-photon excitation microscopy," *Trends in cell biology*, vol. 9, no. 2, pp. 66–69, February 1999.

[147] P. So, *Two-photon fluorescence light microscopy*. Macmillan Publishers Ltd., Nature Publishing Group, 2002.

[148] E. Hoover and J. Squier, "Advances in multiphoton microscopy technology," *Nature Photonics*, vol. 7, no. 2, pp. 93–101, February 2013.

[149] R. Sabnis, *Handbook of fluorescent dyes and probes*. John Wiley & Sons, Inc, 2015, Hoboken, NJ.

[150] H. Ishikawa-Ankerhold, R. Ankerhold, and G. Drummen, "Advanced fluorescence microscopy techniques-FRAP, FLIP, FLAP, FRET and FLIM," *Molecules*, vol. 17, no. 4, p. 4047, April 2012.

[151] M. Minsky, "Memoir on inventing the confocal scanning microscope," *Scanning*, vol. 10, no. 4, pp. 128–138, August 1988.

[152] W. Denk, J. Strickler, and W. Webb, "Two-photon laser scanning fluorescence microscopy," *Science*, vol. 248, no. 4951, pp. 73–76, April 1990.

[153] R. Benninger, M. Hao, and D. Piston, "Multi-photon excitation imaging of dynamic processes in living cells and tissues," *Reviews of Physiology Biochemistry and Pharmacology*. Springer-Verlag, 2008, vol. 160, pp. 71–92, Heidelberg, Germany.

[154] K. Lorenz, "Registration and segmentation based analysis of microscopy images," Ph.D. dissertation, Purdue University, West Lafayette, August 2012.

[155] B. Luck, K. Carlson, A. Bovik, and R. Richards-Kortum, "An image model and segmentation algorithm for reflectance confocal images of in vivo cervical tissue," *IEEE Transactions on Image Processing*, vol. 14, no. 9, pp. 1265–1276, September 2005.

[156] L. Shapiro and G. Stockman, "Computer vision," *Prentice Hall*, 2001, New Jersey, NJ.

[157] A. Dufour, V. Shinin, S. Tajbakhsh, N. Guillén-Aghion, J.-C. Olivo-Marin, and C. Zimmer, "Segmenting and tracking fluorescent cells in dynamic 3-D microscopy with coupled active surfaces," *IEEE Transactions on Image Processing*, vol. 14, no. 9, pp. 1396–1410, September 2005.

[158] P. Sarder and A. Nehorai, "Deconvolution methods for 3-D fluorescence microscopy images," *IEEE Signal Processing Magazine*, vol. 23, no. 3, pp. 32–45, May 2006.

[159] E. Meijering, I. Smal, O. Dzyubachyk, and J.-C. Olivo-Marin, "Time-lapse imaging," Q. Wu, F. Merchant, and K. Castleman, Eds., 2008, no. 15, pp. 401–440, Elsevier Academic Press, Burlington, MA.

[160] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 679–698, November 1986.

[161] C. Harris and M. Stephens, "A combined corner and edge detector," *Proceedings of the Alvey vision conference*, vol. 15, p. 50, August 1988, Manchester, UK.

[162] M. Sezgin and B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 146–168, January 2004.

[163] R. Haralick, S. Sternberg, and X. Zhuang, "Image analysis using mathematical morphology," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-9, no. 4, pp. 532–550, July 1987.

[164] J. Serra, "Morphological filtering: An overview," *Signal processing*, vol. 38, no. 1, pp. 3–11, July 1994.

[165] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, January 1988.

[166] R. Delgado-Gonzalo, V. Uhlmann, D. Schmitter, and M. Unser, "Snakes on a plane: A perfect snap for bioimage analysis," *IEEE Signal Processing Magazine*, vol. 32, no. 1, pp. 41–48, January 2015.

[167] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," *International Journal of Computer Vision*, vol. 22, no. 1, pp. 61–79, February 1997.

[168] T. Chan and L. Vese, "Active contours without edges," *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 266–277, February 2001.

[169] S. Lankton and A. Tannenbaum, "Localizing region-based active contours," *IEEE Transactions on Image Processing*, vol. 17, no. 11, pp. 2029–2039, November 2008.

[170] C. Pluempitiwiriyawej, J. Moura, Y. Wu, and C. Ho, "STACS: New active contour scheme for cardiac MR image segmentation," *IEEE Transactions on Medical Imaging*, vol. 24, no. 5, pp. 593–603, May 2005.

