Towards Geographically-Distributed Immersive Collaborations with Delay Guarantee

Modeling, Simulation, Synthesis, and Compression

MAURITZ HAMONANGAN PANGGABEAN

DOCTORAL THESIS

Submitted in Partial Fulfillment of the

Requirements for the Degree of

PHILOSOPHIAE DOCTOR

Norwegian University of Science and Technology

Faculty of Information Technology, Mathematics and Electrical Engineering

Department of Telematics

NTNU
Norges Teknisk Naturvitenskapelige Universitet
Norwegian University of Science and Technology

Thesis for the degree of Philosophiae Doctor

Faculty of Information Technology, Mathematics, and Electrical Engineering
Department of Telematics

ISBN 978-82-326-0170-7 (printed version)
ISBN 978-82-326-0171-4 (electronic version)
ISSN 1503-8181

Doctoral thesis at NTNU, 2014:123

Printed in Norway by NTNU-trykk, Trondheim

To every thing there is a season,
and a time to every purpose under the heaven.

King Solomon, *Eccleciastes 3:3 (KJV)*

Contents

Abstract

This PhD thesis addresses the vision of a geographically distributed immersive collaboration system that supports real-time delay-sensitive collaborations based on visual cues between performers for synchronization. Examples include collaborative dancing and remote conducting of choirs. The collaborators from different remote places perform in their own collaboration space (CS), but achieve the quality of experience (QoE) as if they perform in the same place and scene. To arrive at that very high level of QoE, all physical surfaces of a CS are constructed from arrays of multiview autostereoscopic displays and high-resolution micro-cameras with microphones and speakers. The CSs are interconnected by a high-speed network over which the audiovisual data are transported. The capacity of the links in the network varies as they may be shared by other users outside the collaboration system.

The information era with rapid developments in many fields is the right time to address the complex collaboration system. It is, however, still non-existent due to at least four technical challenges. First, the synchronization is shown to be harmonious if the maximum end-to-end delay (EED) in processing and transporting video data between the connected CSs can be guaranteed at 11.5ms. As the Internet is not designed to deliver it, the Distributed Multimedia Plays (DMP) system architecture is proposed to address it by means of Quality Shaping. Second, the very low latency constraint becomes more challenging because the video quality rendered in the CSs must also be gracefully degraded regardless of changing network condition. Third, the immense traffic of audiovisual data generated from a CS requires creative data reduction and fast processing to minimize processing delay. The last challenge comes from the transient periods that are expected to occur frequently in such traffic because a CS transmits and receives visual signals only from segmented bodies of the performers. The segmentation is key in the adopted object-based video processing and compression to discard irrelevant data based on the eye gazes of the performers that are detected and tracked in real-time.

This thesis presents research work on four of many aspects of the collaboration system: modeling, simulation, synthesis, and compression. Since human body is the smallest building block for simulating the collaboration system, its modeling as a discrete-event system lies at heart of the modeling and simulation of the collaboration system. By modeling a human body as a system of sixteen interconnected limbs,

an event is defined as the spatial displacement of the two end points of a limb that represents its motion.

The motion of a human body is generated by simulating forward kinematics of its limbs using discrete-event simulation (DES) that includes both stochastic motion and gait cycles for walking and running as deterministic motion. DES guarantees that virtually unlimited unique sets of motions can be exactly reproduced. How any collaboration scenario with arbitrary number of CSs and collaborators can be simulated is illustrated by a detailed example. Based on the silhouette of visualized moving human bodies and the technical specification of the CSs, traces of uniquely reproducible transient traffic are synthesized as input traffic to DES of DMP architecture. Moreover, traffic from motions due to camera zoom and panning are also studied by real measurement and mathematical modeling.

DMP guarantees maximum EED because every DMP node can drop video packets deliberately according to instantaneous network condition to guarantee their local delays. Thus, intelligent packet dropping is the main source of information loss in DMP. Two schemes for such compression of image sequences are studied in pixel- and transform domains. The first employs windowed kriging (WK) for optimal image interpolation in the Near-natural Object Coding proposed in DMP, and the latter is based on discrete cosine transform (DCT). The application of WK to luminance and chrominance is studied in terms of visual quality and computational time. Furthermore, an ultrafast, embedded, quality-scalable, DCT-based image coding scheme for DMP is proposed and shown to be technically feasible for hardware implementation. The application of resampling to regions in an image indicated by the tracked eye gazes is also studied, together with the effects to visual quality.

Addressing the compression aspect is important as the basis for future study of estimating video quality that results from packet dropping. Since this is not possible with the above methods of traffic synthesis, the study on compression complements the aspects of modeling, simulation, and synthesis, showing the coherence of the work.

Acknowledgments

This thesis is submitted to the Norwegian University of Science and Technology (NTNU) in partial fulfillment of the requirements for the degree of *philosophiae doctor* (PhD). This doctoral work was conducted at the Department of Telematics (ITEM) at NTNU in Gløshaugen, Trondheim under full PhD scholarship from NTNU in the period of January 2009 until December 2012. The duration includes one-year duty work at ITEM and completing four courses of total 30 credits with minimum grade B, according to NTNU regulation. The main supervisor and co-supervisor are Professor Leif Arne Rønningen and Associate Professor Harald Øverby, respectively.

I am deeply grateful to Leif Arne and Harald for giving me a healthy balance of freedom and guidance with wisdom and patience that makes it a real pleasure to work with them. Their trust and confidence in me have been an instrumental source of the much needed persistence to deliver this thesis in the right time. I have learned many valuable lessons from them over the transient and stationary periods along my journey of research to be an independent skilled researcher. The casual interactions and friendly conversations with them have brightened many of my days.

It is also a great learning experience to collaborate in research team with Maciej Wielgosz at AGH University of Science and Technology in Krakow, Poland, Özgür Tamer at Dokuz Eylül University in Izmir, Turkey, Ameen Chilwan, and Jiang Wang. Thank you very much to all of you!

I also appreciate the work and constructive comments from the evaluation committee and all the anonymous reviewers that are important for improving the quality of the thesis and the papers resulting from this PhD work. The lessons from them have helped me honing my writing skills.

My sincere appreciation goes to all my colleagues at ITEM who have provided an inspiring and friendly environment for research. In particular, Poul Heegaard (head of department), Randi Fløtnes and Mona Nordaune (executive officers), and Pål Sæther (chief engineer) deserve special thanks for their excellent cooperation and help. It is also a pleasure to work with Norvald Stol and Stig Frode Mjølsnes. Pleasant memories also come from fellow PhD students and researchers during my time at ITEM, especially from my (former) mates at room B-222 in chronological order: Muhammad Qasim Khan, Sergey Gladysh, Elissar Khloussy, Razib Khan, Jonas Wäfler, and Maria Line.

Outside ITEM and NTNU, I will always remember with smile the camaraderie, hospitality, and support from fellow Indonesians in *Keluarga Trondheim*. They have painted auroral Indonesian colors in the moving pictures of my life. I am also indebted to my brothers and sisters in the Indonesian Bible Study group, the Anglican church, and the *Frikirke* in Trondheim for their kind encouragement and prayers.

I cannot overstate my gratitude to my father and mother in Indonesia for their never-ceasing love, prayer, and trust in me. May this long-awaited PhD bring more joy in their inspiring lives! I also thank my family for their loving care and encouragement during my being away 7,000 miles from home since 2006 when I started my master's study in Eindhoven, the Netherlands. This is also a timely opportunity to remember and appreciate all my teachers who have kindled in me the spirit of a lifelong learner and showed me the beauty of science and technology.

Finally I warmly thank my wife Dessy for her love, patience, and understanding for faithfully accompanying me in the journey through the ups and downs in the PhD tunnel, especially in the last miles during the preparation of this thesis.

Ad majorem Dei gloriam.

Technical abbreviations

A	AN	Access node
	AppTraNetLFC	Application Transport Network Link Flow Control
	AV	Audiovisual
	AVC	Advanced Video Coding
	AXI	Advanced eXtensible Interface
B	bpp	Bit per pixel
	BRAM	Block random access memory
	BWT	Burrows-Wheeler transform
C	CAVE	Cave automatic virtual environment
	CbN	Compression by network
	CPU	Central processing unit
	CR	Compression ratio
	CS	Collaboration space
	CVE	Collaborative virtual environment
D	DCT	Discrete cosine transform
	DEMOS	Discrete Event Modeling on Simula
	DES	Discrete event simulation
	DF	Downsampling factor
	DMA	Direct memory access
	DMP	Distributed Multimedia Plays
	DPCM	Differential pulse-code modulation
E	EED	End-to-end delay
F	FFS	Finite Fourier series
	FPGA	Field-programmable gate array
	fps	Frame per second
H	HD	High-definition
	HW	Hardware

I	IDCT	Inverse discrete cosine transform
	IP(v6)	Internet Protocol (version 6)
	IPSec	Internet Protocol Security
J	JPEG	Joint Photographic Experts Group
L	LCD	Liquid crystal display
	LI	Linear interpolation
	LUT	Look-up table
N	NN	Network node
	NOC	Near-natural Object Coding
O	OK	Ordinary kriging
P	PC	Personal computer
	PCIe	Peripheral component interconnect express
	PNG	Portable Network Graphics
	PSNR	Peak signal-to-noise ratio
Q	QoE	Quality of experience
	QoS	Quality of service
	QC	Quality Control
	QS	Quality Shaping
	QSP	Quality Shaping Profile
R	RD	Rate-distortion
	RGB(A)	Red green blue alpha
	RMSE	Root mean square error
	ROM	Range of motion
	RQ	Research question
	RTP	Real-time Transport Protocol
S	SAGE	Scalable adaptive graphics environment
	SP	Scene Profile
	SSIM	Structure similarity index
	SW	Software
T	TCP	Transmission Control Protocol
U	UDP	User Datagram Protocol
V	VQ	Video quality
	VR	Virtual reality
W	WHT	Walsh-Hadamard transform
	WK	Windowed kriging

List of Figures

List of Tables

PART I

Summary

CHAPTER 1

Introduction

This chapter explains what this PhD thesis is about, what it aims to accomplish, and why it matters. It starts by stating the vision and importance of a geographically distributed, immersive, real-time collaboration system as the motivation of the PhD work [Rønningen (2011b)]. As the concept of collaboration can mean differently to different people, various concepts and terminologies related to collaboration are explained, and the focus of the work is stated to set the boundaries of the problem scope clearly. It is important to note that the envisioned collaboration system does not exist yet. Thus, it is exemplified with a simple collaboration for better understanding and imagination by the reader. The exemplified system is referred to as the 'reference system' and revisited later in one of the included papers. The major technical challenges in realizing the vision are derived from the reference system, and some of them are given special attention and formulated as the three research questions (RQs) in the PhD research. A diagram based on the reference system describes how the RQs relate and support each other, showing the coherence and flow of the thesis. The research methodology adopted to address the RQs is discussed and then followed by the thesis structure and a list of all the papers published during the PhD program. Six of the papers are included in this thesis.

1.1 The motivating vision

With rapid developments and innovations, the world has been witnessing how quickly technologies grow from concepts to products that are smaller, faster, smarter, and environmentally more friendly. The tremendous growth opens the way towards the possibility of realizing a collaboration system that is more complex and advanced than the existing videoconferencing [Altunbasak et al. (2011)]. The advanced system for the future will connect users located in remote places, i.e. geographically distributed, to communicate, interact and collaborate with near-natural quality of experience (QoE). The collaboration should include any type of activities that is more sensitive to latency than face-to-face conversation in a meeting. The working definition of QoE is "the degree of delight or annoyance of the user of an application or service. It results from the fulfillment of his or her expectations with respect to the utility and or enjoyment of the application or service in the light of the user's personality and current state" [Callet

et al. (2013)]. The near-natural QoE means that, having the feeling of being at the same place, the collaborators do not perceive any difference between real and virtual collaborations by means of a network of collaboration environments that facilitate audio, visual and haptic senses [Nechvatal (2009)].

At least three major driving forces are behind this vision [Altunbasak et al. (2011)]. The first is the desired ability to communicate and collaborate with people in a most natural manner with uncompromised quality. Second, the information complexity with the intelligent support involved in such a collaboration gradually becomes achievable. The third is the globally increasing sensitivity for environmental and energy issues in the 'green economy'. Information and communication technology is expected to contribute 15% global reduction of CO_2 emission by 2020 and energy efficiency savings of £500 billion [The Climate Group (2008); European Commission 2020 (2010)].

1.2 The focus

Since understanding and imagining the vision might be difficult due to its non-existence, this section and the next are devoted to making it easier. In this section, various important concepts related to collaboration are discussed, and then how the vision differs from them become clearer. Wolff et al. (2007) reviewed collaboration technologies with respect to closely coupled collaboration, which refers to the situation of close collaboration around shared objects between team members at remote locations. They proposed the following requirements for closely coupled collaboration.

1. **Communication of references** Verbal and non-verbal communication including facial expressions, gaze, pointing, posture, gestures, physical distance to others, and the use of shared objects and the environment around the participants.

2. **Shared object manipulation** The simultaneous action of modifying an object through its attributes, such as position or color. Real-time response and consistency are key here.

3. **Shared context** This is a logical consequence of shared object manipulation as it requires a level of proximity between collaborators and objects within a shared workspace. Thus shared context consists of three key aspects. The first is a level of *mobility within the workspace* as it is necessary to enable shared object manipulation. Furthermore, the awareness of the action of other participants is known as a fundamental feature in supporting cooperative work. In closely coupled collaboration, awareness is supported by sharing both *social context* between collaborators, and *spatial context* between collaborators and the shared objects and environment. Co-presence, or a feeling of "being there with them together" is seen as the perception of spatial and social togetherness between remote people when collaborating around shared objects.

They also classified geographically-distributed collaboration technologies into seven general categories below, and the last three are grouped as collaborative mixed reality.

1. **Audioconferencing** This audio-only technology spans both fixed and mobile telephony services as well as Internet-based audio tools.

2. **Groupware** It refers to window-based collaborative applications used on desktop computers and commonly provides a form of shared 2D desktop accessible for a group of people over a network. Conversational interaction is supported via text messages and, sometimes, live audio channels.

3. **Videoconferencing** This technology allows multiple remote people to participate in a tele-conference by exchanging live audiovisual (AV) data between remote sites. The video-signal transmission enables face-to-face conversations between the participants and may include non-verbal cues, such as gesturing, as long they are in the viewing field of the camera.

4. **Telepresence** Developed from videoconferencing, it comes with the aim to 'teleport' a person to a remote place, rather than providing a fixed 'window' as in conventional videoconferencing. It may be coupled with tele-robotics that might weaken the shared object manipulation.

5. **Distributed collaborative augmented reality** The goal of augmented reality (AR) is to enhance the real world with virtual objects. The AR users usually use see-through head-mounted displays (HMDs) to perceive synthetic 3D objects overlaid on the surrounding real environment. Besides interfacing motion tracking, HMD allows natural interactions with synthetic objects. A group of co-located people may share and manipulate a set of projected virtual objects in a common place.

6. **Collaborative desktop-based virtual reality** Virtual reality (VR) can be defined as a set of interfaces that provide the sensory experience which immerses the user in a completely synthetic environment. The 'virtual' environment is usually composed of geometric objects that are computer generated or other media, such as documents or video, which inhabit a 3D space and may provide spatial sound or haptic feedback. The endpoints of distributed VR system may be interconnected with audio-conferencing tools and a collaborative virtual environment (CVE) software (SW) system that enables the users to share the context of the virtual environment and to interact with each other and the inhabiting objects.

7. **Collaborative immersive virtual reality** A class of immersive displays in VR is spatially immersive display which provides a surrounding imagery of a virtual space. Thus, users are inside, rather than in front of, the 3D environment, unlike desktop display-systems and large flat or curved screens which display 3D graphics based on a user's tracked viewpoint environment.

Figure 1.1 illustrates the concept of sharing space and data for each category. From the extent of closely coupled collaboration summarized in Table 1.1 under the seven categories above, the collaborative immersive virtual reality obviously excels in all the requirements. Therefore, the focus of this work is on the collaboration via networked immersive CVEs [Alregib (2009)].

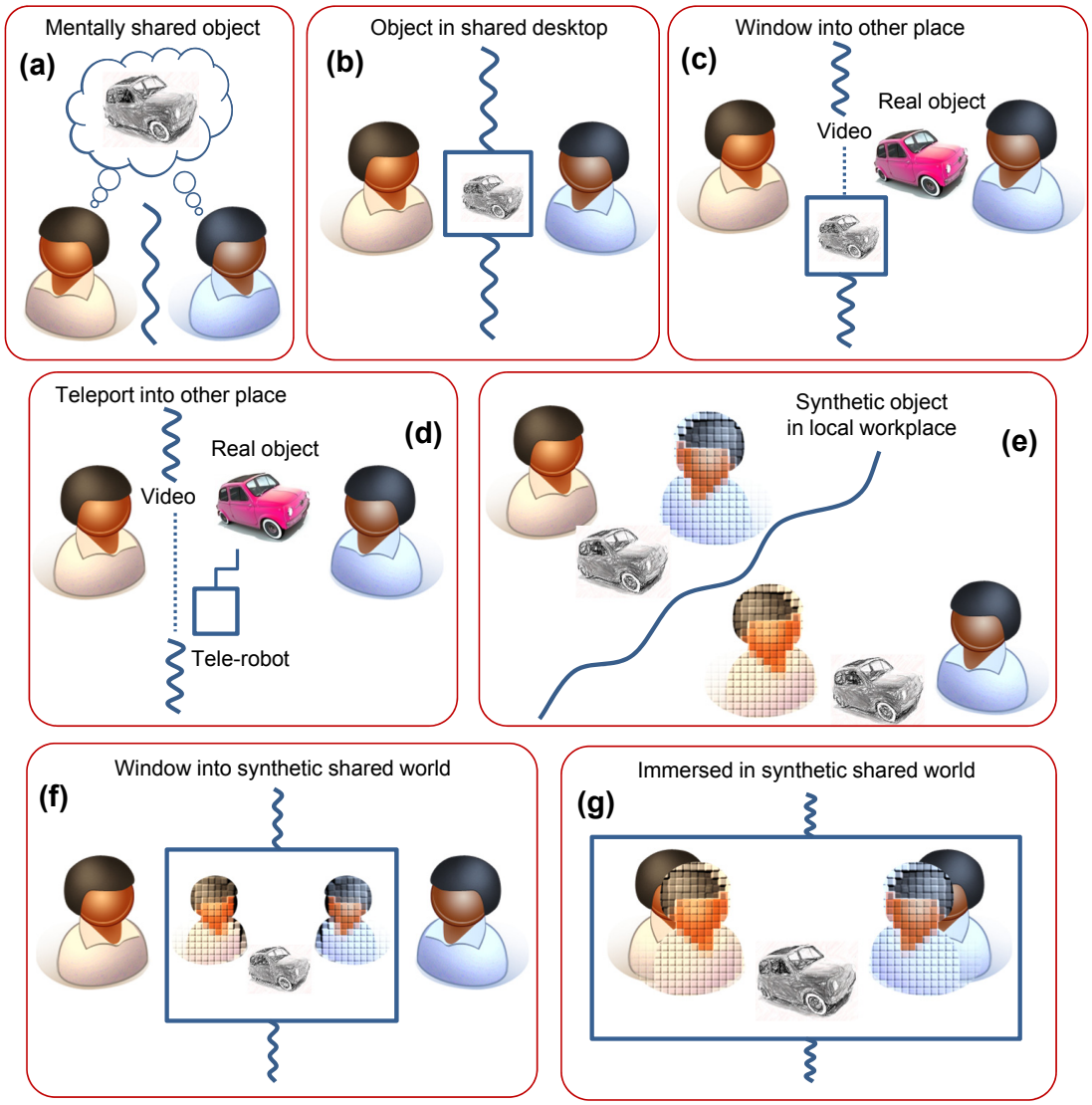


Figure 1.1: The concept of sharing context and data in audioconferencing (a), groupware (b), videoconferencing (c), telepresence (d), distributed collaborative AR (e), collaborative desktop-based VR (f), and collaborative immersive VR (g) [Wolff et al. (2007)].

1.3 The reference collaboration

This section clarifies the focus of the vision further with a simple example of the envisioned collaboration system referred to as the 'reference collaboration' in the thesis. It is shown in Figure 1.2 with a simple collaboration scenario that involves three groups of users in Trondheim, Tromsø and Oslo, three major cities in Norway.

In this setup, separated by the distance of around 771 km, a pair of dancer and singer in Tromsø (S1 and D1) collaborate with another pair in Trondheim (S2 and D2) via interconnected collaboration environments called collaboration spaces (CSs). Without time consuming and costly travel, they use the system to present a live art performance together in front of an audience in Oslo, which is approximately 390 and 1190 km away

Table 1.1: The extent of closely coupled collaboration under the seven categories of tele-collaboration technologies [Wolff et al. (2007)].

Technology	Shared object manipulation	Communication of references	Shared spatial context	Shared social context	Mobility within the shared space.
Audio-conferencing	NS	NS	S	S	NS
Groupware	UN	UN	S	S	NS
Video-conferencing	UN	N	S	PS	L
Tele-presence	UN	UN	FS	PS	UL
Augmented Reality	N	N	PS	PS	UL
Desktop-based CVE	UN	UN	FS	PS	UL
Immersive CVE	N	N	FS	PS	UL

NS: not supported, S: separated, UN: unnatural, N: natural, FS: fully shared, PS: partially shared, UL: unlimited, L: limited.

from Trondheim and Tromsø, respectively. The audience enjoy the show by looking at an aggregated view of the pairs on a large display. Thanks to the achieved near-natural QoE, they should think that the artists are performing live on the very stage in front of them. As shown by the aggregator in Figure 1.2, the two singers sing and interact with each other as a duo at the center of the stage, while the dancers perform a choreographed modern dance as a team besides the singers. This setup affects how each pair is positioned to each other in their collaboration environment.

The two CSs exchange AV data with each other, as indicated by the blue and red arrows. They start by negotiating their scene profiles (SPs) which define the technical specifications such as the spatial resolution and frame rate of the video data. Furthermore, the two CSs transmit data to the system in Oslo to be displayed in front of the audience. It is assumed that the audience does transmit data to the artists; hence, one-way communication to Oslo (dashed lines). End-to-end delay (EED) is a critical factor in a two-way communication, particularly when the collaboration is very sensitive to EED and requires a bounded value. Therefore, unlike the the video transmission to Oslo, the collaboration between Trondheim and Tromsø should operates on a network that guarantees network latency.

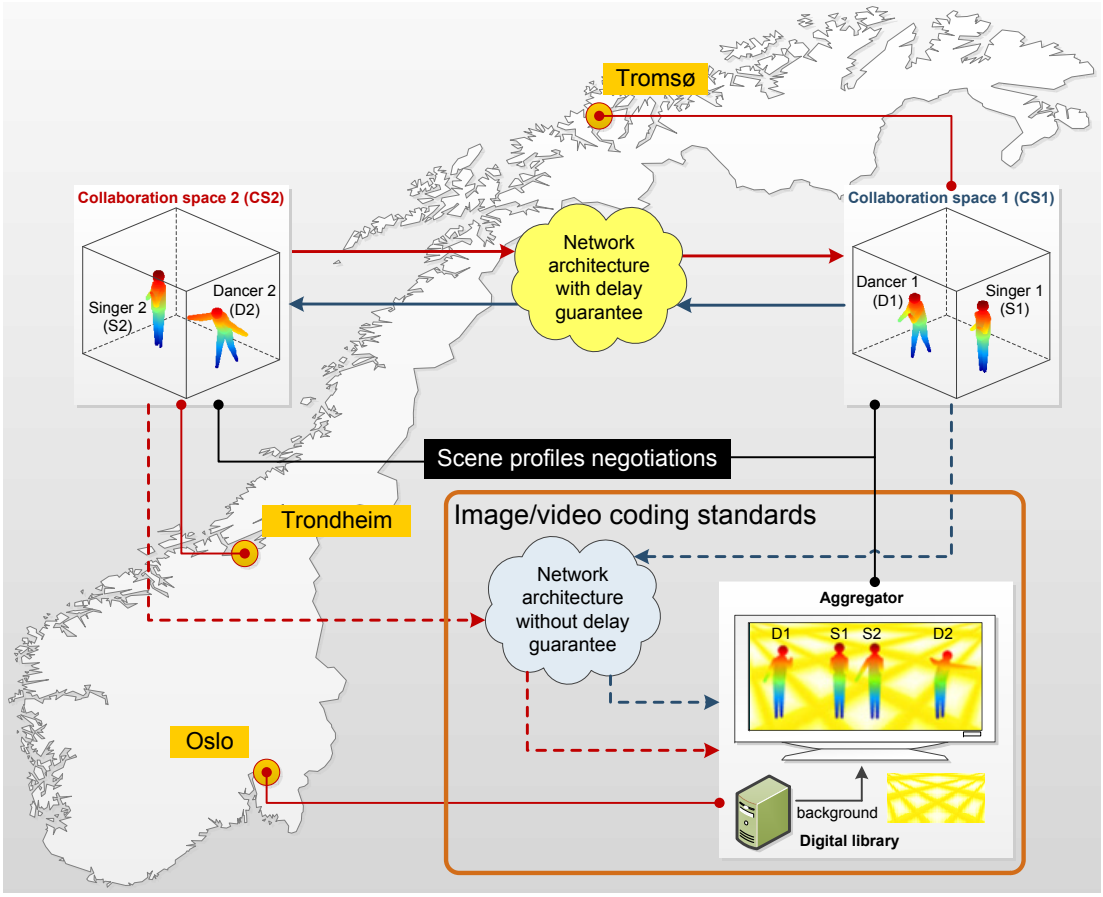


Figure 1.2: The reference collaboration.

1.4 Technical challenges

Some technical challenges in realizing the envisioned collaboration system are identified and highlighted in bold type in this section as logical implications derived from analyzing the reference collaboration. Instead of providing an exhaustive coverage, the goal is to reveal the magnitude of the complexity in addressing the vision.

Let us begin with the aspects that determine the process of achieving the near-natural QoE for the artists. The first is the design and specification of a CS which is shown in Figure 1.2 as a cube. **How to determine the best physical structure of a CS** is an interesting problem.

The artists must see the other pair in the other end displayed with pristine quality on the surfaces of their CS. Consequently, the display must be 3D, multiview, and autostereoscopic because performing by wearing 3D glasses should be avoided. Having acquired by high-end cameras in the CS, the video is processed and transferred to the interconnected CSs in the collaboration. The CS must also be equipped with high-end speakers and microphones for the audio signal. To fully support all possibilities in a collaboration scenario and achieve the immersive QoE, all the surfaces of the CS, including the floor and the ceiling, are tiled with arrays of these devices (Figure 1.3). The challenge in this aspect is **how the devices should be designed and incorporated**

Figure 1.3: Impressions of the displays in a CS [Rønningen (2011b)].

in a CS to achieve the near-natural QoE. Some designs are proposed and detailed in [Rønningen (2012)].

The acquisition and presentation of the video data become more complex at higher spatial and temporal resolution. Rønningen and Heiberg (2009) reported that stereoscopic video at full high-definition (HD) 1080p60 (1920×1080 pixels at 60 Hz with progressive scan) is perceived at substantially lower quality than the corresponding real scenes. With screen update rate of 200 or even 400 frames per second (fps) in current HDTVs, the interlaced video input at only 30 or 60 fps must be interpolated. At least 300 fps is required to reduce the smearing and jerkiness due to very fast motion to be nearly invisible [Armstrong et al. (2008)]. These translate into **production and transmission of video data at extremely high bitrate, even from a CS alone**.

Processing and transporting the data are also major technical challenges. **Investigating creative ways for reducing the tremendous amount of video data** is an interesting open question for research. They include segmenting the important objects and exploiting the eye gazes of the artists which indicate their points of attention [Rønningen (2011b)]. Since the audience in Oslo are interested only in the artists, they must be segmented as objects from the unimportant background. Independent processing and transmission of the objects allow the stage background displayed in Oslo to be changed, as shown by the aggregating display in Figure 1.2. The background in yellow and white is retrieved from a local server in Oslo as a digital backdrop library.

The performance between the duo singers and the dancers is synchronized mainly through their eye gazes, which can be exploited for further data reduction. Assuming that these can be automatically detected in real time, the resulting coordinates can be transmitted and used to activate the correct array of cameras in the other CSs for video recording. The resulting video data are then processed and transported to the CS of the source eye gaze to be displayed on the gazed surface.

The video data from a CS might have many transient periods, which can be caused, for example, by occlusions and the number of users in the CS. The occlusion is related to how they are positioned with respect to each other in the CS when captured by the camera arrays. For instance, when D1 is fully occluded by S1 when S2 looks at S1, then S1 will be the only object in the video captured by the array of cameras activated by S2's

eye gaze. When D1 moves forward or backward very quickly to a position completely free from occlusion by S1, two objects then appear on the captured video. Since the data rate is rapidly doubled because there are two users in the CS, the increase of the data rate is higher with more users.

Furthermore, due to the delay-sensitive activities in the collaboration, **the EED must be maintained below a very low value to establish consistent synchronization between the artists**. The EED is counted from the capture of a video frame by an activated camera array, say in CS1, until the rendering of the received frame on the activated array of display panels in CS2. Chafe et al. (2004) studied the effect of time delay on ensemble accuracy by placing pairs of musicians apart in isolated rooms and asking them to clap a rhythm together. They found that, in musical collaboration, longer delays produce increasingly severe tempo deceleration and shorter delays yield a modest, but surprising acceleration. The optimal delay for the synchronization in their experiment is 11.5ms, the reference EED in this thesis, which is far below than that for a convenient videoconferencing which should not exceed 150ms [ITU-T (2003)]. Similar phenomenon is also evident in recent tests on remote conducting at ITEM NTNU [Conca (2012)], cf. Figure 1.4 which shows a person conducting two singers, who represent a choir, over a network with direct wired connection.

Figure 1.4: Remote-choir conducting test at ITEM NTNU [Conca (2012)].

According to the delay source, Delaney et al. (2006) decomposed an EED into three types: packet processing delay, bit processing delay, and packet propagation delay. The total latency τ_{total} for a single packet is given as:

$$\tau_{\text{total}} = \sum_{i=1}^{N} \tau_{\text{total}}^{i} + \sum_{i=1}^{N+1} \Delta\tau_{i}^{i+1} + \sum_{i=1}^{N-1} \frac{M}{B_{i}^{i+1}} \tag{1.1}$$

where τ_{total}^{i} is the time to process a packet at node i, N refers to the number of nodes (including source and destination nodes), τ_{i}^{i+1} is the transmission time between nodes i and $i+1$, B_{i}^{i+1} denotes the bandwidth between nodes i and $i+1$, and M is the number

of bits in the packet. The three parts in the right-hand side of the equation refers to the three types of delay, respectively.

The packet processing delay is defined as the time taken to manage and process the data as it migrates through the network hardware and to process and parse the data at both source and destination nodes. This includes not only compression, decompression, encryption, and decryption, but also any processing performed by the operating system or network hardware at the end-point computers together with the time delay associated with flow control and congestion control, buffering, and packet queuing. The delay can be reduced, for example, by reducing the quantity of data on the network, increasing the processing power at routers and source/destination nodes, and using more efficient processing algorithms.

In the envisioned collaboration, **every processing step in a CS must be designed and implemented with very fast computation to meet the stringent EED level**. A CS functions both as transmitter and receiver. The processing steps as a transmitter include acquisition and object segmentation, whereas rendering, projection, eye tracking and gaze detection contribute to the processing delay as a receiver. They can be accelerated on parallel platforms such as field-programmable gate arrays (FPGAs) and graphic processing units (GPUs).

The bit propagation delay refers to the delay associated with the physical speed of transmission, which is determined by the given distances and the medium of transmission. It cannot be eliminated and the speed of light in a vacuum imposes the theoretical limit. On the other hand, the packet propagation delay denotes the time required for all bits in a packet to be transmitted across the network from source to destination node considering only the internode bandwidth. Increasing the available network bandwidth and reducing the amount of data to transmit between nodes reduce the packet propagation delay.

The very high video quality (VQ) and the guaranteed EED to achieve the near-natural QoE add significant complexity to the processing of the high-bitrate transient video data. It becomes more complicated when the network capacity changes over time as the network is shared with other services. This situation lead to the next challenges in the following questions. **If the visual quality must be degraded to guarantee the maximum EED, how to do it gracefully?** Since the current Internet cannot guarantee maximum EED and graceful VQ degradation at the same time, **what is the network architecture that can deliver these?**.

The vision and the challenges above have been addressed at ITEM NTNU since 2003 with the concept of the Distributed Multimedia Plays (DMP) system architecture [Rønningen (2011b)]. One of the fundamental ideas in DMP, as detailed in Chapter 3, is that the DMP network nodes can drop video packets fast and intelligently according to the instantaneous network condition to deliver both guarantees. Nevertheless, **if video compression is applied in the collaboration on DMP, what properties must be met by such a compression scheme, and how to design it?** Figure 1.2 shows that existing image/video coding standards can be applied in one-way data communication between the two CSs and the audience in Oslo because EED is not a critical issue. This, however, is not the case for the EED-sensitive interaction between the artists, for example because of the computational demand of the coding algorithms.

1.5 Research questions and their interconnection

Through the evolution of the research during the PhD period, the work has been focused on three research questions (RQs). They were developed over time with a healthy balance of supervision from the supervisors and the candidate's independent thinking. Alignment to the ongoing research on DMP and the general research themes at ITEM NTNU is a key consideration in selecting the RQs.

A system can be studied in at least three ways: measurement, mathematical analysis, and simulation. In this study, data measurement is certainly not an option because the envisioned collaboration system does not exist yet. Analytical solution for such a complex system would be very difficult and requires assumptions that make the reduced system too simplistic. Therefore, simulation has been the main methodology in this research, and measurement from existing collaboration systems can also be useful.

The envisioned collaboration system consists of a set of networked CSs and the DMP network that interconnects them. Networking aspects of the DMP architecture have been through in-depth investigation, mostly by means of discrete event simulation (DES). It is very important to emphasize that this PhD research does not aim at any detailed study on networking aspects of the DMP architecture. What is needed to advance the study and simulation of the DMP network and architecture includes a generator of transient traffic that is expected from the collaboration system. Since the latter is non-existent, it has to be modeled and simulated first, which leads to the transient-traffic synthesis. Because DES has been used for simulating DMP network and architecture, applying DES in the simulation of the collaboration system creates a consistent workflow. Since the input traffic to a DMP network is produced by a set of interconnected CSs, the interactions between performers in a CS and those between CSs must be studied first. Therefore, the first two RQs which are closely related in the PhD work are as follows:

Research Question 1

How to model and simulate the interactions between human performers within a CS and in remotely connected CSs in the envisioned collaboration system in valid ways that are reproducible with exactly the same unique results?

Research Question 2

How to synthesize the appropriate transient traffic from the human interaction in a CS as the input trace to future study and simulation of the DMP architecture?

The solution to RQ-2 is constructed from the results of the answer to RQ-1. Note, however, that how to use the synthesized traffic in simulating the DMP network is outside the scope of this work. Since the solutions to RQ-2 provide the input traffic to DES based on visual signals generated by a CS, the information loss due to packet dropping in DMP can be simulated.

The size or bitrate of the data reduced in this way, however, informs nothing about the VQ. The relationship between video bitrates and the corresponding quantified VQ of the reconstructed image/video can be provided by means of models estimated from rate-distortion (RD) plots in image/video compression. In fact, constructing video

compression schemes that are compatible with the envisioned collaboration system and the DMP architecture is still an open question, as formulated as the third RQ below:

Research Question 3
How to construct video compression schemes that encode and decode video signal from a CS in the envisioned collaboration system by fulfilling the requirements of the DMP architecture?

Figure 1.5 overlays the areas covered by the RQs on the components that they address to show the relationship and coherence between the RQs. They are interrelated in how each of them contributes to advancing the ongoing study and simulation of the DMP architecture towards realizing the vision. The dashed lines with arrows denote input/output relationship and those without arrows represent knowledge transfer. The latter is used in the connection between RQ-3 and existing image/video coding schemes as the knowledge basis for addressing RQ-3.

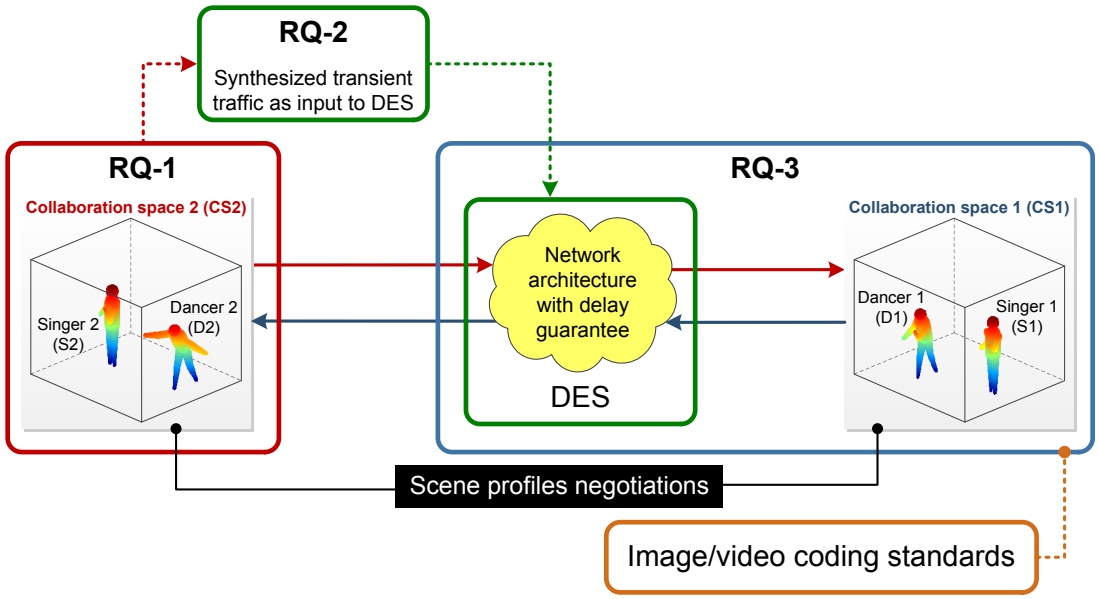


Figure 1.5: The relationship between the RQs and the reference collaboration.

In summary, the solution to RQ-1 simulates both the interaction between users in a CS and the complex collaboration between interconnected CSs. By using this simulation and the results from addressing RQ-2, the traces of the synthesized transient traffic become the input to future simulations of DMP networks. Solutions to RQ-3 benefit from existing video compression schemes that can be used for this work, and the resulting RD curves will enhance the solution to RQ-1 and RQ-2 by enabling VQ estimation due to packet dropping in future DMP simulations.

1.6 Research methodology

As the nature of the PhD work is mainly experimental and focuses on design and engineering issues, it resorts mostly to computer modeling and simulation for all RQs.

Specifically for RQ-1 and RQ-2, DES is the main method for experiment, and DEMOS (Discrete Event Modeling on Simula) is the chosen DES tool because it has been instrumental in research and teaching on wired, wireless and optical networks at ITEM NTNU for decades. By quickly producing reproducible and unique results, DEMOS are used in replications to reduce bias and dependence in statistical results from DES.

Simulation via prototyping is undertaken for RQ-3 using Matlab as the chosen tool due to the complete toolbox needed for research work in signal processing and image/video compression. Whenever necessary and possible, measurements are conducted to gain insight from real data for further analysis and modeling.