[171] L. Coulot, H. Kirschner, A. Chebira, J. Moura, J. Kovacevic, E. Osuna, and R. Murphy, "Topology preserving STACS segmentation of protein subcellular location images," *Proceedings of the IEEE International Symposium on Biomedical Imaging*, pp. 566–569, April 2006, Arlington, VA.

[172] M. Velliste and R. Murphy, "Automated determination of protein subcellular locations from 3D fluorescence microscope images," *Proceedings of the IEEE International Symposium on Biomedical Imaging*, pp. 867–870, June 2002, Washington, DC.

[173] B. Li and S. Acton, "Active contour external force using vector field convolution for image segmentation," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2096–2106, August 2007, URL: http://viva-lab.ece.virginia.edu/downloads.html.

[174] H. Li, T. Shen, M. Smith, I. Fujiwara, D. Vavylonis, and X. Huang, "Automated actin filament segmentation, tracking and tip elongation measurements based on open active contour models," *Proceedings of the IEEE International Symposium on Biomedical Imaging*, pp. 1302–1305, June 2009, Boston, MA., URL: http://athena.physics.lehigh.edu/jfilament/download.chy.

[175] K. Lorenz, P. Salama, K. Dunn, and E. Delp, "Three dimensional segmentation of fluorescence microscopy images using active surfaces," *Proceedings of the IEEE International Conference on Image Processing*, pp. 1153–1157, September 2013, Melbourne, Australia.

[176] B. Li and S. Acton, "Automatic active model initialization via poisson inverse gradient," *IEEE Transactions on Image Processing*, vol. 17, no. 8, pp. 1406–1420, August 2008.

[177] L. Vincent and P. Soille, "Watersheds in digital spaces: An efficient algorithm based on immersion simulations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 6, pp. 583–598, June 1991.

[178] V. Grau, A. Mewes, M. Alcaniz, R. Kikinis, and S. Warfield, "Improved watershed transform for medical image segmentation using prior information," *IEEE Transactions on Medical Imaging*, vol. 23, no. 4, pp. 447–458, April 2004.

[179] K. Keraudren, M. Spitaler, V. Braga, D. Rueckert, and L. Pizarro, "Two-step watershed segmentation of epithelial cells," *Proceedings of the 6th International Workshop on Microscopic Image Analysis with Applications in Biology*, September 2011, Heidelberg, Germany.

[180] L. Vincent, "Morphological grayscale reconstruction in image analysis: Applications and efficient algorithms," *IEEE Transactions on Image Processing*, vol. 2, no. 2, pp. 176–201, April 1993.

[181] J. Sijbers, P. Scheunders, M. Verhoye, A. Van der Linden, D. Van Dyck, and E. Raman, "Watershed-based segmentation of 3D MR data for volume quantization," *Magnetic Resonance Imaging*, vol. 15, no. 6, pp. 679–688, 1997.

[182] G. Srinivasa, M. Fickus, Y. Guo, A. Linstedt, and J. Kovacevic, "Active mask segmentation of fluorescence microscope images," *IEEE Transactions on Image Processing*, vol. 18, no. 8, pp. 1817–1829, August 2009.

[183] J. Cardinale, G. Paul, and I. Sbalzarini, "Discrete region competition for unknown numbers of connected regions," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3531–3545, August 2012.

[184] P. Quelhas, M. Marcuzzo, A. Mendonça, and A. Campilho, "Cell nuclei and cytoplasm joint segmentation using the sliding band filter," *IEEE Transactions on Medical Imaging*, vol. 29, no. 8, pp. 1463–1473, August 2010.

[185] A. Narayanaswamy, S. Dwarakapuram, C. Bjornsson, B. Cutler, W. Shain, and B. Roysam, "Robust adaptive 3-D segmentation of vessel laminae from fluorescence confocal microscope images and parallel GPU implementation," *IEEE Transactions on Medical Imaging*, vol. 29, no. 3, pp. 583–597, March 2010.

[186] G. Paul, J. Cardinale, and I. Sbalzarini, "Coupling image restoration and segmentation: A generalized linear model/Bregman perspective," *International Journal of Computer Vision*, vol. 104, no. 1, pp. 69–93, March 2013, URL: http://mosaic.mpi-cbg.de/?q=downloads/imageJ.

[187] M. Jacob and M. Unser, "Design of steerable filters for feature detection using Canny-like criteria," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 8, pp. 1007–1019, August 2004, URL: http://bigwww.epfl.ch/demo/steerable/.

[188] D. Ciresan, A. Giusti, L. Gambardella, and J. Schmidhuber, "Deep neural networks segment neuronal membranes in electron microscopy images," *Proceedings of the Advances in neural information processing systems*, pp. 2843–2851, December 2012, Lake Tahoe, NV.

[189] M. Seyedhosseini, M. Sajjadi, and T. Tasdizen, "Image segmentation with cascaded hierarchical models and logistic disjunctive normal networks," *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2168–2175, December 2013, Sydney, Australia.

[190] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *CoRR*, vol. abs/1505.04597, 2015, http://arxiv.org/abs/1505.04597, Last accessed: 05/15/2016.

[191] J. Kawahara, A. BenTaieb, and G. Hamarneh, "Deep features to classify skin lesions," *Proceedings of the IEEE International Symposium on Biomedical Imaging*, April 2016, Prague, Czech Republic.

[192] M. Abràmoff, P. Magalhães, and S. Ram, "Image processing with ImageJ," *Biophotonics International*, vol. 11, no. 7, pp. 36–42, July 2004.

[193] J. Schindelin, I. Arganda-Carreras, E. Frise, V. Kaynig, M. Longair, T. Pietzsch, S. Preibisch, C. Rueden, S. Saalfeld, B. Schmid, J.-Y. Tinevez, D. White, V. Hartenstein, K. Eliceiri, P. Tomancak, and A. Cardona, "Fiji: an open-source platform for biological-image analysis," *Nature methods*, vol. 9, no. 7, pp. 676–682, June 2012.

[194] F. De Chaumont, S. Dallongeville, N. Chenouard, N. Hervé, S. Pop, T. Provoost, V. Meas-Yedid, P. Pankajakshan, T. Lecomte, Y. Le Montagner, T. Lagache, A. Dufour, and J.-C. Olivo-Marin, "Icy: An open bioimage informatics platform for extended reproducible research," *Nature methods*, vol. 9, no. 7, pp. 690–696, June 2012.

[195] J. Clendenon, C. Phillips, R. Sandoval, S. Fang, and K. Dunn, "Voxx: a PC-based, near real-time volume rendering system for biological microscopy," *American Journal of Physiology-Cell Physiology*, vol. 282, no. 1, pp. C213–C218, January 2002.

[196] K. Dunn, T. Sutton, and R. Sandoval, "Live-animal imaging of renal function by multiphoton microscopy," *Current Protocols in Cytometry*, no. 12.9, pp. 1–18, July 2007.

[197] J. Ryan, K. Dunn, and B. Decker, "Effects of chronic kidney disease on liver transport: quantitative intravital microscopy of fluorescein transport in the rat liver," *American Journal of Physiology-Regulatory, Integrative and Comparative Physiology*, vol. 307, no. 12, pp. R1488–R1492, October 2014.

[198] M. Valera and S. Velastin, "Intelligent distributed surveillance systems: a review," *IEE Proceedings of Vision, Image and Signal Processing*, vol. 152, no. 2, pp. 192–204, April 2005.

[199] N. Haering, P. Venetianer, and A. Lipton, "The evolution of video surveillance: an overview," *Machine Vision and Applications*, vol. 19, no. 5-6, pp. 279–290, September 2008.

[200] M. Shah, O. Javed, and K. Shafique, "Automated visual surveillance in realistic scenarios," *IEEE Transactions on Multimedia*, vol. 14, no. 1, pp. 30–39, January 2007.

[201] H. Dee and S. Velastin, "How close are we to solving the problem of automated visual surveillance?" *Machine Vision and Applications*, vol. 19, no. 5-6, pp. 329–343, September 2008.

[202] B. Zhani, D. Monekosso, P. Remagnino, S. Velastin, and L. Xu, "Crowd analysis: A survey," *Machine Vision and Applications*, vol. 19, no. 5-6, pp. 345–357, October 2008.

[203] J. Aggarwal and Q. Cai, "Human motion analysis: A review," *Proceedings of the IEEE Nonrigid and Articulated Motion Workshop*, pp. 90–102, June 1997, San Juan, Puerto Rico.

[204] J. Aggarwal and M. Ryoo, "Human activity analysis: A review," *ACM Journal of Computing Surveys*, vol. 43, no. 3, pp. 1–43, April 2011.

[205] P. Turaga, R. Chellappa, V. Subrahmanian, and O. Udrea, "Machine recognition of human activities: A survey," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1473–1488, September 2008.