RQ-1 implies the analysis of human body and its stochastic/deterministic motion as the starting point. As the work on RQ-3 assesses VQ using only objective quality measures, the use of subjective assessment is reserved for future work. Since processing time in RQ-3 is very critical, the complexity of the proposed solutions is estimated by addressing the HW design for implementation on recent FPGA boards.

1.7 Thesis structure and scientific publications

This thesis is structured into two parts. Part I presents four chapters, including this one, that introduce and summarize the thesis, and Part II contains the included scientific papers resulting from the PhD work. Unlike the chapters in Part I, an included paper in Part II is referred to as 'Paper' with an alphabet, for example 'Paper A'.

Chapter 2 provides a survey of the state of the art to highlight the evolution and main developments in immersive collaboration and with respect to each of the RQs. The descriptive presentation and critical assessment of related work in the last five to ten years position the RQs with respect to the state of the art.

Chapter 3 gives an overview of the DMP architecture that covers the technical details relevant to the RQs. The relationship between the DMP architecture and the state of the art is also discussed, and the DMP architecture is also critically evaluated. Note again that, instead of being aimed directly at the DMP architecture and its improvements, the PhD work focuses on the futuristic collaboration on DMP.

The last in Part I, Chapter 4 highlights the main work and ideas reported in Part II and gives a synopsis of the resulting contributions without the technical details. A visual summary of the contributions for each RQ should help the reader follow the workflow more easily. This chapter is very important to make this thesis accessible to readers who are not familiar with the involved research fields. Furthermore, it offers some ideas for future work that raise from the work on each RQ, which can be viewed as contributions as well because ideas are always at the core of any research.

A PhD program is a comprehensive training towards the graduation of a professional researcher under supervision of experienced and accomplished researchers. The PhD candidate has been given the opportunity to work on a number of ideas and problems that include the three RQs. A fruitful research collaboration with the supervisors, postdoctoral researchers, and fellow PhD and master's students, the PhD work produces 18 peer-reviewed published papers which consist of 4 journal papers and 14 conference papers, as listed chronologically below according to the time of publication.

Journal papers

1. M. Panggabean, M. Wielgosz, H. Øverby, and L.A. Rønningen, "Ultrafast scalable embedded DCT image coding for tele-immersive delay-sensitive collaboration," *International Journal of Advanced Computer Science and Applications*, vol. 4, no. 12, 2013, pp. 202–211 (SAI, 2012 Impact Factor: 1.324).

2. M. Wielgosz, M. Panggabean, and L.A. Rønningen, "FPGA architecture for kriging image interpolation," *International Journal of Advanced Computer Science and Applications*, vol. 4, no. 12, 2013, pp. 193–201 (SAI, 2012 Impact Factor: 1.324).

3. M. Panggabean, L.A. Rønningen, and H. Øverby, "Modeling and simulating motions of human bodies in a futuristic distributed tele-immersive collaboration system for synthesizing transient input traffic," *Simulation Modelling Practice and Theory*, vol. 31, 2012, pp. 132–148. (Elsevier, 2012 Impact Factor: 1.159)

4. M. Wielgosz, M. Panggabean, J. Wang, and L.A. Rønningen, "An FPGA-based platform for a network architecture with delay guarantee," *Journal of Circuits, Systems and Computers*, vol. 22, no. 6, 2013. (World Scientific, 2012 Impact Factor: 0.238)

Conference papers

1. M. Wielgosz, M. Panggabean, A. Chilwan, and L.A. Rønningen, "FPGA-based platform for real-time Internet," in *the Proceedings of 3rd International Conference on Emerging Security Technologies (EST)*, Lisbon, Portugal, September 5-7, 2012.

2. M. Panggabean, "Toward world-class faculty with nationality in Indonesian higher education institutions: a model and paradigm (in Indonesian)," in *the Proceedings of 2nd Olimpiade Karya Tulis Inovatif (OKTI)*, Paris, France, October 8-9, 2011.

3. M. Panggabean, "Indonesia's performance in international scientific publication during 1996-2010: a comprehensive comparative study (in Indonesian)," in *the Proceedings of 2nd Olimpiade Karya Tulis Inovatif (OKTI)*, Paris, France, October 8-9, 2011.

4. M. Panggabean and L.A. Rønningen, "Synthesizing transient traffic of temporal visual signals for discrete event simulation," in *the Proceedings of IEEE 3rd International Congress on Ultra Modern Telecommunications and Control Systems (ICUMT)*, Budapest, Hungary, October 5-7, 2011.

5. M. Panggabean and L.A. Rønningen, "Parameterization of windowed kriging for compression-by-network of natural images," in *the Proceedings of IEEE EURASIP 7th International Symposium on Image and Signal Processing and Analysis (ISPA)*, Dubrovnik, Croatia, September 4-6, 2011.

6. M. Panggabean and L.A. Rønningen, "Chroma interpolation using windowed kriging for color image compression-by-network with guaranteed delay," in *the Proceedings of IEEE EURASIP 17th International Conference on Digital Signal Processing (DSP)*, Corfu, Greece, July 6-8, 2011.

7. M. Panggabean, Ö. Tamer, and L.A. Rønningen, "Parallel image transmission and compression using windowed kriging interpolations," in *the Proceedings of IEEE 10th International Symposium on Signal Processing and Information Technology (ISSPIT)*, Luxor, Egypt, December 15-18, 2010.

8. L.A. Rønningen, M. Panggabean, and Ö. Tamer, "Toward futuristic near-natural collaborations on Distributed Multimedia Plays architecture," in *the Proceedings of IEEE 10th International Symposium on Signal Processing and Information Technology (ISSPIT)*, Luxor, Egypt, December 15-18, 2010.

9. H. Berge, M. Panggabean, and L.A. Rønningen, "Modelling video-quality shaping with interpolation and frame-drop patterns," in *the Proceedings of 23rd Norsk informatikkonferanse (NIK)*, Gjøvik, Norway, November 22-24, 2010.

10. M. Panggabean and L.A. Rønningen, "Resampling HD images with the effects of blur and edges for future musical collaboration," in *the Proceedings of 23rd Norsk informatikkonferanse (NIK)*, Gjøvik, Norway, November 22-24, 2010.

11. Ö. Tamer, L.A. Rønningen, and M. Panggabean, "Real time edge detection using three dimensional systolic array," in *the Proceedings of IASTED 22nd International Conference on Parallel and Distributed Computing and Systems*, Marina del Rey, CA, USA, November 8-10, 2010.

12. M. Panggabean, S. Salater, and L.A. Rønningen, "Eye tracking for foveation video coding and simple scene description," in *the Proceedings of IEEE 2nd International Conference on Image Processing Theory, Tools and Applications (IPTA)*, Paris, France, July 7-10, 2010.

13. M. Panggabean and L.A. Rønningen, "Character recognition of the Batak Toba alphabet using signatures and simplified chain codes," in *the Proceedings of IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, Kuala Lumpur, Malaysia, November 18-19, 2009.

14. M. Panggabean, S. de Waele, G. de Haan, and L.A. Rønningen, "Automatic texture-detection algorithm for texture synthesis in video compression," in *the Proceedings of IASTED 11th International Conference on Signal and Image Processing (SIP)*, Honolulu, HI, USA, August 17-19, 2009.

Six papers from the list (journal papers 1 and 3, and conference papers 4, 5, 6, and 10) are included in Part II based on two criteria. First, the PhD candidate must be the main original contributor and the first author of the paper. Second, the paper must address one of the three RQs and present considerable contributions without significant overlap with other papers. Note that the included papers might be slightly modified from the original manuscripts without changing the meaning to comply with the thesis template/style, improve the language/content, or suppress redundancy. Conference papers 9 and 12 are adapted from two master's theses at ITEM NTNU which address the eye tracking and packet dropping problems in the DMP project. The students also become co-authors, and the adaptation is based on the merit of the contents which are deemed worthy of better access by international scientific community.

CHAPTER 2

A survey of the state of the art

This chapter presents a survey of the developments of immersive collaboration and the state of the art related to the three RQs as the foundation of the PhD work.

2.1 Developments of immersive collaboration

The focus here is on immersive collaboration systems and their applications which have been reported in scientific publications since 2003. They are summarized chronologically starting from the most recent to the best of our knowledge and access range, and contemporary surveys on this topic can be found, e.g. in [Abbasi and Baroudi (2012); Vasudevan et al. (2011); Kurillo and Bajcsy (2012); Wolff et al. (2007)]. The goal of this non-exhaustive survey is to illustrate how the systems and the applications have been developed over the last ten years towards the vision.

UltraGrid, a high-quality video transmission SW [Holub et al. (2012)], is claimed as the first system to support gigabit rate HD interactive video conferencing on commodity systems and networks. Using a commodity PC and Mac HW, it allows for video transmission with resolutions ranging through HD up to 4320p with EED as low as 75 ms. The high-quality is achieved by transmitting video streams without compression or with low compression ratio. Perkins et al. (2002) originally developed it to demonstrate viability of uncompressed 720p HD video delivery in IP networks for possible applications in collaborative environments [Perkins and Gharai (2004)].

The Cave Automatic Virtual Environment (CAVE) is an immersive VR environment that has been developed since 1992 [Cruz-Neira et al. (1992)]. With a cost of about $ 1 million, the original environment is a 3×3×3-meter room within a room with a resolution of 4 Megapixels. Projection screens make the walls and floor to which high-resolution projectors are directed, and users inside the CAVE wear special glasses to view the generated 3D graphics. Sensors track the movements of the users as the basis to adjust the projected video, and multiple speakers placed in multiple angles in the CAVE provide 3D sound. The development of the CAVE over two decades and its future are presented in [DeFanti et al. (2011)].

CAVE2, also called the Next-Generation CAVE, is a scientific instrument that enables researchers to visualize data in a fully immersive 3D stereoscopic environment [EVL

UIC (2012c)]. Approximately 7.2 meters in diameter and 2.4 meters tall, the prototype design consists of 72 near-seamless passive stereo off-axis-optimized 3D LCD (liquid crystal display) panels, a 36-node high-performance computer cluster, a 20-speaker surround audio system, a 10-camera optical tracking system and a 100 Gbps connection to the outside world (Figure 2.1). CAVE2 provides users with a 320-degree panoramic environment for displaying information at 37 Megapixels in 3D or 74 Megapixels in 2D with a horizontal visual acuity of 20/20, almost 10 times the 3D resolution of the original CAVE. This is achieved by using SAGE (Scalable Adaptive Graphics Environment), the near-seamless, flat-panel, passive-stereo, LCD technology to create a multi-person collaborative space. Figure 2.2 shows the interior of the environment.

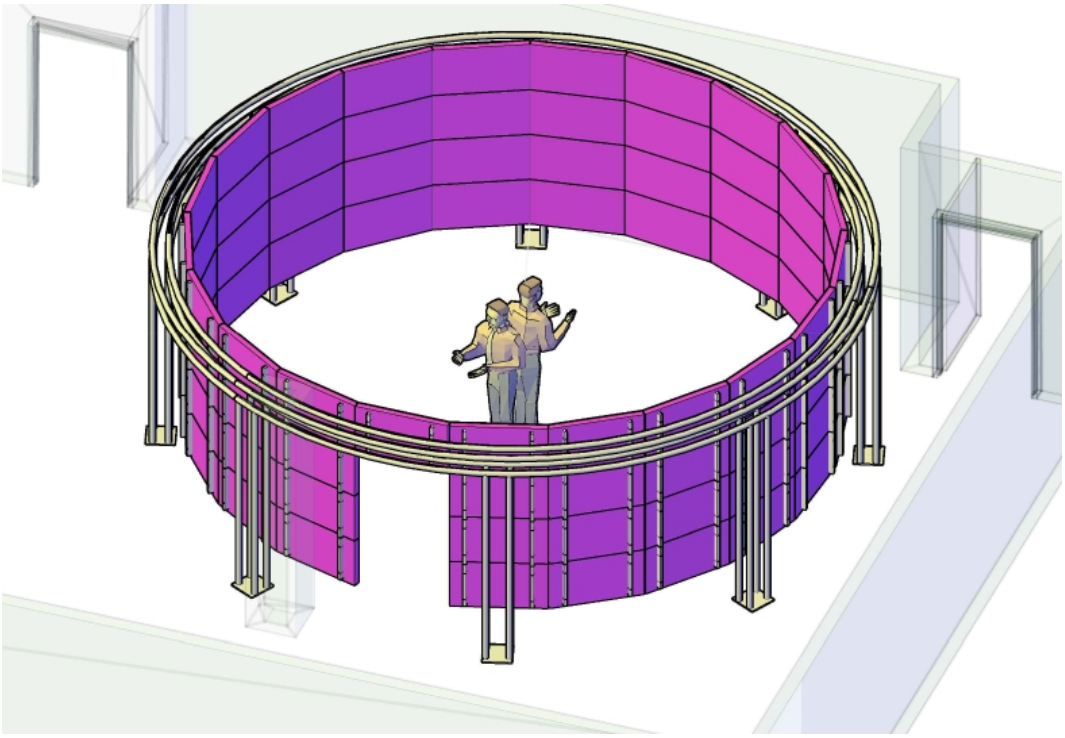


Figure 2.1: Prototype design of the CAVE2 [EVL UIC (2012c)].

Figure 2.2: Snapshots of the CAVE2 interior [EVL UIC (2012a,b)].

Systems for multi-camera telepresence that utilize large high-resolution displays generate large amounts of dynamic data which must be reduced due to limited resources and many phases for processing and transmission. Willert et al. (2012) analyzed and evaluated some solutions to the bandwidth optimization problem in networked multiview camera setups without lossy compression and with low latency.

Vasudevan et al. (2011) described a multi-camera system that creates a highly accurate (on the order of a cm), real-time (under 30 ms), 360-degree 3D reconstruction of users. It addresses the problem of accurately constructing 3D data at high frame rates to enable fully 3D tele-immersive system.

An enhanced telepresence system, developed from the previous version [Maimone and Fuchs (2011)], offers fully dynamic, real-time 3D scene capture and continuous-viewpoint, head-tracked stereo 3D display without requiring the user to wear any tracking or viewing apparatus [Maimone et al. (2012)]. The complete SW and HW framework for implementing the system is based on an array of commodity Microsoft Kinect color-plus-depth cameras. Being able to capture a fully dynamic 3D scene with the size of a cubicle, it allows a remote user to look around the scene from any viewpoint.

Kurillo and Bajcsy (2012) presented research work at University of California at Berkeley in the last several years in developing a 3D tele-immersive technology using image-based stereo and, more recently, Kinect together with several applications. Besides addressing the issues on acquisition, transmission, rendering, and interaction, they also exhibited the application for remote interaction and collaboration among professional and scientific users.

Zhang and Ho (2012) proposed a prototype of mixed reality system for tele-immersive applications by combining real-time 3D modeling and human-agent interaction. A mesh model is reconstructed and then loaded into a dynamic virtual environment, coupled with a created autonomous virtual agent. Tele-immersive user-user and user-agent remote interactions are achievable through the proposed system.

A cost-effective immersive gaming environment is implemented in Blender, an open-source game engine [Bourke and Felinto (2010)]. Focusing on seamless hemispherical displays, planetariums in general, and the iDome in particular, it extends the traditional approaches to immersive gaming that tend to emphasize multiple flat screens or cylindrical displays.

The study of Lu et al. (2010) on four representative video conferencing applications reveals their characteristics and different QoE aspects. They recommended the incorporation of two aspects when designing such applications. First, traffic load control/balancing algorithms to better use the limited bandwidth resources and to have a stable conversation. Second, the use of traffic shaping policy or adaptive real-time stream re-encoding to limit the overall traffic.

In the tele-immersive system based on multi-camera 3D modeling in [Petit et al. (2010)], users from distant sites are immersed in a rich virtual environment served by a parallel terrain rendering engine. Distant users, present through their 3D model, can perform some local interactions while having a strong visual presence. Experimenting the system between three large cities a few hundreds kilometers apart from each other, this work demonstrates the feasibility of a rich 3D multimedia environment ensuring users a strong sense of presence.

Based on distributed wireless sensor networks, the ASPIRE (Collaborative Signal Processing for Efficient Wireless Sensor Networks) project explores the fundamental challenges of deploying sensor networks for immersive multimedia, with concentration on multichannel audio capture, representation, and transmission [Mouchtaris and Tsakalides (2009)]. The interesting ideas include a user's immersive presence in a concert hall performance in real time, virtual music performances, and collaborative environments for music production.

Lien et al. (2009) introduced a skeleton-based compression method using motion estimation in which kinematic parameters of the human body are extracted from the point cloud data in each frame. It provides high and flexible compression ratios from 50:1 to 5000:1 with reasonable reconstruction quality (PSNR from 28 to 31 dB) while preserving real-time (10+ fps) processing.

SAGE was started in 2004 with LamdaVision which consists of 11×5 tiled display with a total resolution of 100 Megapixels [Renambot et al. (2009); Naveen et al. (2004)] (Figure 2.3). By allowing the seamless display of various networked applications over the high-resolution displays, the goal is to enable scientists to explore datasets and share applications at the highest resolution available. SAGE is claimed to provide a solution to heterogeneity problem and offers scalability with extremely fast access to huge databases at remote or local sites by taking advantage of affordable ultra-high-bandwidth networks.

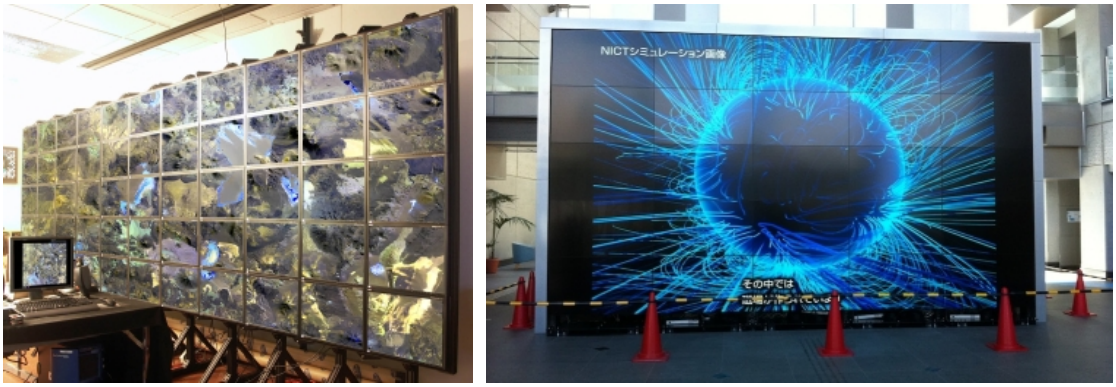


Figure 2.3: LambdaVision in 2004 (left) and a SAGE display in 2011 (right) [EVL UIC (2012c)].

The networked virtual environment for globally distributed avionics SW development in [Bartholomew (2008)] is based on observations that improving the speed and cost of SW development distributed across a global team requires remediation, such as the acquisition and deployment of collaboration technology. The system involves 4 continents, 7 countries, and 20 regional facilities.

3DPresence is a European FP7 research project to realize a high-end 3D AV conference that provides an experience of presence [3DPresence (2008); Schreer et al. (2008); Feldmann et al. (2009)]. It includes a mock-up system which uses novel multiview 3D display and multiview camera system that enable life size of objects, eye contact and gesture awareness in multiparty, multiuser shared-table concept.

Having been developed since 2004, a distributed 3D tele-immersive system called TEEVE (Tele-immersive Environments for Everybody) captures 3D full-body human

motion in geographically distributed sites, aggregates the video data across the Internet, and visualizes them in a joint "cyber-space" at each site through which the users can interact or collaborate in real time [Wu et al. (2008)]. An overview of the system with emphasis on the vision aspects is presented in [Jung and Bajcsy (2006)], and [Yang et al. (2005)] covers the data adaptation and networking aspects. The view dissemination and management among multiple sites is handled by the ViewCast protocol proposed in [Yang et al. (2007)]. Learning Tai-Chi [Kurillo et al. (2008)] and tele-immersive dance [Sheppard et al. (2008)] are some of its applications. In the latter, TEEVE provides better learning of physical movements than a 2D video [Patel et al. (2006)].