[206] S. Srivastava and E. Delp, "Video-based real-time surveillance of vehicles," *Journal of Electronic Imaging*, vol. 22, no. 4, pp. 041 103:1–16, October 2013.

[207] S. Srivastava, K. Ng, and E. Delp, "Crowd flow estimation using multiple visual features for scenes with changing crowd densities," *Proceedings of the 8th International Conference on Advanced Video and Signal-Based Surveillance*, pp. 60–65, August 2011, Klagenfurt, Austria.

[208] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Computing Surveys*, vol. 38, no. 4, Article No. 13, December 2006.

[209] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 34, no. 3, pp. 334–352, August 2004.

[210] I. Saleemi, K. Shafique, and M. Shah, "Probabilistic modeling of scene dynamics for applications in visual surveillance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 8, pp. 1472–1485, July 2009.

[211] S. Khan and M. Shah, "Tracking multiple occluding people by localizing on multiple scene planes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 3, pp. 505–519, April 2009.

[212] B. Benfold and I. Reid, "Stable multi-target tracking in real-time surveillance video," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3457–3464, June 2011, Colorado Springs, CO.

[213] J. Howe, "The rise of crowdsourcing," *Wired magazine*, vol. 14, no. 6, pp. 1–4, 2006, Dorsey Press.

[214] J. Surowiecki, *The wisdom of crowds.* Random House Digital, Inc., 2005, New York, NY.

[215] A. Doan, R. Ramakrishnan, and A. Halevy, "Crowdsourcing systems on the world-wide web," *Communications of the ACM*, vol. 54, no. 4, pp. 86–96, April 2011.

[216] K. Chen, C. Chang, C. Wu, Y. Chang, and L. Lei, "Quadrant of euphoria: a crowdsourcing platform for QoE assessment," *IEEE Network*, vol. 24, no. 2, pp. 28–35, March 2010.

[217] Ó. Salas, V. Adzic, A. Shah, and H. Kalva, "Assessing Internet video quality using crowdsourcing," *Proceedings of the 2nd ACM international workshop on Crowdsourcing for Multimedia*, pp. 23–28, October 2013, Barcelona, Spain.

[218] T. Hoßfeld, M. Seufert, M. Hirth, T. Zinner, P. Tran-Gia, and R. Schatz, "Quantification of YouTube QoE via crowdsourcing," *Proceedings of the IEEE International Symposium on Multimedia*, pp. 494–499, December 2011, Dana Point, CA.

[219] M. Davis, "Media streams: an iconic visual language for video annotation," *Proceedings of the IEEE Symposium on Visual Languages*, pp. 196–202, August 1993, Bergen, Norway.

[220] C. Vondrick, D. Ramanan, and D. Patterson, "Efficiently scaling up video annotation with crowdsourced marketplaces," *Computer Vision-ECCV 2010, Lecture Notes in Computer Science*, vol. 6314, pp. 610–623, September 2010, Springer-Verlag, Germany.

[221] A. Francois, R. Nevatia, J. Hobbs, R. Bolles, and J. Smith, "VERL: an ontology framework for representing and annotating video events," *IEEE Transactions on Multimedia*, vol. 12, no. 4, pp. 76–86, October 2005.

[222] J. Assfalg, M. Bertini, C. Colombo, and A. Bimbo, "Semantic annotation of sports videos," *IEEE Transactions on Multimedia*, vol. 9, no. 2, pp. 52–60, August 2002.

[223] S. Vijayanarasimhan and K. Grauman, "Large-scale live active learning: Training object detectors with crawled data and crowds," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1449–1456, June 2011, Providence, RI.

[224] A. Sorokin, D. Berenson, S. Srinivasa, and M. Hebert, "People helping robots helping people: Crowdsourcing for grasping novel objects," *Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2117–2122, October 2010, Taipei, Taiwan.

[225] W. Willett, J. Heer, and M. Agrawala, "Strategies for crowdsourcing social data analysis," *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pp. 227–236, May 2012, Austin, TX.

[226] A. Quinn, B. Bederson, T. Yeh, and J. Lin, "Crowdflow: Integrating machine learning with mechanical turk for speed-cost-quality flexibility," *Human Computer Interaction Lab, Technical Report*, May 2010, University of Maryland, College Park, MD.

[227] R. Brantingham and A. Hossain, "Crowded: a crowd-sourced perspective of events as they happen," *Proceedings of the IS&T/SPIE Conference on Defense, Security, and Sensing*, pp. 87 580D–1–8, February 2013, Burlingame, CA.

[228] C. Vondrick, D. Patterson, and D. Ramanan, "Efficiently scaling up crowdsourced video annotation," *International Journal of Computer Vision*, vol. 101, no. 1, pp. 184–204, January 2013.

[229] D. Brabham, *Crowdsourcing.* The MIT Press, 2013, Cambridge, MA.

[230] M. Yang, N. Gadgil, M. Comer, and E. Delp, "Adaptive error concealment for temporal-spatial multiple description video coding," *Signal Processing: Image Communication*, 2016.

[231] R. Gandhi, M. Yang, D. Koutsonikolas, Y. Hu, M. Comer, A. Mohamed, and C. Wang, "The impact of inter-layer network coding on the relative performance of MRC/MDC WiFi media delivery," *Proceedings of the 21st international workshop on Network and operating systems support for digital audio and video*, pp. 27–32, June 2011, Vancouver, Canada.

[232] T.-L. Lin, S. Kanumuri, Y. Zhi, D. Poole, P. Cosman, and A. Reibman, "A versatile model for packet loss visibility and its application to packet prioritization," *IEEE Transactions on Image Processing*, vol. 19, no. 3, pp. 722–734, March 2010.

[233] S. Kanumuri, P. Cosman, A. Reibman, and V. Vaishampayan, "Modeling packet-loss visibility in MPEG-2 video," *IEEE Transactions on Multimedia*, vol. 8, no. 2, pp. 341–355, April 2006.

[234] R. Zhang, "Efficient inter-layer motion compensation and error resilience for spatially scalable video coding," Ph.D. dissertation, Purdue University, West Lafayette, 2009.

[235] U. Horn, K. Stuhlmller, M. Link, and B. Girod, "Robust internet video transmission based on scalable coding and unequal error protection," *Signal Processing: Image Communication*, vol. 15, pp. 77–94, September 1999.

[236] N. Gadgil and E. Delp, "Vpx error resilient video coding using duplicated prediction information," *Proceedings of the IS&T Electronic Imaging: Conference on Visual Information Processing and Communication VII*, February 2016, San Francisco, CA.

[237] "VP8 Encode Parameter Guide," URL: http://www.webmproject.org/docs/encoder-parameters/, Last accessed: 01/14/2016.

[238] N. Gadgil, P. Salama, K. Dunn, and E. Delp, "Jelly filling segmentation of fluorescence microscopy images containing incomplete labeling," *Proceedings of the IEEE International Symposium on Biomedical Imaging*, April 2016, Prague, Czech Republic.

[239] Y. Le, R. Kroeker, H. Kipfer, and C. Lin, "Development and evaluation of TWIST Dixon for dynamic contrast-enhanced (DCE) MRI with improved acquisition efficiency and fat suppression," *Journal of Magnetic Resonance Imaging*, vol. 36, no. 2, pp. 483–491, August 2012.

[240] C. Giardina and E. Dougherty, *Morphological methods in image and signal processing*, 1988, Prentice Hall Inc., Engelwood Cliffs, NJ.

[241] M. Roberts, J. Packer, M. Sousa, and J. Mitchell, "A work-efficient GPU algorithm for level set segmentation," *Proceedings of the Conference on High Performance Graphics*, pp. 123–132, June 2010, Saarbrucken, Germany.

[242] N. Gadgil, P. Salama, K. Dunn, and E. Delp, "Nuclei segmentation of fluorescence microscopy images based on midpoint analysis and marked point process," *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 37–40, March 2016, Santa Fe, NM.

[243] M. Van Lieshout, *Markov point processes and their applications.* Imperial College Press, 2000, London, United Kingdom.

[244] A. Papoulis and S. Pillai, *Probability, random variables and stochastic processes.* Tata McGraw-Hill Education, 2002, Noida, India.

[245] X. Descombes and J. Zerubia, "Marked point process in image analysis," *IEEE Signal Processing Magazine*, vol. 19, no. 5, pp. 77–84, September 2002.

[246] D. Snyder, *Random point processes.* John Wiley & Sons Inc., 1975, New York, NY.

[247] W. Hastings, "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, vol. 57, no. 1, pp. 97–109, April 1970.

[248] P. Green, "Reversible jump Markov chain Monte Carlo computation and Bayesian model determination," *Biometrika*, vol. 82, no. 4, pp. 711–732, December 1995.

[249] X. Descombes, R. Minlos, and E. Zhizhina, "Object extraction using a stochastic birth-and-death dynamics in continuum," *Journal of Mathematical Imaging and Vision*, vol. 33, no. 3, pp. 347–359, March 2009.