An experimental framework of tele-immersive system by [Leong et al. (2008)] is based on polyhedral visual hull algorithm and consists of local and remote sites. Images of the users are taken at the local site from a set of synchronized cameras and then the 3D mesh is reconstructed using the algorithm. The texture information and the mesh model are sent to the remote site for rendering on immersive walls.

Ishida et al. (2008) proposed a tele-immersive multimodal communications system by connecting immersive displays and tiled displays over Japan Gigabit Network 2 on the next generation network. Much cheaper than the expensive CAVE system, it provides a shared virtual space targeted for collaboration in the traditional Japanese crafting. Moreover, the human sensibility so called 'Kansei' in Japanese could be reflected to the tele-immersive environment to improve non-verbal communication capability.

Ebara et al. (2007) described the construction of tele-immersive collaborative environment with scalable tiled displays which consist of 4×2 arrays of LCD panels with one master node and four display nodes connected by Gigabit Ethernet network. SAGE is applied to deliver streaming pixel data with virtual high-resolution frame and to buffer a number of graphical sources for tiled displays. By locating a camera at the center of the tiled displays environment, the authors examined the possibility of remote communication that eye-to-eye contact can realize with each other.

Relying on 2D stereo-based video avatars, the novel low-cost method for visual communication and telepresence in a CAVE-like environment in Rhee et al. (2007) provides convincing stereoscopic real-time representation of a remote user acquired in a spatially immersive display. Two color cameras and two additional B/W cameras are used for infrared-based image segmentation. The compressed stereo images are streamed across a network and displayed as a frame-sequential stereo texture on a billboard in the remote virtual environment.

The Sarcos Treadport experimental platform [Hollerbach (2007)] is a unique locomotion interface that consists of a large tilting treadmill, an active mechanical tether, and a visual CAVE-like display. The locomotion experience is made as realistic as possible through mechanical and perceptual aspects of the locomotion interfaces.

Lunenburger et al. (2007) described a system that combines immersive virtual environments with robot-aided gait training to enhance the performance of patients and the quality of rehabilitation. Besides assessing a patient's movements and providing (bio-)feedback to the person, it also delivers instructions and increases motivation by allowing the patient to navigate through exchangeable virtual environments by modulating the performance of the left and right legs.

Bailenson et al. (2006) explored the possibilities and implications of employing

immersive virtual environments for courtroom training and practice with some applications. The motivating argument is that the created immersive and interactive reality adds significant value as a simulation of experience to enhance courtroom practice.

Therapeutic hysteroscopy is a common technique in gynecological practice, yet with a number of potentially dangerous complications. Specialized training is needed to reduce the rate of complications, and surgical simulation based on VR is an option to provide a corresponding learning environment. Harders et al. (2006) presented a highly-realistic and immersive training environment for hysteroscopy as a generic surgical training simulator.

The high-quality collaborative environment in [Holub et al. (2006)] uses HD video to achieve near realistic perception of a remote site. The capture part consists of an HD camera, a Centaurus HD-SDI capture card, and the UltraGrid SW. A 1.5 Gbps UDP data stream of uncompressed HD video is produced and transferred over a 10GE network interface to the high-speed IP network. A user-controlled UDP packet reflector is responsible for distributing the data to individual participants of the videoconference. The system has been demonstrated in a three-way high-quality videoconference among sites in the Czech Republic, Louisiana, and California.

Lee et al. (2006) implemented a real-time dynamic 3D avatar from multiview cameras for tele-immersive communication. The space includes a spherical display system, a motion simulator, and a 3D sound system. Active segmentation method is proposed to generate the user's video avatar in real time with a high sense of presence in the 3D shared virtual world. A given exemplary scenario namely "Heritage Alive" enables interactive group exploration of cultural heritage in tangible space.

The virtual-studio approach framework in [Kim et al. (2006)] facilitates actors/users to interact in 3D systems more naturally with the synthetic environment and objects by employing personal computers (PCs) and inexpensive special cameras. It is targeted for producing movies in which real life actors "seem" to act and interact with the synthetic characters. In addition to broadcast production, it can also be used for creating virtual/augmented-reality environments for training and entertainment.

Distributed Immersive Performance (DIP) is a real-time, multi-site, interactive, and collaborative environment which has been developed and extensively tested since 2003 as a technology for live, interactive musical performances. The architecture, technology, and experimental applications are introduced in [Sawchuk et al. (2003); Zimmermann et al. (2008)]. The participants are in different physical locations and interconnected by very high fidelity multi-channel audio and video links.

Baker et al. (2005) described a multi-user immersive remote teleconferencing system called Coliseum which is designed to provide collaborative workers the experience of face-to-face meetings from their desktops. Each PC display in the system has five cameras directed at the participant. The proposed view-synthesis methods produce arbitrary-perspective renderings of the participant from the resulting video streams and transmit them to others at interactive rates, about 15 fps. The system is shown to support virtual mobility as participants move around the shared space with reciprocal gaze. It has been demonstrated in collaborative sessions of up to ten Coliseum workstations which span two continents. Different aspects of resource, network, CPU, memory, and disk usage are measured and evaluated to uncover the bottlenecks, guide enhancement,

and control of system performance.

In the collaborative infrastructure called Scape (stereoscopic collaboration in augmented and projective environments) [Hua et al. (2004)], the core display components are multiple custom-designed prototypes of projective HMDs. The ubiquitous projective environment mainly consists of a 1×1.5-meter workbench, a 3.6×3.6×2.7-meter four-walled mural display surrounding the workbench, and selected object surfaces in the room. They claim that the system provides a shared space in which multiple users (a theoretically arbitrary number) can concurrently observe and equally interact with a 3D synthetic environment. They view the space from their individual perspectives, while face-to-face cooperation is preserved.

The novel telepresence system in [Ikeda et al. (2004)] enables users to walk through a photorealistic virtualized environment by actual walking. To provide users with rich sense of walking in a remote site, a wide-angle high-resolution movie is projected on an immersive multi-screen display, and a treadmill is controlled according to detected user's locomotion. An omnidirectional multi-camera system is used to acquire images of a real outdoor scene.

Kim et al. (2004) demonstrated and experimented with haptic interaction between two users over a network of significant physical distance and a number of network hops. A number of techniques are presented to mitigate instability of the haptic interactions induced by network latency, and the use of haptics in a collaborative situation mediated by a networked virtual environment is also evaluated. Using the described technology, transatlantic touch was successfully demonstrated between two places at USA and UK.

Eisert (2003) presented a next generation 3D video conferencing system that provides immersive tele-presence and natural representation of all participants in a shared virtual meeting space. Based on the principle of a shared virtual table environment, correct eye contact and gesture reproduction are guaranteed that enhance the quality of human-centered communication. Instead of video streams, facial expression and motion information using explicit 3D head models are transmitted to achieve low bitrates of a few kbps per user. It enables new possibilities for image enhancements such as digital make-up, digital dressing, or modification of scene lighting.

Jaynes et al. (2003) described the Metaverse, a tele-immersive system based on commodity components that automatically configures and then monitors itself so that it can detect changes to the environment that would require reconfiguration. The objective is to provide users with an open, untethered, immersive environment that fools their visual senses. To simplify setup and maintenance, it particularly supports automatic self-calibration of an arbitrary number of projectors and facilitates novel communication models that enhance the system scalability.

Blue-c is an immersive projection and 3D video acquisition environment for virtual design and collaboration [Gross et al. (2003)] (Figure 2.4). To create the impression of total immersion, simultaneous acquisition of multiple live video streams is combined with advanced 3D projection technology in a CAVE-like environment. The portal consists of three rectangular projection screens built from glass panels containing liquid crystal layers, and active stereo using two LCD projectors per screen is used for projection. From multiple video streams, a 3D video representation of the user is computed in real time and integrated into a networked virtual environment. Portals with less

sophisticated HW can be easily connected to blue-c due to scalable design.

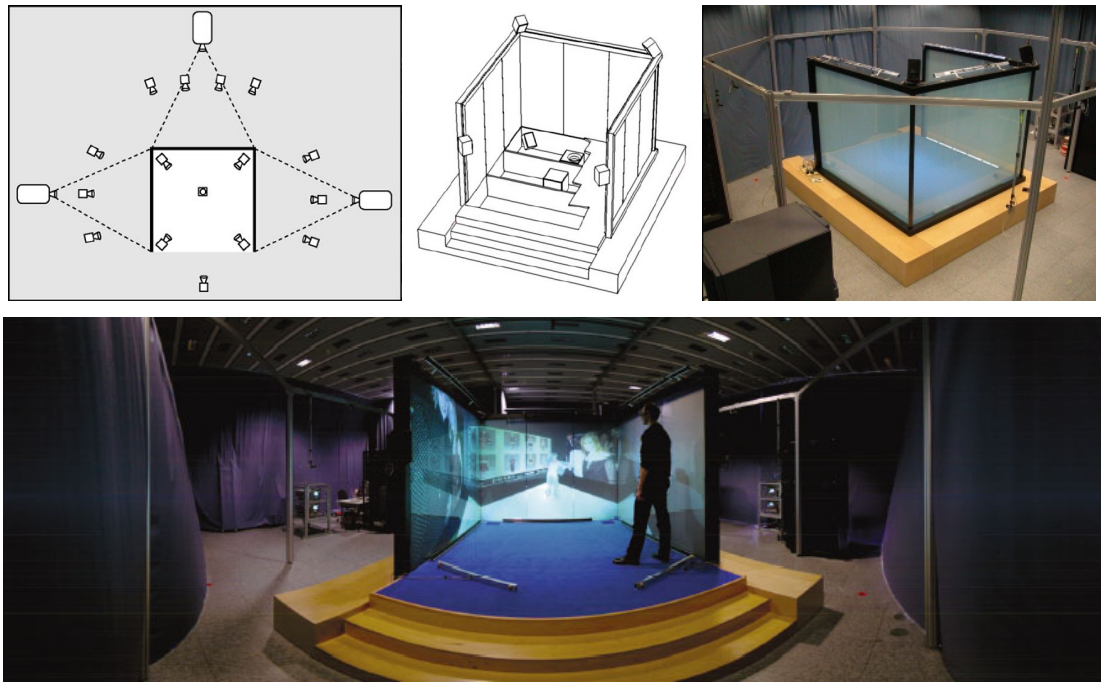


Figure 2.4: The first blue-c portal. Top: Camera arrangement (left), design (middle), and installation (right). Bottom: the portal [ETH Zurich (2003)].

2.2 State of the art for Research Question 1

The envisioned collaboration essentially can be viewed as inter-human and human-machine interactions. In the reference collaboration, the interaction between humans, or social interaction, is represented by the collaboration between a pair of a singer and a dancer in a CS as well as between pairs in the interconnected CSs. The use of CS as means of collaboration sets the role of human-machine interaction.

DES and agent-based modeling are important in social interaction and usually coupled with animation and visualization. According to Allen (2011), they are two of six types of computer simulation related to emergent system response prediction as a tool for system improvement (Table 2.1). "Low-level" data are numbers that relate to specific subsystems, e.g. service times and waiting times.

DES usually involves generic entities that follow rules dictated by system agents or processes, e.g. servers which change their status and cause the entities to wait. Thus the entities do not make decisions nor adapt to changing circumstances. Although many problems in routine systems fit such situation, sometimes the entities need to learn and make choices. This motivates agent-based modeling which is defined as the activity of simulating system-wide properties as they derive from the actions and interactions of autonomous entities. In agent-based models, the agents have decision-making rules along with learning rules or adaptive processes. It focuses on the individuals and pairwise interactions between them.

Table 2.1: Selected types of simulation and how they operate [Allen (2011)]

Type	Engine	Particular relevance and data source
Agent-based	Agent rule iteration and Monte Carlo	For studying incentives and restrictions; based on low-level data
DES	Event controller and Monte Carlo	For studying production systems; based on low-level data
Forecasting	Empirical modeling including least squares	For predicting new emergent properties; based on high-level data
Markov chain models	Linear algebra	Relatively simple and transparent; based on expert opinion and/or high-level data
Systems dynamics models	Differential equation numerical solvers	For studying the impacts of decisions; based on expert opinion
Physics-based	Finite element methods (FEM) numerical solvers	For designing engineered products; based on low-level data

The degree of autonomy of agents in the models may varies. In DES, the entities generally have low levels of autonomy since they rarely change state or learn based on model conditions and follow tightly controlled paths. DES can then be viewed as a type of low autonomy agent-based modeling. In short, agent-based modeling and DES differ largely in emphasis and, to a lesser extent, in structure. If only a few agents can act at a given time, then DES is simply more computationally efficient than iterating through all possibly relevant agents since the event controller goes immediately to the next entity that can act. This affirms the selection of DES as the chosen methodology for this PhD work, as stated in Chapter 1.

Allen (2011) also provided many examples of applications of DES and agent-based modeling in voting systems, health care, military, and manufacturing, as reported in literature. In particular, human motion has been simulated with different environments and goals, such as in virtual simulation for better manufacturing processes [Fuo and Wang (2012)] and for simulating real-world vehicle-pedestrian impacts [Ramamurthy et al. (2012)]. However, to the best of our knowledge, there is no work that addresses a problem that is the same with or similar to that in RQ-1.

2.3 State of the art for Research Question 2

The void in the reported work closely related to RQ-1 also means that RQ-2 has a unique position not addressed until now. Nevertheless, some work have been reported on the problem of transient traffic modeling and synthesis, which is not strictly limited to tele-immersive collaboration.

In related study on transient video traffic, videoconference can be used to approximate the envisioned collaboration. Besides providing a complete outline of the previous work related to the topic, Domoxoudis et al. (2013) studied the characteristics of video traffic from videoconference applications from H.261 to H.264. Based on the analysis

of videoconference data from measurements during realistic point-to-point videoconference sessions, the traffic can be distinguished as unconstrained and constrained. In the unconstrained traffic, a direct relation exists between the encoder and the form of the frequency histogram of the frame-size sequence, and strong correlations between successive video frames can be found. In the second type, bandwidth constraints are imposed during the encoding process, and the generated traffic appears to show similar characteristics for all the examined encoders. The very low autocorrelation values are the most notable.

To study multiplexed traffic from H.264/AVC videoconference streams, Lazaris and Koutsakis (2010) modeled the statistical behavior of bursty video traffic originating from single H.264/AVC streams with well-known distributions. The Pearson type V distribution is shown to provide the relatively best fit for modeling videoconference traffic from single H.264 sources. This is the basis for proposing a Discrete Autoregressive model (separately for I, B, and P frames) that captures well the behaviour of multiplexed H.264/AVC videoconference sources with variable bit rate (VBR).

High frame-rate cameras are planned to be incorporated in the envisioned collaboration system. Two mathematical models that describe the relationship between frame-rate and bitrate for video over 1000 Hz are proposed in [Bandoh et al. (2010)]. The first model corresponds to temporal sub-sampling by frame skip, and the second one models temporal down-sampling by mean filtering which triggers the integral phenomenon that occurs when the frame-rate is downsampled.

2.4 State of the art for Research Question 3

Low end-to-end latency signal transmission is usually limited to 150 ms to achieve transparent perception [ITU-T (2003)]. For 1080p video, this threshold can be achieved on commodity computer HW when the video is sent uncompressed [Perkins and Gharai (2004); Holub et al. (2006); Jo et al. (2006)]. Some specific applications require even lower latency, e.g. geographically-distributed performance of chamber music which needs 10-40 ms latency range [Buso and Allochio (2012); Brock et al. (2011)]. This, however, requires custom HW designs if both audio and high-resolution video are to be delivered at the same latency [Buso and Allochio (2012); Shirai et al. (2009); Halak et al. (2011)].

Low-latency transmission of compressed high-resolution video often requires custom HW support, most often using FPGA, such as 4K JPEG2000 [Shimizu et al. (2006)], 8K JPEG2000 [Kitamura et al. (2011)], or layered multipoint 2K/4K JPEG2000 [Shirai et al. (2011)]. According to the computing power of recent commodity CPUs, the use of rather simple compression schemes is favored for interactive network transmission of high-resolution images, with either low compression ratio or significant image deterioration. The examples include CPU optimizations of DXT1 compression [van Waveren (2006); Renambot et al. (2007)], which supports up to 4K at 25 fps resolution at the expense of significant image-quality degradation, such as strong posterization of long gradients, heavy staircase effects, and noise on sharp edges. DXT and JPEG compression schemes are accelerated using GPU for low-latency network transmissions of HD, 2K, and 4K video [Holub et al. (2013)].

JPEG 2000 is an image compression standard and coding system based on wavelet transform that is proposed to supersede the JPEG standard [Schelkens et al. (2009); Taubman and Marcellin (2001); Taubman (2000)]. It offers a number of improvements over JPEG, for example: superior compression performance; multiple resolution representation; progressive transmission by pixel and resolution accuracy; spatial, quality and channel scalability; support of lossless and lossy compression; embedded coding; facilitated processing of regions of interest; error resilience. Figure 2.5 illustrates the scalability of JPEG 2000 in resolution and quality, and the encoding and decoding of JPEG 2000 are illustrated in Figure 2.6. These improvements, however, make JPEG 2000 encoders and decoders more complex and computationally demanding than those of JPEG. Hence, major applications of JPEG 2000 are in digital cinema.

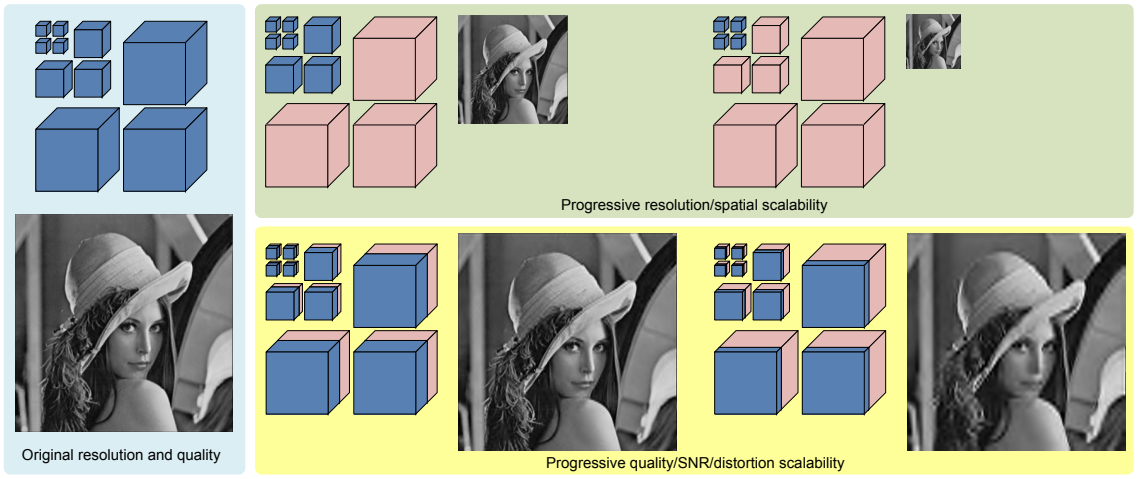


Figure 2.5: JPEG 2000 progressive scalability in resolution and quality.

A situation similar to the vision can be found in processing and coding textures for 2D/3D graphics. Visual details are added to geometric shapes by applying digitized images called textures, and a tremendous amount of detail is mapped onto geometric shapes during rasterization in today's computer graphics. Textures are used not only with colors, but also surface properties (e.g. reflection) or fine surface details.

The large amount of system and video memory consumed by these textures can be reduced by using S3TC or DXT compression supported by most graphics cards. A relatively simple lossy compression scheme with fixed compression ratio based on Block Truncation Coding (BTC) [Delp and Mitchell (1979)], the scheme divides a picture into 4×4 pixel blocks and processes them independently into 64-bit or 128-bit code words. Compressed textures require significantly less memory on the graphics card and render faster than uncompressed textures due to reduced bandwidth requirements. Since the reduced memory footprint allows the use of higher resolution textures, a significant gain in quality is possible. Although the decompression is very simple and real-time during rasterization, the DXT compression is computationally intensive; hence the acceleration using GPU.

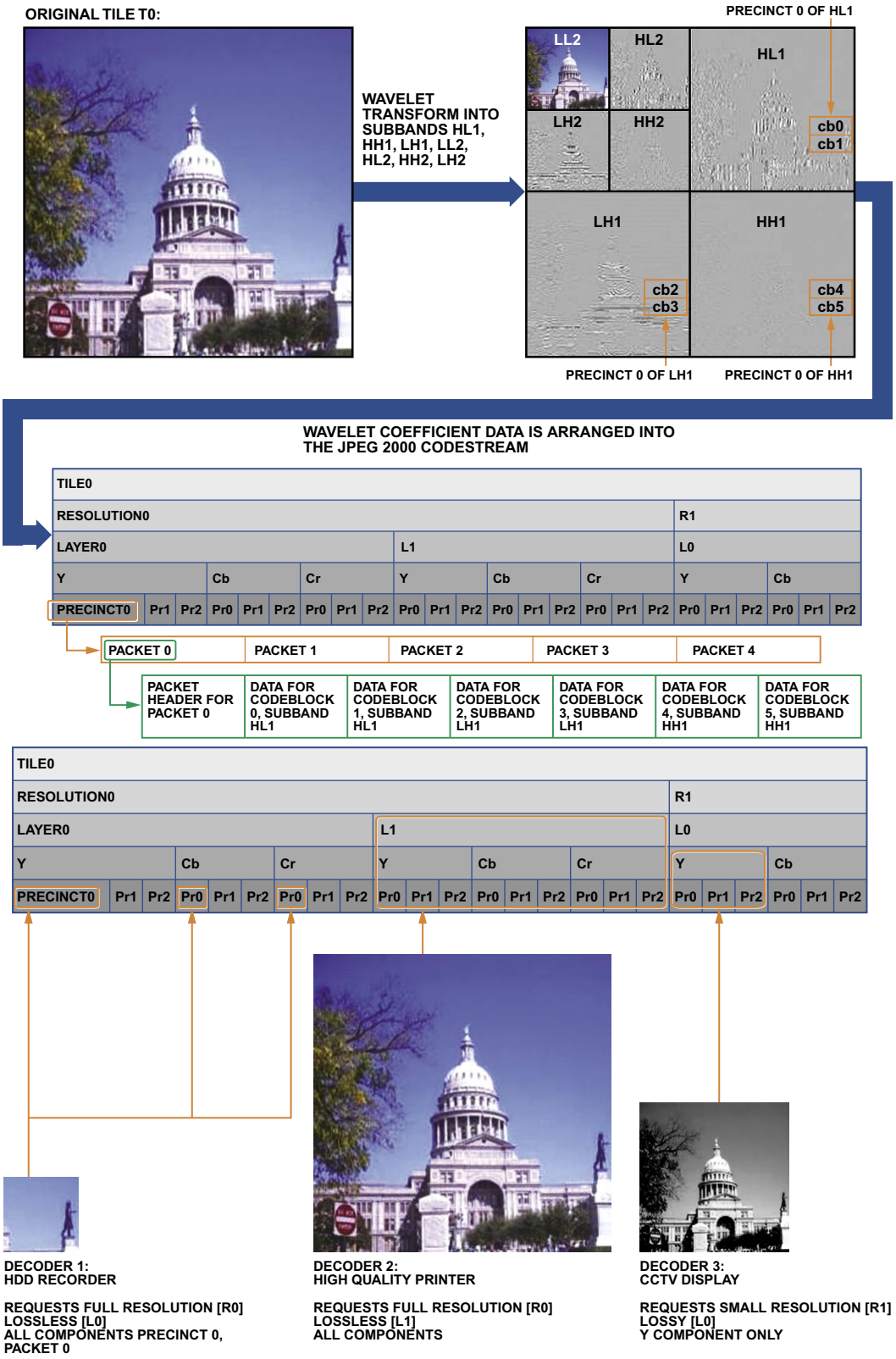


Figure 2.6: JPEG 2000 encoding and decoding [Bako (2004)].

CHAPTER 3

An overview of Distributed Multimedia Plays architecture

This chapter gives an overview of the original technical ideas of the DMP architecture which are relevant to the PhD work by summarizing directly from the most recent documents on DMP [Rønningen (2011b); Rønningen and Chilwan (2012)]. The relationship of DMP to the state of the art and to the RQs are also presented.

3.1 The essentials of DMP

This section gives a brief introduction, definition, objective, and main features of DMP, particularly on the imaging and networking aspects.

3.1.1 DMP in a nutshell

The concept of DMP was first proposed as an extension to the coming digital TV system called Multimedia Home Platform (MHP) in a Telenor project in 1996-1999 [Rønningen (1999)], and the original aim was to enhance the existing and coming TV systems with new features. It is now a three-layer system architecture that provides near-natural virtual networked autostereoscopic multiview video and multichannel sound collaboration between players as well as between players and servers.

The scenes in DMP are composed of real and virtual (stored or generated) sub-scenes, and the smallest real entity in a scene or a sub-scene is called object. After being shot by a camera array, an object can be subdivided further into sub-objects which can be processed, encoded, and transmitted independently. Thus, users can directly select which level of quality from the large stream of sub-objects.

To approach the natural level of human perception, the AV quality must be increased to levels that temporarily require data rates which could be more that three orders of magnitude larger than that in existing telepresence systems. The dynamics in a collaboration also results in extremely variable and transient traffic. The scene quality must adapt to give graceful degradation without exceeding a minimum level when traffic overloads the network or system components fail. The proposed adaptation and quality

control scheme namely Quality Shaping (QS) handles this process. It works by means of controlled dropping of packets in DMP network and at DMP Access Nodes (ANs) based on the feedback of measured load in DMP Network Nodes (NNs). Admission control is needed to guarantee the minimum AV quality. Besides guaranteeing maximum EED without any resource reservation, the QS also prioritizes and guarantees the delivery sequence of packets from end to end.

Scenes in DMP are described in the Scene Profiles (SPs), and the variable parameters in a collaboration are detailed in a Quality Shaping Profile (QSP); both profiles are transmitted over a DMP network. A QSP, which is a subset of an SP, includes several classes of objects with predefined quality parameters which can be controlled adaptively such as the EED, the number of 3D scene sub-objects, the temporal and spatial resolution, the adaptive and scalable compression of sub-objects by the proposed Near-natural Object Coding (NOC), and the number of spatial views. The integrated scene composition and the QS scheme use traffic classes, measurements and forecasting of traffic, feedback control, and traffic and scene quality shaping. QS uses the SPs and QSPs, and users can only change DMP parameters according to the QSP.

Three new layers and a novel network protocol have been proposed and defined to support QS. Being highly efficient for HW implementation, they simplify the necessary processes and extend the quality compared to existing collaborative systems. The layers are the Physical-Link layer, the AppTraNetLFC layer, and the Application layer. Figure 3.1 depicts the layers at ANs, NNs, and CSs.

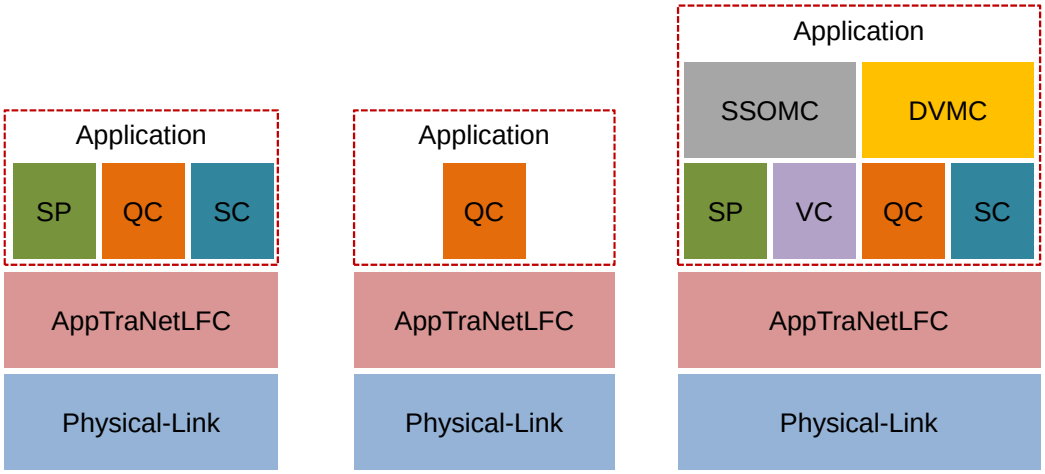


Figure 3.1: DMP architecture at ANs (left), NNs (middle), and CS at user's site (right).

Using IPSec for security, DMP gives more responsibility to the providers of network/service and reduces the ability of users to make changes in other user's control premises. Thus, in addition to the services, DMP provides new opportunities for making business with the network itself. CSs, ANs with various servers, and NNs must be installed in well-planned network structures to provide the DMP services and implement the QS concept. Actual DMP applications include next-generation TV systems, immersive games, long-distance education, virtual meetings, and artistic collaborations such as jazz sessions, song lessons, and distributed opera.

3.1.2 Imaging aspects

The imaging aspects discussed here are the concepts of SP, QSP, and the object-oriented scenes and compression.

Scene Profiles and Quality Shaping Profiles

SPs describe the characteristics and limits of the shooting and presentation of scenes, sub-scenes, objects and sub-objects, movements of objects, and characteristics of each sub-object. During a collaboration, scenes may vary fast, objects appear and disappear, and the importance of the objects changes. The scene dynamics is within the limits in the SP which are determined by physical constraints as well as technical specifications of equipment for shooting and display; otherwise, the events are not shot and presented.

Each of the CSs interconnected in a collaboration exchanges and negotiates its SP to each other before the collaboration begins. For example, the SP of CS1 in the reference collaboration includes the following: one singer and one dancer as the two person-objects to be tracked; each person object consists of the very important face object and the important body object; the person objects are projected side-by-side; the background object is replaceable with synthetic backdrop from a library server at a given address; 2 × 6-camera array is used against a blue wall; five horizontal views and two vertical views are used; the artists are not closer than 1m from the screen and the camera arrays; nine sub-objects are used; a pre-defined compression scheme is given for the face and body objects.

A QSP is based on some parameters that describe the temporal and spatial scene resolution and composition, such as the number of sub-scenes of a scene, the number of objects per sub-scene, the number of stereoscopic views per object, the number of sub-objects per object, the update rate of each sub-object, the channels representing the sub-object, the alpha channel depth, the sample-depth of each channel, the sampling rate of each channel, the shape and size of the sub-objects and the shape masks, and the compression and coding scheme for the sub-objects. The following is an example of a QSP. To obtain near-natural perception of the content, an important object such as the human body should meet the following requirements: YCbCbr with [16,12,12] bits resolution, 4:2:2 sampling, adaptable from [12,10,10] to [16,12,12]; nine sub-objects per object, pixel size 0.25×0.25 mm, 1 to 8 sub-objects can be dropped; 5 views, adaptive 1 to 5 views (determined by the number of viewers at the receiving end); Compression: 5 times, lossless NOC scheme; 5 ms temporal update over network, adaptive 5 to 10 ms. The QSP for the audio may include the related parameters such as stereo, no compression, and scalable 20-bit resolution and 96 kHz sampling. It yields the audio data at 3.84 Mbps before any error control or overhead is added.

Object-oriented scenes and compression

Figure 3.2 describes the concepts of scene, object, sub-object and shape mask through an example. The exemplary scene consists of three objects: the background, the face, and the body. The latter two are denoted as important objects, and their non-rectangular shapes are defined by the two shape masks. By tiling 3 × 3 squares as a transparent

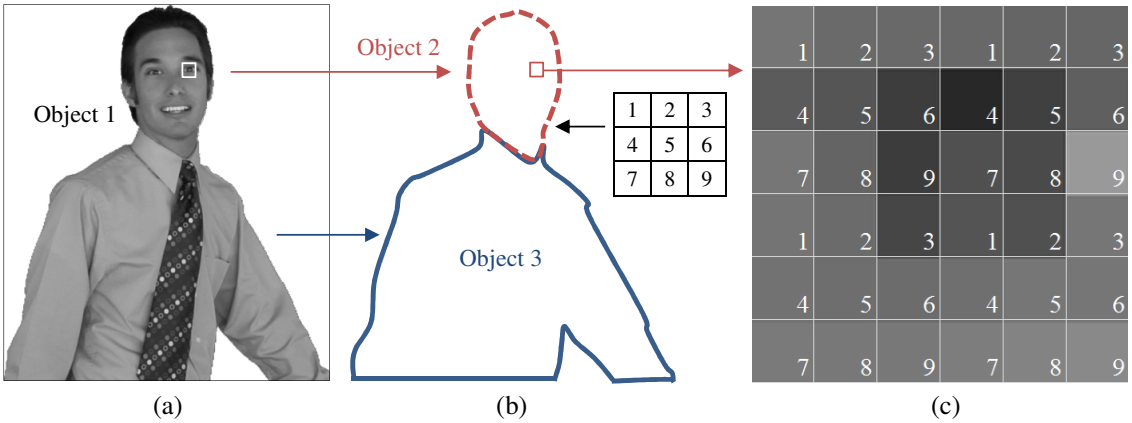


Figure 3.2: An example of an object-oriented scene consisting of three objects (a). Objects 1, 2, and 3 refer to the background, the face, and the rest of the body, respectively. Two masks with arbitrary shapes and 3 × 3 sub-objects are applied to objects 2 and 3 (b). The pixels that contains a part of the eye in the white bounding box are grouped into the nine sub-objects (c).

layer over the luma pixels of the important objects, each of the pixels is associated with and grouped into one of the nine sub-objects. An edge sub-object is always generated, which gives the outer and inner edges of an object.

Grouping the pixels in an object into sub-objects avoids progressive layering and obtains equal-priority entities which are independent until viewed on the screen by the user. In the case of progressive coding, the basic layers are sent as important packets, while higher layers have decreasing priority. At first sight, progressive coding with successive refinement seems to be an advantage regarding scalability, but it is a drawback when the packets have to be dropped randomly. Therefore, a flat coding structure enabled by the proposed concept of sub-object is expected to give a simpler and smoother quality degradation when packets are dropped or lost in the network. Sub-objects are sorted out according to the SP associated with an actual service.

The division into sub-objects also provides unique possibility for variable resolution over the surface of any object, simply by coding each sub-object of an object with variable shape, size, and resolution. Sub-objects are compressed independently using shape adaptation with a new compression scheme called the NOC, which is lossless and has a flat priority structure (Figure 3.3).

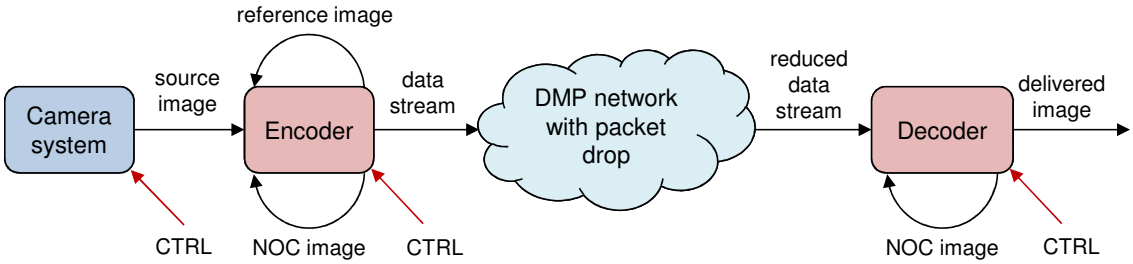


Figure 3.3: An overview of the NOC encoding and decoding.

Inspired by the PNG compression standard [Roelofs (2003)], the NOC is expected to support the extreme quality requirements of the DMP and the 'near-natural' level of perception. The progressive structure in JPEG 2000 can be flattened by dividing an object into sub-objects before encoding. The NOC uses the filtering and deflating compression principles in PNG. In special cases, the filtering scheme of PNG gives high compression ratios without loss [ROE03]. PNG can use up to 64 spectral sub-bands from the visible spectrum, each with a bit depth of up to 24 bits. It includes the PNG RGBA tricolor type and uses a bit depth up to 16 bits per component, and each layer of the multi-layer objects can be treated separately. The filtering has a great potential to be improved further in the future, and the NOC scheme also supports parallel and independent packet decoding.

The contenders to the NOC compression scheme could be lossless JPEG 2000 [Schelkens et al. (2009)], the lossless JPEG-LS, arithmetic compression, or a scheme based on the Burrows-Wheeler transform [Burrows and Wheeler (1994)]. On average, the performance of these schemes is quite close [Santa-Cruz et al. (2000)]. JPEG 2000 has become the cinema standard and performs very well on natural still pictures with compression ratios 10-50, but the PNG is free of patents or fees.

The terms, definitions, and concepts used for NOC correspond to those in the PNG standard with some changes and additions as follows. In NOC, the chunks are covered by the AppTraNetLFC protocol and adaptive QSPs, and an image is one of many representations of a sub-object of arbitrary shape. The image has the same shape as the 2D projection of the sub-object. To obtain a stereoscopic image, two images of the same object are shot with at least two cameras, and more cameras are needed to generate multiview data, depending on the wanted quality. Missing views can be synthesized from the available views and depth maps.

Four types of image are distinguished in NOC (Figure 3.3), and the PNG RGBA tricolor type is considered here as a special case, following the PNG standard. The source image, which is the output of the camera array system, contains the information necessary for the encoder to extract a reference image. The reference image, which only exists inside the encoder, represents an arbitrary shaped area of pixels as a sub-object from the source image which can be recovered from an NOC datastream. The pixels are equal and contain a number of samples from 4 to 65, each is adaptable from 1 to 24 bits. The last sample is an alpha sample, and the others represent the sub-bands of the visible spectrum. Each horizontal row of the pixels is called a scan line. The rectangular shape mask uses the lower left corner as the position address relative to the scene, and the reference pixel coordinate is included in the shape mask data. A bitmap is used to indicate which pixels within a rectangular area belong to the reference image because an object can have 'holes' where objects in the background are visible. An encoder generates an NOC datastream from the NOC image, and the reverse occurs in a decoder.

The compressed datastream consists of independent AppTraNetLFC packets, and each contains data from a certain part of a sub-object. Carried by the correct type of AppTraNetLFC packet, the SPs describe the packing format of the compressed data. Generally, only a part of the sub-object data can be placed in each packet. A packet is filled up with the compressed data (5×256 bytes) from, for example, n spectral sub-bands, one alpha channel, and the shape mask, starting at the last pixel position of the

sub-object in the previous packet. Each packet is self-contained, but each decoder needs the current SP and QSP for decoding the packet correctly. On average, the compression ratio of NOC is about 5:1, which means that 5×5×256 bytes of uncompressed data are stuffed into one packet.

The decoder reconstructs the delivered image from the NOC image using a series of transformations. The missing pixels, due to partly or fully dropping sub-objects, are regenerated by means of linear or non-linear interpolation from the received pixels. By fully dropping sub-objects 3, 4, and 8 from the image in Figure 3.2, the traffic is reduced by 33%.

3.1.3 Networking aspects

The aspects discussed here are the three layers of the DMP architecture, the QS and the transient traffic from DMP scene.