[250] C. Benedek and X. Descombes and J. Zerubia, "Building development monitoring in multitemporal remotely sensed image pairs with stochastic birth-death dynamics," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 1, pp. 33–50, January 2012.

[251] M. Ortner, X. Descombes, and J. Zerubia, "A marked point process of rectangles and segments for automatic analysis of digital elevation models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 1, pp. 105–119, January 2008.

[252] C. Benedek, X. Descombes, and J. Zerubia, "Building extraction and change detection in multitemporal remotely sensed images with multiple birth and death dynamics," *Proceedings of the IEEE Workshop on Applications of Computer Vision*, pp. 1–6, December 2009, Snowbird, UT.

[253] H. Zhao, M. Comer, and M. De Graef, "A unified Markov random field/marked point process image model and its application to computational materials," *Proceedings of the IEEE International Conference on Image Processing*, pp. 6101–6105, October 2014, Paris, France.

[254] X. Descombes, F. Kruggel, G. Wollny, and H. Gertz, "An object-based approach for detecting small brain lesions: Application to Virchow-Robin spaces," *IEEE Transactions on Medical Imaging*, vol. 23, no. 2, pp. 246–255, February 2004.

[255] Q. Liao and Y. Deng, "An accurate segmentation method for white blood cell images," *Proceedings of the IEEE International Symposium on Biomedical Imaging*, pp. 245–248, June 2002, Washington D.C.

[256] N. Gadgil, K. Tahboub, D. Kirsh, and E. Delp, "A web-based video annotation system for crowdsourcing surveillance videos," *Proceedings of the IS&T/SPIE Electronic Imaging: Imaging and Multimedia Analytics in a Web and Mobile World*, no. 90270A, pp. 90 270A:1–12, February 2014, San Francisco, CA.

[257] J. Le, A. Edmonds, V. Hester, and L. Biewald, "Ensuring quality in crowdsourced search relevance evaluation: The effects of training question distribution," *ACM Special Interest Group on Information Retrieval 2010 Workshop on Crowdsourcing for Search Evaluation*, pp. 21–26, July 2010, Geneva, Switzerland.

[258] D. Oleson, A. Sorokin, G. Laughlin, V. Hester, J. Le, and L. Biewald, "Programmatic gold: Targeted and scalable quality assurance in crowdsourcing," *Proceedings of the Association for the Advancement of Artificial Intelligence Human Computation Workshop*, vol. 11, pp. 43–48, August 2011, San Francisco, CA.

[259] University of Edinburgh, The BEHAVE Dataset, http://groups.inf.ed.ac.uk/vision/.

[260] E. Keogh, Q. Zhu, B. Hu, Y. Hao, X. Xi, L. Wei, and C. Ratanamahatana, (2011) The UCR Time Series Classification/Clustering Homepage: www.cs.ucr.edu/.

[261] K. Tahboub, N. Gadgil, J. Ribera, B. Delgado, and E. Delp, "An intelligent crowdsourcing system for forensic analysis of surveillance video," *Proceedings of the IS&T/SPIE Electronic Imaging: Video Surveillance and Transportation Imaging Applications*, no. 94070I, pp. 94 070I:1–9, February 2015, San Francisco, CA.

VITA

VITA

Neeraj J. Gadgil was born in Mumbai, India. He received the B.E. (Hons.) in Electrical and Electronics Engineering (EEE) from Birla Institute of Technology and Science (BITS) Pilani, India.

Mr. Gadgil joined the Ph.D. program at the School of Electrical and Computer Engineering, Purdue University, West Lafayette, Indiana in August 2010. He worked at the Video and Image Processing Laboratory (VIPER) under the supervision of Professor Edward J. Delp. He was an intern at the video compression group of Qualcomm Inc., San Diego, CA in the summers of 2012 and 2015. Prior to the doctoral studies, he worked as a software engineer with the optical transport group at Cisco Systems Inc., Bangalore, India. He did a semester internship at the digital signal processing group of Tensilica Inc., Pune, India.

Mr. Gadgil is a recipient of the BITS Pilani Merit Scholarship, the M. K. Tata Trust Grant and the National Talent Search Scholarship awarded by the Govt. of India. His research interests are video compression, video transmission, multimedia systems, statistical image processing and computer vision. He is a student member of the IEEE, the IEEE Signal Processing Society, SPIE, IS&T and Eta Kappa Nu.