The Physical-Link layer

Given optical fibers between NNs, the Physical-Link layer sends packets out at the maximum packet rate given by the link capacity with PCIe as the recommended protocol since there are many PCIe HW solutions that support the needs of DMP. PCIe, the latest generation I/O bus for data communication, has a static flow-control mechanism that leads to non-dynamic use of credits. It is also designed not for transmission over links with large delay as the sequence numbers or maximum credit are too small; hence, PCIe transactions are used in posted mode, i.e. without flow control (no ACK or NAK). Non-posted transactions, however, can be used for local transmission with very small delays, such as between printed circuit boards, integrated circuits, and boxes interconnected by a few meters of PCIe cables, which are useful for cameras and display processing in a CS. Besides frame- and clock synchronization, the PCIe-transaction frame includes the AppTraNetLFC packet as the payload. Conversion between optical and electrical signals is made at each end of the optical links of 10-100 Gbps Ethernet fiber.

The AppTraNetLFC layer

The only protocol on top of the Physical-Link layer, the AppTraNetLFC layer is a combination of application, transport, network, and partial link-protocol (link flow control) layers. The packet length is 1.5 kB, and the protocol has only one combined packet header, i.e. the AppTraNetLFC protocol header so that parallel HW in ASIC efficiently processes the parameters. Some of them are introduced to increase the performance, support efficient HW design, and reduce network complexity.

The AppTraNetLFC packets in the DMP network consist of AV packets and control packets. Mostly allocated for the visual part, the contents of the AV packets are acquired by the camera and microphone arrays from a transmitting CS. Control packets are used such as for establishing the collaborations and adaptive control during collaborations. Because AV packets are very sensitive to delay, most of them are subject to dropping during transmission. On the contrary, the real-time requirements and the loss probability of control packets depend on the applications and situations; hence, their transport

delays are highly variable, depending on traffic patterns. Thus, five priority classes are defined in Table 3.1 for both packet types. Note that AV packets belong to Class A, but control packets are distributed in all the classes. The loss probability for certain control packets could be very low.

Table 3.1: Five priority classes of AppTraNetLFC packets

Class	Real-time requirement	Delivery requirement
A	Guaranteed	Very low loss probability
B	Moderate	Correct sequence
C	None	Correct sequence
D	None	Any sequence
E	None	Possible loss

DMP applies IPv6 [IETF (1999)] and IPsec [IETF (2005a)] with slight adaptation. The IPv6 header is used according to the standard, except the IPv6 addresses which are used to uniquely identify users. Integrity, authentication, and encrypted payloads are provided by IPsec's Authentication Header (AH) and Encapsulating Security Payload (ESP) in the Transport Mode. Moreover, the Internet Security Association and Key Management Protocol (ISAKMP) [IETF (1998)] and Internet Key Exchange (IKE) [IETF (2005b)] are used for key exchange. Although one drawback could be that the AN and its servers have to be trusted, the service provider in DMP is generally trusted because it is responsible for the services, quality, servers, and network. The protocol parameters in the AppTraNetLFC packet header are presented in Table 3.2.

By using IPSec, the packets are self-sufficient regarding security. Moreover, the use of IPv6 and IPsec makes it possible to transmit AppTraNetLFC packets through the existing Internet. The ability of TCP to support reliable end-to-end transfer is not required by DMP architecture because it is guaranteed by the AppTraNetLFC protocol. UDP is not used because it does not provide functions useful to DMP. Since the AppTraNetLFC protocol also supports the necessary functionality for the application layer, other protocols such as H.323, RTP, UDP, and SIP are also not used in DMP.

The Application layer

The Application layer consists of six main functional blocks. The SP, (SC) and Quality Control (QC) blocks are common functions in ANs, but DMP NNs only have QC. The blocks of SOC (Sub-object Codec), DMVC (Display and Viewer Movement Control), and SSOMC (Shooting, Segmentation and Object Movement Control) are implemented only in the user equipment, i.e. the CS.

SCs are the 'beacons' of the DMP system with top responsibility for setting up, managing, and releasing collaborations between users, on request from a user. Besides cooperating with the QC in the user terminals, it also manages and routes packet exchange between users and various servers allocated to ANs and between access- and destination NNs. The QC handles the QS of the scenes through controlled dropping of the AV packets, and the SOC is responsible for encoding and decoding sub-objects.

Table 3.2: The AppTraNetLFC protocol header (lengths in bits)

Parameter	Length	Description
PT	8	Payload type
PacketRate	16	The current packet rate from the sub-object, indicated by the user
Sub-objectDrop	2×8	For controlled dropping of progressive JPEG 2000 compression layers sub-objects compressed by the NOC scheme
Timestamp	32	For delay measurements from an NN to an AN, and from users to ANs
SurfaceAdr	16	Addresses the surface of a CS
ModuleAdr	16	Addresses the module of a surface
View	16	Addresses the view of an object
PixelAdrB	32	Addresses the start pixel (x, y) of the sub-object represented in this packet (rectangle)
PixelAdrE	32	Addresses the end pixel (x, y) of the sub-object represented in this packet (rectangle)
Layer	16	The layer number of objects (background lowest number 1)
SPQSP	32	Reference to SP and QSP, used for collaborations
Reserve count	8	Counts the number of Reserve bytes
Reserve bytes	0–64 bytes	For future use

The SSOMC engages with multiview tracking and shooting of objects moving relatively to a background and other objects in a scene. The DVMV identifies the viewers with their current position and movements in the room, tracks each viewer's eye gazes, and selects which object in the scene that each viewer focuses on at any time. The SP determines the objects that are likely to be focused on together with their importance and characteristics.

Quality Shaping

In the simplified overview of the QS in ANs and NNs (Figure 3.4), the maximum EED is guaranteed with a number of specialized servers, e.g. to support collaboration establishment and management, by means of the QS in a DMP network. In both node types, control packets enter queue Q2 which holds packets on a very large store, and they enter Q1 for link output if SEL selects them with probability $P1$. The maximum length of Q2 must be so large that reaching the maximum should have an extremely low probability before the admission control decreases the input traffic to the network. The module H is an abstraction of the output Physical-Link layer, and the measured delays

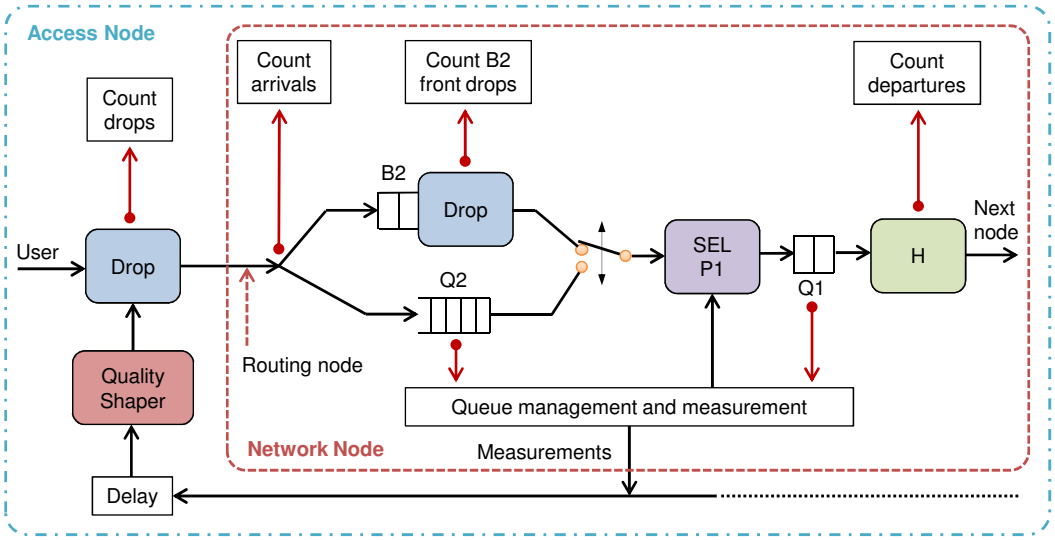


Figure 3.4: A simplified Quality Shaping in a DMP NN and AN.

determine the admission control.

When AV packets must be dropped, they are sent to buffer B2, and the selector SEL fetches them from B2 with probability $1 - P1$ and forwards them to Q1. Q1 has a limited length and determines the maximum jitter of a transfer through the node. If Q1 is full, packets start queuing in B2, a very short buffer used for dropping according to the packet sequence-number which is directly based on the importance of the contents for object-oriented image/video coding. Optimizing the maximum lengths of B2 and Q1, the drop-decision lengths of B2, and the value of $P1$ are some open challenges in designing a QS scheme, which are beyond the scope of this work.

DMP networks are designed to use fixed routes as the basic rule to guarantee delays. Nevertheless, load sharing of links between nodes and hot standby of the processing section for each link are used to provide alternatives when a processing part or a link goes down. Alternate routes via different nodes give the same number of hops, and maximum delays may vary only because of different propagation times. The traffic is equally shared between alternate routes, which is a must for the QC because each node reports its traffic load independently.

The network part of the QS scheme includes the Queue Management and Measurement (QMM) entity in all nodes and the queuing system with the Quality Shaper in DMP ANs (Figure 3.4). The latter is the main difference between ANs and NNs, and the measurements are performed typically every 5 ms, a small period compared to maximum round-trip delay which is > 60 ms. It is also in the same range of the limit for human perception of fast changes. Nevertheless, users can increase the measurement frequency, for example by measuring every 10μs. In every measurement, the QMM at a DMP NN first counts and records the number of arrivals, dropped packets, and departures from all DMP ANs that load the NN by retrieving their addresses from the packet headers. After logging the queue length contributed by these ANs to Q1 and Q2, and records the current time, the time since the previous QS packet is sent, and the current link capacity.

After a number of intervals given by a variable, which is typically 5 ms for normal situation and 1 ms when packet drop starts, the QMM is activated to read the counters and forward QS packets from its node to the ANs that loads it. Based on the measurements in the NNs over a long time that are reported back to the ANs, the load on the nodes, for say a few hundred ms, ahead in a given path can be predicted. When a QS packet arrives at the source AN, the QC analyzes the measurements and decides to either scale the packet rates from users by start dropping packets or not.

This information will be used in the negotiation of SPs and QSPs which occurs when a multiparty DMP collaboration is being established. If one or more nodes in the path are heavily loaded, the request for collaboration setup may be rejected. This is the basis for admission control in DMP networks, so that a minimum scene quality can be guaranteed for the ongoing collaboration.

Packet delays through nodes and processing delays in a CS can be guaranteed to be lower than a specified value. There is no buffering or waiting, except for the output link queues in the nodes. Sufficient information is included in each AV packet to make it independent of the other packets. The content of a packet is used to present parts of an object immediately at the right place and with the right quality without waiting for other parts of the object or other objects. Used without delay in the rendering process, pre-stored (negotiated) configuring data such as the SPs can result from negotiations when a complex multi-party scene is established, or when some configuring actions occur during the collaboration. In short, thanks to the controlled dropping of packets in the network, objects can be automatically synchronized from the source and presented within a guaranteed time with a controlled variable quality.

Maximum EED can be guaranteed because a DMP network has a fixed route with a series of DMP nodes. If the processing times in the user equipment and the nodes are constant, the EED of a packet is the sum of propagation times in the path, the constant processing times, and the waiting times in Q1 in the path nodes. The waiting time in B2 can be neglected compared to the waiting time in Q1.

Given that there is only one path through the network, and the path via B2 is only for AV packets, a QS system can be modeled as a feedback control system. Older control models using discrete PID, P or PI regulators, or newer models such as the 'receding horizon' or 'moving horizon' models from predictive control can be applied. A simulation model in DEMOS [Birtwistle (2003)] based on PI regulator is presented in [Rønningen (2005)] together with other extensive simulations to test and validate DMP.

Transient traffic from DMP scenes

Since the AV traffic sources of DMP appear to be non-linear and transient, established stochastic process theory assuming stationary processes cannot be applied. Instead of on mean values, the interest is on transient slopes and their variations as well as extreme values and their durations. In some cases, the sources are independent of each other, but they can have strong dependencies in their packet streams, such as in an artistic collaboration. By assuming stationary processes, aggregated streams of control packets may be modeled using established queuing theory [Iversen (2007)]. Following each AV packet through the network is of less interest, since the path through the network gives the minimum and maximum user-to-user delay (jitter). Assuming a maximum jitter of

10ms, the variation is not perceivable on the display; hence, the distribution of drop occurrences vs. time interval vs. drop rate is more interesting.

The packet rate, as seen from the sources, typically can be constant for a long time, and then instantly steps to another constant value. For a packet stream from a sub-object, the time between each rate step behaves randomly, but the distribution is presently unknown. Manual inspection of some cases suggests that uniform probability-density functions can be used as an approximation. Thus, if many independent streams are merged, the negative exponential distribution reasonably approximates the (unknown) distribution of time between changes of packet rate.

Typical for the sources, the extreme variability of packet generation rates can be regarded constant in time intervals that are approximately uniformly distributed. The packet rate can vary from a few kilo packets per second (kpps) to several Mega packets per second (Mpps). A simple random rate slope can be modeled by drawing four points and three line segments in the time-rate plane. The start and end points of each line segment are drawn from 2D uniform distributions. By generating several objects, several slopes can be concatenated to form any slope as a continuous curve of line-segments. Constant rates in short time-steps are used as an approximation to the (random) lines, and the number of time-steps is an input to the mode. The random and transient lines are obtained by using samples from four uniformly distributed squares (input). The merged stream packet rate from 50×4 step-rate generators in Figure 3.5 shows transient slopes that vary from about 60 to 260 Mpps.

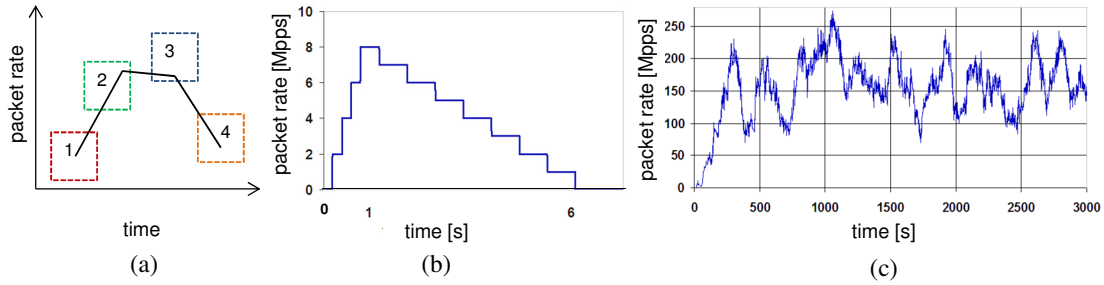


Figure 3.5: A random four-line slope for a step-rate generator (a); a predicted packet rate from an object that comes into a scene from one side in 1 second and disappears from the other side in 6 seconds (b); the packet rate merged from 50×4 step-rate generators as in (a) (c).

3.1.4 Relationship to the state of the art

First proposed by Rønningen (1982), the traffic shaping as the basis of the QS concept evaluates variance reduction of traffic streams by using packet drop and feedback control. It was further developed by the 'ATM world' that led to the proposal of the Leaky Bucket [Turner (2002)] and Token Bucket flow-control schemes. Because the TCP protocol (over IP) does not support real-time transfer of AV contents, the protocol stack RTP/UDP/IP is widely used to support AV packet networks, but they neither support traffic control nor guarantee quality of service (QoS). On the contrary, UDP traffic streams fill a channel according to the 'best effort' principle and suppress TCP traffic [Lie et al. (2005)].

A large number of traffic-control algorithms have been proposed and, as examples, only some of them are mentioned here. Note that all these proposals add complexity to TCP/UDP protocols and do not support adaptive control of user scenes and traffic as accommodated by DMP. The following coverage is not exhaustive because such traffic issues are outside the scope of this PhD thesis.

Alternate Best Effort is a queuing scheduling proposal that trades loss for latency in giving fairness to links carrying both TCP and UDP traffic [Hurley et al. (2001)]. Explicit Congestion Notation (ECN) packets are marked instead of dropped, and the destination node signals the marking back to the source which acts accordingly [Floyd and Jacobson (1997); Ramakrishnan et al. (2001)]. Additive-Increase Multiplicative-Decrease (AIMD) is a control method used in traditional TCP [Dumas et al. (2002)]. TCP Friendly Rate Control (TFRC) gives TCP friendliness in the long run, but the variance of data rate is less than that for AIMD [Handley et al. (2003)]. Data Congestion Control Protocol, which is connection-oriented and runs on top of UDP, can switch between AIMD and TFRC [IETF (2007)]. Random Early Detection incoming packets are randomly dropped with a probability based on a cost function of average queue length [Floyd and Jacobson (1997)]. Random Exponential Marking uses the quantity known as 'price' to measure the congestion in a network [Athuraliya et al. (2001)]. XCP is a transport protocol which generalizes the ECN, decouples utilization control from fairness control, and avoids instability problems [Katabi et al. (2002)]. Active Queue Management (AQM) regulators are applied in routers and terminals in packet networks with mixed TCP and RTP/UDP traffic to obtain fairness [Lie et al. (2004a,b); Kim and Low (2003); Hollot et al. (2001)]. P-AQM outperforms the aforementioned schemes and provides high link utilization and fairness between TCP and UDP traffic [Lie (2007, 2008)].

3.2 Relationship to the Research Questions

Existing work related to RQ-1 focuses on the design and implementation of the micro-cameras and display panels for the envisioned CS [Rønningen (2012, 2011a)]. Since the solution to the problem formulated in RQ-1 is still a void, the contributions to RQ-1 are novel to the work on DMP.

The transient traffic synthesized using the step-rate generator above has some weaknesses. It comes only from one view angle in a CS because how the collaboration scenario and the eye gazes of the collaborators affect the traffic cannot be incorporated. Moreover, the proposed model of the four-point slope so far represents scenes that can be modeled as camera panning movement. Other types of scenes and the corresponding transient traffic produced have not been investigated for modeling. These drive the need to address RQ-2 to advance future DMP simulations.

In terms of RQ-3, there is no work on closer approximation to the actual object-oriented compression. How the missing pixels are interpolated to generate the theoretically best visual quality is still an open question. The use of JPEG 2000 is prohibitive for the envisioned collaboration system due to its high computational complexity.

CHAPTER 4

Contributions and future outlook

This chapter summarizes and describes the contributions from the included papers in Part II in lay terms as much as possible. The goal is that, after reading this chapter, one can obtain a comprehensive and essential picture of the contributions from the PhD work without having to go through the included papers. There are four types of contributions in this thesis: the proposed and implemented ideas, the work, the results from the work, and the ideas for future research. The latter is included because usually only when the previously uncharted territory is explored can they appear. No previous related work is cited here because they are discussed in Chapter 2 to present the unique position of the thesis.

Figure 4.1 shows the relationship between the papers, the three RQs, and the four aspects in the subtitle of the thesis. The papers can be seen as divided into two groups with respect to the subtitle. The first group, which comprises Papers A and B, covers the modeling, simulation, and synthesis aspects of the envisioned collaboration system (RQ-1 and RQ-2). Paper A covers the three aspects where the transient traffic is obtained by means of DES, and in Paper B, the transient traffic is studied via measurement.

The second group on the compression aspect (RQ-3) consists of the other four papers, i.e. Papers C to F, which can be divided further into two approaches. Papers C and D are on optimal interpolation on pixel domain where packet dropping is applied to losslessly compressed video data. Packet dropping of compressed data using discrete cosine transform (DCT) and resampling is presented in Papers E and F.

The contributions to each RQ are visually summarized to help the reader view them as a coherent flow. Table 4.1 introduces three identifiers that point out the type of contributions in the visual summary. The asterisk means that the contribution is produced from the work in which the PhD candidate is not the major or sole contributor. Besides the identifiers, the red arrows and triangles also guide the flow of the contributions from the beginning to the end. Ideas for future research for all the RQs are discussed in the last section, and the other types of contribution for each RQ is elaborated in their own section. The number in the last two identifiers is incremented from RQ-1 to RQ-3 to show the total numbers at the end.

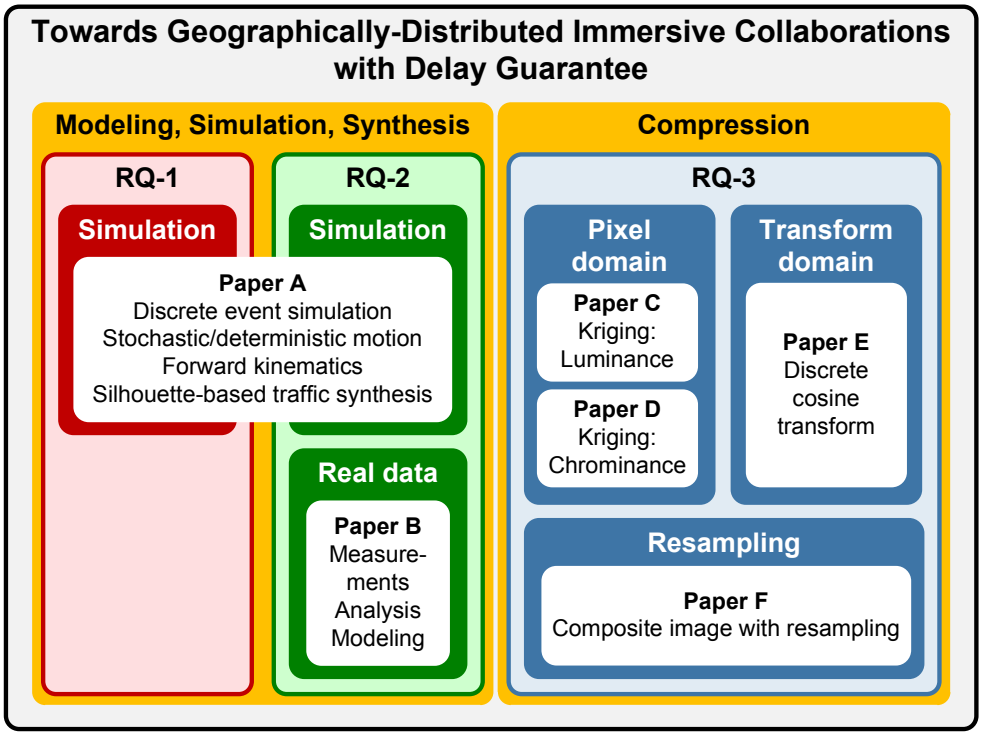


Figure 4.1: Relationship between the three RQs, the included papers, the four aspects in the thesis subtitle, and the title of the thesis with keywords.

Table 4.1: The three identifiers with the denoted types of contribution in the visual summary.

Identifier	Types of contributions	Example
A letter and a number in red box	A step of work	A1
A number in a blue box	A conceived and implemented idea, or a result from a step of work	12 13*
A number in a green circle	An idea for future work	②

4.1 Contributions to Research Question 1

Figure 4.2 summarizes the contributions to RQ-1, and the details can be found in Paper A. They consist of modeling a human body and simulating the forward kinematics of its stochastic and deterministic motions as an independent entity for DES of a real-time collaboration with arbitrary scenario. The envisioned CS is modeled as a cubic environment in which an arbitrary number of people perform a set of activities that consist of human motions A1. To model and simulate the envisioned collaboration with an arbitrary scenario from an arbitrary number of interconnected CSs, the motion of a human body needs to modeled and simulated first. Valid models and simulations of the interaction between moving human bodies in a CS are key answers to RQ-1 7.

Simulating human motions requires a model of a normal human body, which consists of sixteen interrelated limbs A2. The model is simplified further as follows. The

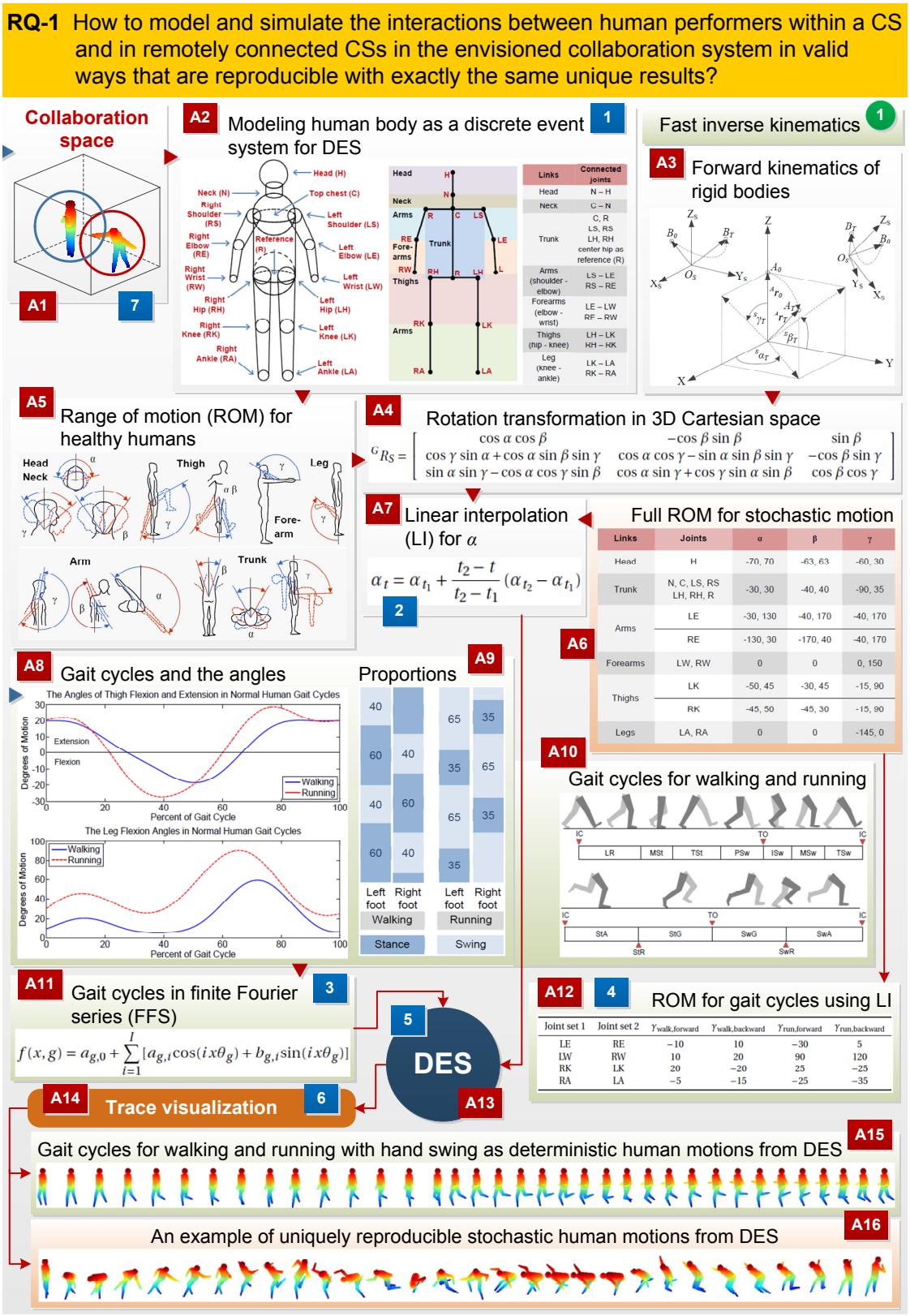


Figure 4.2: Summary of contributions to RQ-1.

model excludes the limbs with relatively insignificant sizes, such as the feet and the fingers, to make the simulation simpler and faster. A limb is also modeled as a line with two end points. The parent-child interconnections between the limbs are shown in A2, e.g. the right arm is the parent of the forearm. Because only the motion matters here, the realistic appearance of the limbs, e.g. using computer graphics, becomes irrelevant.

Moreover, the chain connection between the end points of a limb implies that the displacement of one end point solely determines the motion of the limb. Hence, the motions of the sixteen limbs can be simulated independently and concurrently. It means that the model can be viewed as a discrete event system where an event refers to the displacement of a moving end-point of a limb to a new position. It makes the proposed model of human body suitable for DES 1.

Forward kinematics governs the motion of a limb since it is viewed as a rigid body in 3D space A3. Simple and suitable to the model of human body, forward kinematics models the motion of a point as a rotational and translational transformation with respect to a reference point as the parent. The rotational transformation in 3D space works a function of the α, β, and γ angles A4, which must be within the range of motion (ROM) for a healthy adult person. The ROMs for the key limbs A5 are key input to the motion simulation which can be customized according to the type of motions involved in an arbitrary collaboration scenario.

Human body-motions are divided into stochastic and deterministic motions according to the pattern. A limb with stochastic motion moves freely within its ROM A6 without any given repetitive pattern. In contrast, a deterministic human motion follows a repetitive pattern naturally or deliberately, e.g. the human gait cycle for walking and running in A8 to A10. A gait cycle can be simulated by faithfully following any function that represents the curves in A8 such as the finite Fourier series (FFS) which gives the smallest mean square error A11. In this work, FFS is used up to the fourth harmonics with the given values of the parameters 3.

For both types of motion, the transition of a point from a position in 3D space to a new one is formulated as a linear interpolation (LI) applied over time to the three angles 2, cf. the LI function for α A7. LI is fast to compute in DES and sufficient for RQ-1 because it does not require realistic emulation of human motion.

The FFS, however, is not very flexible to adapt with an arbitrary cycle which requires a different set of parameter values. An alternative to FFS is to approximate the gait cycles by applying LI to the relevant angles with customized ROMs for the involved limbs. For instance, the ROM in A12 is not only for the gait cycles, but also for the accompanying arm swing 4. An approximated human gait-cycle can be simulated simply by modifying the related ROMs.

The LI and FFS for stochastic and deterministic human motion are incorporated in the resulting simulator which simulates the motions of an arbitrary number of people performing in a CS A13. The object-oriented, fast, and powerful Discrete Event Modeling using Simula (DEMOS) is the chosen DES tool which has been widely used to simulate complex discrete-event systems 5. By implementing each limb and performer as a class in DEMOS, they are easily instantiated and executed concurrently.

The new position of a moving end-point of a limb is computed using rotational and translational transformation according to the parent-child relationship that belongs to

the limb. This yields the final position with respect to the general-reference Cartesian coordinate. Given the frame rate F for intraframe processing, all the final instantaneous positions of the limbs generated every $1/F$ second are saved and appended to the output files. The simulator reads the input simulation parameters such as the ROMs, simulation time, seed number, and F from the input files prior to simulation. As required for DES, a unique seed number produces a unique and reproducible set of positions of the limbs.

As input to the visualization step in 3D space [A13], the output files contain the coordinates of the end-points of the limbs. This step visualizes the resulting moving human bodies in a CS for evaluation and further use. Note that the DES concerns only about the positions of the points [A14] and does not consider the physical shapes of the limbs. The shapes, however, are incorporated in the visualization by modeling them in cylinders for the lines and spheres for the points [A12], e.g. the head is described as a sphere and the neck a cylinder. Each limb has its own shape parameters, and users can specify the values in a special input file to an external visualization program. The shape parameters define the physical shape of a person who performs in a CS, such as the body height and width. Transforming the coordinates of the points from the DES translationally using the parameters yields the final coordinates for visualization [6].

Exemplary sets of deterministic and stochastic motions of a human body in [A15] and [A16] are produced using Matlab and depicted in gradual colors from red to blue for easy inspection by the reader. The deterministic motion includes a cycle of walking and running with hand swing, and the reference point R in [A2] is set to be stationary [A16]. All the steps of work from [A2] to [A16] model and simulate an arbitrary number of people who perform in a CS [A1] with stochastic or deterministic motions [7]. With the vast range of available seed numbers in DEMOS, virtually unlimited configurations of motions can be uniquely generated, and each configuration is also uniquely reproduced by a unique seed number. The proposed simulation framework offers an easy and valid way to investigate the envisioned collaboration system despite its inexistence, and it is applied later to synthesize the transient traffic from a CS.

4.2 Contributions to Research Question 2

The crux of RQ-2 is to synthesize transient traffic from a CS as the input traffic for future simulation of DMP network. Figure 4.3 summarizes the contributions to RQ-2 from Papers A and B. Paper A covers transient traffic synthesis using the visualization of moving human bodies as shown from [B1] to [B6]. On the other hand, Paper B measures the traffic from segmentation of human body from uncompressed video clips of human motion. The results are approximated using mathematical models incorporated in DES.

The image in [B1] of two persons assumed to perform in a CS is seen as a video frame captured by an array of cameras on a surface of the CS. One person occludes the other, and their bodies are segmented from the background.

The hypothesis is that the data rate from the segmented region of the bodies in an image is proportional to the area of its silhouette [8] which is exhibited in [B1]. The hypothesis is empirically validated with measurements in [B2] where the body of the person in each frame of an input video clip is segmented. The frame numbers give an idea of the clip contents, and each frame is saved as very high quality images in

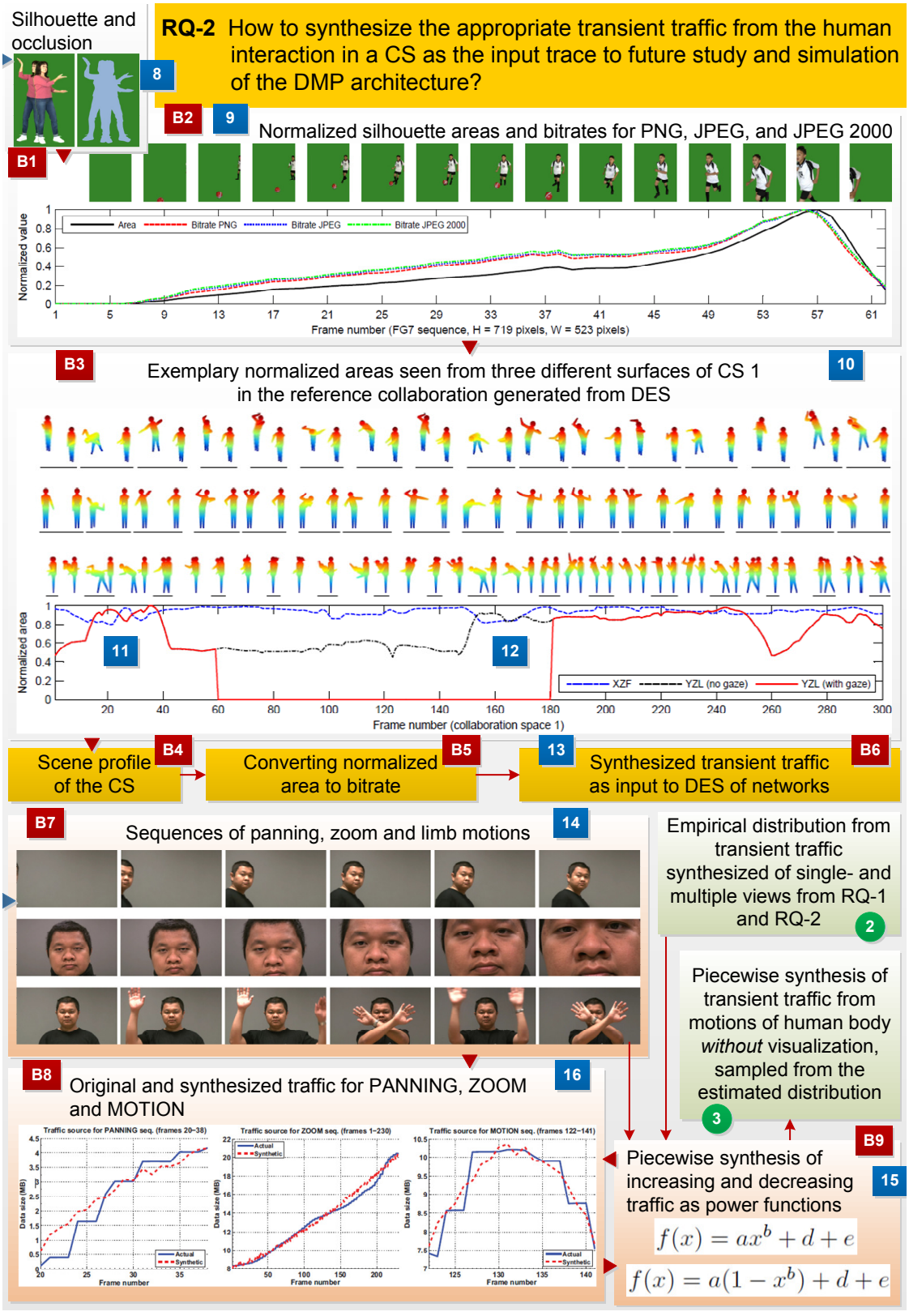


Figure 4.3: Summary of contributions to RQ-2.

PNG, JPEG and JPEG 2000 formats because the DMP network receives (almost) lossless video signals from a CS. The silhouette area of the segmented body in every frame is computed based on the number of pixels therein. The data rates from the three image formats and the areas are then normalized to their respective maximum values so that all the curves can be shown in one plot B2.

The same envelopes of the curves confirm that the hypothesis is valid and correct for the clip. The same phenomenon also applies for all the clips tested 9. The local differences in the envelopes are due to various frequency contents in the segmented region. Nevertheless, the relationship between the normalized data rates and the silhouette areas is linear. Therefore, the expected data rates as the traffic can be synthesized from the normalized silhouette area of a segmented moving human body.

Before explaining the synthesis, some important features of the visualization step in A15 and A16 need to be discussed first. A set of motions from two human performers in a CS are shown from three different views B3 under the scenario for CS1 in the reference collaboration. Azimuth and elevation angles define an arbitrary view that represents which surface and cameras are active in a CS. The motion in the top row is generated from the view where the two angles are 30 and -135 degrees, respectively. Those in the middle and the bottom row are from the frontal and the left-hand views, respectively. By defining how a performer is located relatively to the other in a CS, the visualization simulates the camera motion and the locations of the performers 10.

From the visualization of the two moving human bodies, the silhouette areas of the segmented bodies seen from two different views are normalized and plotted B3, which produces the visual signal 11. Based on the validated hypothesis, such a signal is logically expected from analysis and measurement of a video captured from a real performance. Traffic synthesis from real videos should be avoided, not only because it is time consuming, but also due to the costs for the equipment and hiring professional people to perform. Moreover, the results are impossible to be uniquely and exactly reproducible unlike our proposed DES framework.

From the resulting plots 12, the visual signal from the frontal view is considerably stationary, but it can change when converted into bitrates as the input traffic. In contrast, many transient periods are present in the visual signal from the left-hand view because the two bodies occlude each other. One of them moves one step forward or backward from the original position while the other does not; hence, when there is no occlusion, the amount of data is doubled.

Another cause of transient periods is the eye gaze. When none of the performers in CS2 looks at their right-hand side, i.e. the left-hand view of the performers in CS1, transmitting the visual signal from that view of CS1 to CS2 is unnecessary. The eye gazes are simulated by setting an arbitrary period in the visual signal from that view to zero until any of the performers in CS2 looks back to the right-hand surface of CS2 B3.

Afterwards, the transient traffic in bitrates is generated as the input to future DES of DMP network. The technical specification of each CS, such as the spatial and temporal resolution B4, are defined as scalar values by which the visual signal is multiplied to yield the transient traffic B5. It is saved into files that are used as the input trace to the network simulation 13.

The study of the traffic by using real video clips of various motions spans B7 to

B9. Three recorded clips represent the three most common types of motion: those due to camera panning, camera zoom, and motions of limbs (exemplified by moving hands), as shown in some exemplary frames B7. The person is segmented in all the frames, and the traffic is computed and plotted B8. The plots show that the traffic can be analyzed in piecewise manner because each piece of the plot increases or decreases 16. Two power functions are proposed to describe the two possibilities B9, and they can approximate the measurement results B8. Although fictitious transient traffic in piecewise fashion is easy to implement in DES, such traffic only comes from one view, unlike the multiview traffic using DES and visualization of human motion.

4.3 Contributions to Research Question 3

4.3.1 Pixel domain

Visually summarized in Figure 4.4 from Papers C and D, the contributions to RQ-3 on pixel domain address the dropping of packets containing video data compressed using the NOC proposed in DMP C1. By tiling $N \times N$ blocks on an important object segmented from an image, the pixels in the object can be classified into N^2 sub-objects. The aggregation of the pixels in the same sub-object gives N^2 sub-streams that are losslessly encoded using NOC and dropped partly or entirely. For example, sub-objects 3, 4 and 8 are completely dropped in C2, as indicated by the symbol $\times$. The compression philosophy in DMP is described with the term 'compression by network' to differentiate with those existing in image/video compression.

The problem of reconstructing the segmented object from the remaining pixels at the receiver is formulated in 17. Due to the combination of full and partial dropping of sub-objects, the coordinates of the remaining pixels have regular or irregular pattern. The LENA image, tiled with 3×3 blocks with only one sub-object remaining, is an example of a regular pattern C3.

The search for the solution to the problem leads to kriging, an optimal interpolation technique which has been widely used in geostatistics. Given a number of locations without any regular pattern and known data for each location, such as estimated oil deposit, kriging is instrumental for computing an optimal estimate for an unknown deposit in a point between the locations. The known samples here are the remaining pixels scattered on the image with the intensity values as the known data.

Since the number of samples here is much more than that in geostatistics, applying kriging directly on a whole image is formidable. Instead, a tiled version of kriging called windowed kriging (WK) interpolation is proposed to reduce the complexity by promoting parallel computation 18. A layer of overlapping blocks is overlaid on the image, and the kriging is computed on each block concurrently 19. HW design for the implementation on FPGA C5 has been proposed 19*.

The effects of the WK parameters to image quality and computational time are studied on several standard test images 20 C6. WK consists of modeling and prediction. In general, fewer samples reduce the time for modeling and increase that for prediction. A major drawback of WK, the quality of the interpolated image is surprisingly quite constant despite the number of samples in the received images.

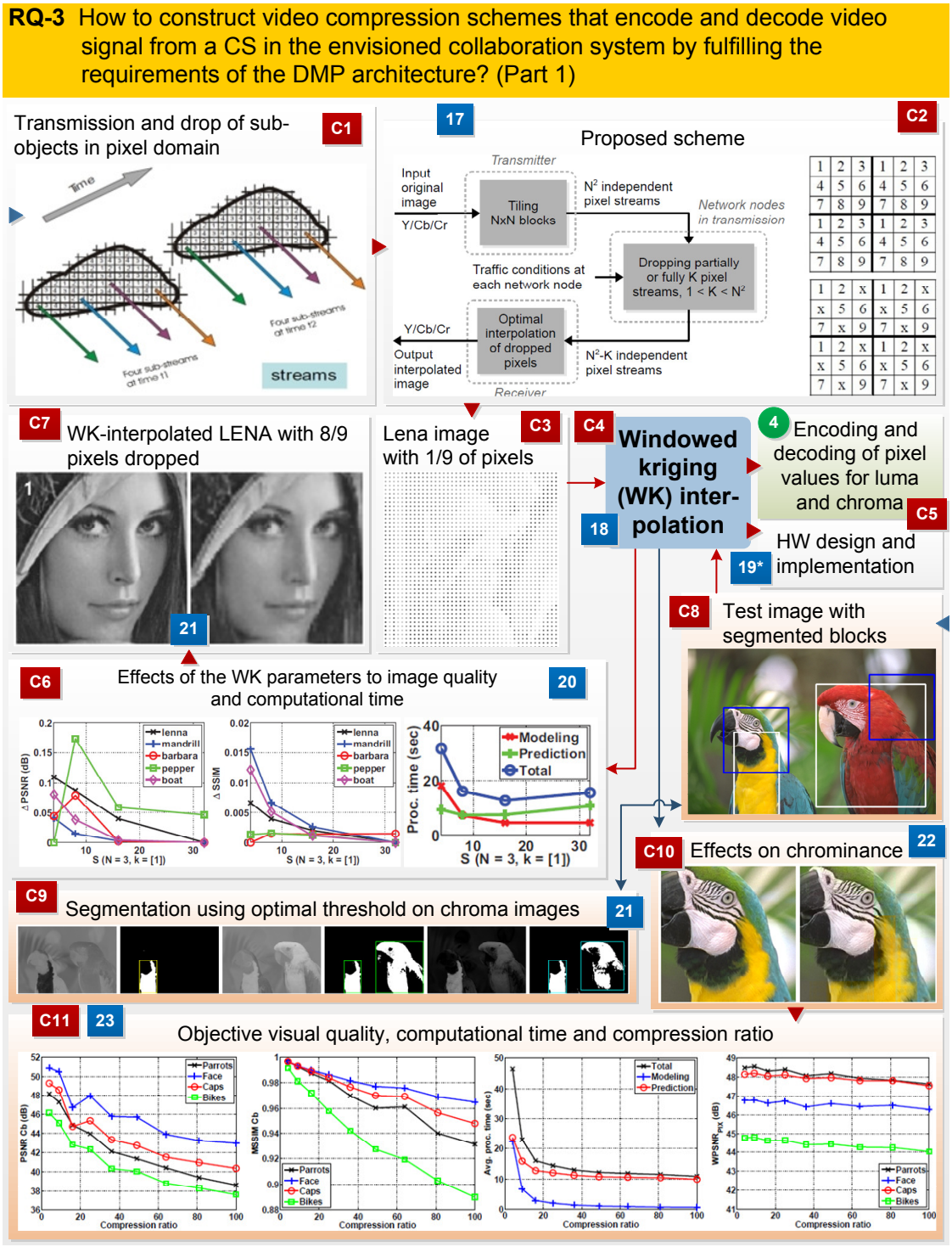


Figure 4.4: Summary of contributions to RQ-3 (pixel domain).

Moreover, edges and textures in natural images are very challenging to interpolate without visible distortion. Boundary artifact may also appear along the edges between two adjacent blocks. A number of overlapping pixels over the border need to be included in computing WK on the two neighboring blocks to avoid the blocking artifact. The

number of pixels in the overlap is one of the WK parameters, and the quality of the interpolated image is still quite decent [21]. The image in [C7] is interpolated using WK from LENA image in [C3].

Unlike the luma, chrominance has less edges and textured areas, making it more suitable for WK interpolation. Using several standard color test images such as PARROT image [C8], colors are shown to have different levels of difficulty to interpolate without noticeable artifact; a phenomenon called color compressibility in this work. Red and yellow in PARROT image are examples of colors that have low compressibility, i.e. interpolating them with small number of samples easily produces noticeable artifact. Segmented using optimal thresholding [C9] [21], the areas filled with these colors are indicated with bounding boxes with white lines, as exemplified in [C8]. The area outside the bounding boxes can be interpolated with much less samples than those within.

The visual quality of the interpolated chroma from one of the two blue bounding boxes in [C8] is shown in [C10]. The yellow color spills outside its area in the right-hand figure when the chroma images are dropped and interpolated at the same compression ratio (CR). This is a typical distortion for chrominance due to applying block-based WK interpolation to non-rectangular regions [22]. CR is defined here as the ratio of the total number of pixels and that of those that remain, and fully retaining only one sub-object gives CR=N^2. The relationship between VQ, computational time, and CR for the test images is also studied [C11] where PSNR and SSIM are the objective metrics, and N = 2, 3, ..., 9, 10. Comparing the plots with the reconstructed images shows that very high CR (up to 100) in the areas with insensitive colors can be achieved without noticeable artifact [23].

4.3.2 Transform domain and resampling

Figure 4.5 visually summarizes the contributions from incorporating an image compression scheme based on discrete cosine transform into the DMP architecture and from using resampling to produce composite images.

The proposed image compression scheme [D1] differs in a number of ways from existing image coding standards [24]. The proposed encoder has no quantization step, but employs block ranking. During transmission, every DMP NN deliberately drops the packets that encapsulate the decoded DCT AC coefficients, which are allowed to be discarded. The dropping occurs in compressed domain to avoid repetitive step of decoding, dropping, and encoding which introduces more delay. Without de-quantization, the receiver reconstructs the image from the incoming reduced bitstream of packets by tiling 8×8 blocks of the decoded DCT coefficients into the correct locations in the output image.

The block ranking analyzes the frequency content in a block and assigns it a rank number based on the quantification of the content. Higher frequency contents, such as in edges and textured area, reflect high importance of the block. The number of ranks is arbitrary and depends on the user and the ranking method used. A simple yet effective block-ranking technique using entropy and variance [D3] [25] can be computed very fast in every block independently. The ranges of the block entropy and the block variance, as shown in [D2] for LENA image, are always constant for all standard test images used.

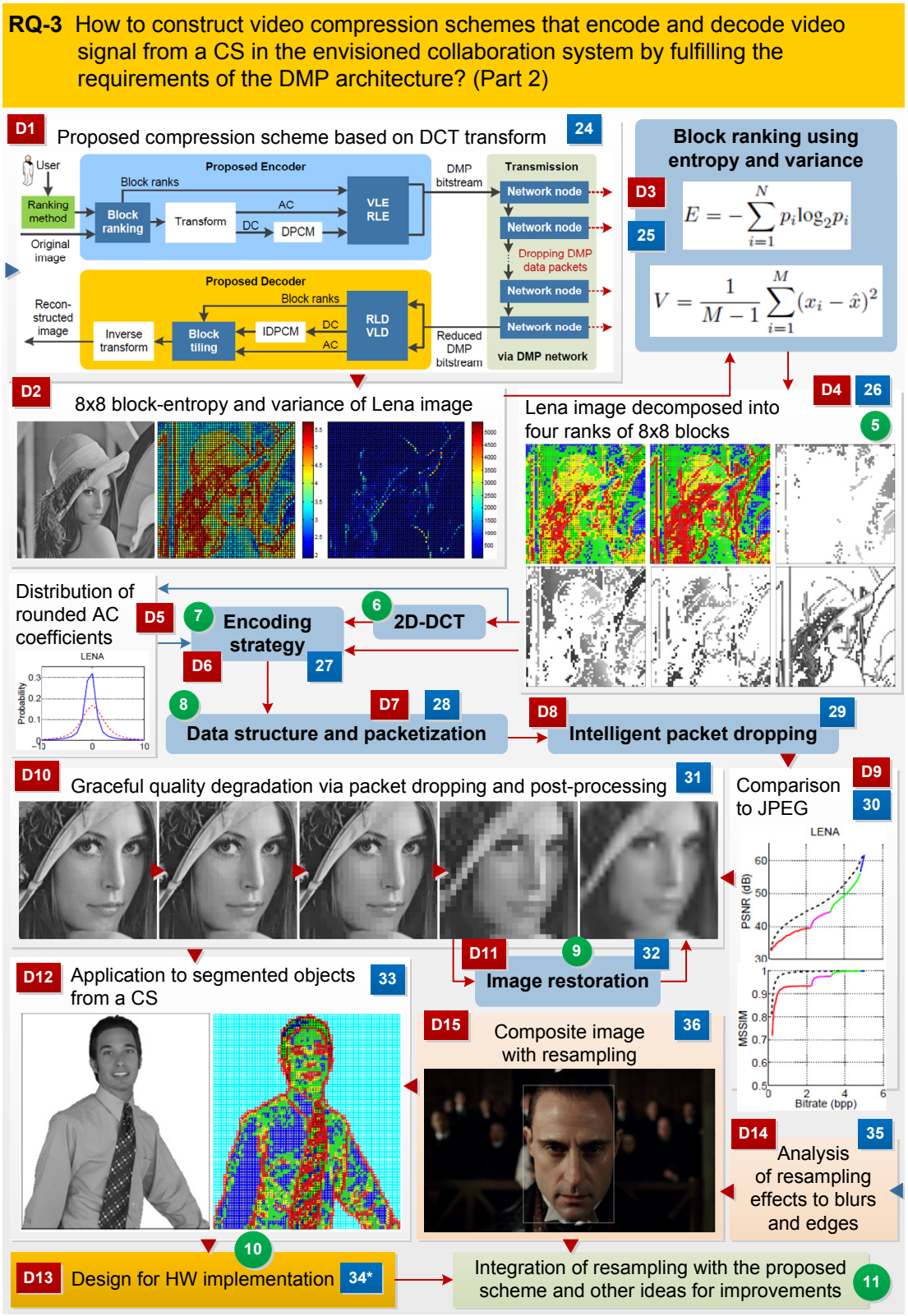


Figure 4.5: Summary of contributions to RQ-3 (transform domain and resampling).

The reader can assess that the color plots from the two measures correspond well to human perception [25].

The proposed block-ranking algorithm defines four block ranks based on the fixed ranges. LENA image is decomposed into the four ranks of blocks shown in blue, green, yellow, and red in ascending order of the importance [D4]. The first figure is the output of the algorithm without using the block variance, and the second one is from using both measures. The difference of the red and yellow blocks in the two figures indicates that block variance is important to make the ranking more consistent to human perception. The remaining four figures show the blocks in the image which belong to each of the four ranks. The results reveal that the proposed ranking categorizes the blocks accurately [26]. The information about the block ranks is encoded and must be available at every NN to enable intelligent packet dropping.

After the block ranking, 2D-DCT is applied in parallel fashion to every block regardless of the block rank to produce one DCT DC coefficient and 63 DCT AC coefficients. Besides the excellent energy compaction, DCT can be computed very fast in HW. The DC coefficient must never be dropped because it contains the average content of the block. The DCT coefficients are rounded to reduce the dynamic range for more efficient computation, which is the only loss of information at the transmitter, and the effect of rounding to the image quality is also negligible.

Encoding the DCT coefficients [D6] should be based on their distribution. Because of their quantity, the rounded AC coefficients are more influential to the distribution than the DC coefficients and the block ranks. The distribution of the rounded AC coefficients of LENA image is shown in [D5]. The distributions and literature study lead to the use of universal codes and Burrows-Wheeler transform (BWT) [27] for encoding. Differential pulse-code modulation (DPCM) produces the difference between two consecutive DC coefficients, and the Fibonacci code is used to encode the differences and the rounded AC coefficients. The Fibonacci code is a universal code that suits well to encode data with distribution as shown in [D5] and can be computed very fast in HW only by using look-up tables (LUTs). The 2D array of the block ranks is encoded using BWT which gives impressive compression with fast computation. The encoded array must never be lost as it is the only means for the receiver to rearrange the received blocks of the DCT coefficients to their correct positions. The data size of the ranks is insignificant, merely around 2 kB for an HD image. The AC and DC coefficients and the block ranks can be encoded independently from each other.

A data structure for packetization is proposed for straightforward packet dropping in compressed domain [D7]. All the red blocks are transmitted first and followed chronologically by the yellow, green, and blue blocks [28]. The AC coefficients of the same index are arranged together by following the zigzag pattern used in JPEG. The blocks are arranged from the most top-left position in the image until that of the most bottom-right. Afterwards the resulting packets enter the DMP network [D8].

Implemented and tested in Matlab, the fully automatic encoding, packet dropping, and decoding [29] produce the relationship between the VQ of the reconstructed images and the bitrates [30] from several standard test images. The curves from the proposed compression scheme and JPEG for LENA image [D9] use PSNR and SSIM as objective VQ metrics. The ranks and their contributions to VQ are indicated by their colors. Evidently,

JPEG performs better than the proposed compression scheme, but the fundamental differences between them make such comparison essentially irrelevant. Note that the curves from the proposed scheme have not taken the image restoration into account as the post-processing step. The comparison aims at showing that the proposed scheme delivers graceful quality degradation, as shown by the continuous curves. Furthermore, the bitrates for the same quality are not very different, which shows that the proposed encoding scheme performs sufficiently well given all the parallelization efforts. On the contrary, non-scalable image coding techniques such as JPEG are in general not compatible with the DMP philosophy.

The graceful VQ degradation for LENA image D10 shows that the blocking distortion is strictly localized in the blocks which AC coefficients are gradually dropped 31; hence, it is called pixelization artifact. This is a major difference compared with blocking artifact in block-based image-coding techniques which often occupies a region of blocks in the image. Since the receiver knows from the remaining DCT coefficients which blocks are affected with artifact, the restoration becomes simpler and faster as searching for the regions with the artifact is not necessary. The fourth image in D10 is the worst reconstruction because all the AC coefficients have been dropped.

At the receiver, post-processing steps include the restoration of the reconstructed image D11. Although constructing a comprehensive algorithm that completely fixes the artifact at any level is outside the scope of this work, a simple, fast, and effective depixelization algorithm is proposed as an example of restoring the worst images 32. The last figure in D10 is the output of the algorithm on LENA image. Notice how the VQ is significantly improved although the edges are not fully restored yet.

The left-hand figure in D12 gives an example of an important object segmented from a video frame that is assumed to be captured in a CS. The next figure is the map of the block ranks where an additional rank is used to indicate the empty white area. Since no DCT coefficient is included in the white background, the blocks of the additional rank add much less than 1 kB to the total size of the encoded ranks. Therefore, the proposed scheme supports object-based processing with arbitrary shapes 33.

The complexity of the algorithms in the proposed scheme is analyzed, and HW designs for fast implementation on FPGA are proposed D13 34*. After the resource consumption for the implementation is estimated, it reveals that the proposed encoder and decoder can process an HD color image in less than 10 μs by maximizing the use of resources of a recent FPGA board. Thus, the proposed image-compression scheme for the envisioned collaboration system is technically feasible.

The effect of resampling HD color images to blurs and edges is also investigated D14. Larger downsampling factor (DF) causes more blur effect and particularly more intense ringing artifact to edges. Because the periphery of human eye gaze is more blurry than the center area, downsampling the areas in an image which are not gazed is proposed as an additional pre-processing step to the proposed compression scheme for more data reduction D35. Although some information is lost due to resampling, the distortion to human eyes is unnoticeable, as shown in the so-called composite image in D15. The area outside the bounding box at the center of the image is downsampled at DF = 4, which means that the size after downsampling is 16 times smaller than the original. The area is upsampled again at the inverse of the DF at the receiver. Human

faces usually attract the most attention of human eyes, which make the distortion in the background much less noticeable [36]. Resampling can be seen as integrating perceptual coding into the compression scheme for DMP.

4.4 Future outlook

Some of the ideas for future research are identified in the visual summaries. The fast inverse kinematics for DES of human motion is worth investigating ① for RQ-1. In this way, the coordinate for the next motion is generated by using a random number generator, and the path from that point to the current coordinate is then estimated in ways that produce a more realistic human motion. It is, however, more complex and computationally demanding than forward kinematics.

For RQ-2, it is interesting to study the empirical distributions from transient traffic synthesized of single and multiple views for RQ-1 ②. The distributions can be used later as the sampling basis for generating transient traffic from human-body motions without visualization in piecewise fashion ②. Although much input traffic can be produced in less time this way, the modeling of dependencies between the traffic from different views is challenging. Unless this problem can be solved, this synthesis method is only for generating traffic from one view.

As for the pixel domain in addressing RQ-3, a scheme to encode the pixel values from WK for luma and particularly chroma besides the NOC would be an interesting endeavour ④. By exploiting the existing redundancies in the pixel domain, it will produce RD curves to reveal the compression performance from WK.

For the second approach to RQ-3, it is preferred that the proposed scheme can work with arbitrary block sizes such as 4×4 to accommodate various types of content in an image ⑤, although the parameter values for block ranking might change. Other transforms, such as the wavelet transform which is at the heart of JPEG 2000, is interesting to study ⑥. The encoding strategy for DCT coefficients can be improved by employing better coding schemes with less number of bits than those used by the schemes selected in this work ⑦. This goes together with improved data structure and methods to construct the packets, e.g. to achieve better error resilience ⑧.

Fully restoring pixelization artifact at all levels of intensity is still an open problem ⑨. Furthermore, the proposed HD design of the coding scheme can be implemented in platforms with parallel architectures such as FPGA, GPU, and ASIC ⑩. Integrating resampling into the proposed scheme is interesting to study by including not only the compression performance, but also the subjective VQ assessment ⑪.

References

3DPresence, 2008. http://www.3dpresence.eu/.

Abbasi, A., Baroudi, U., 2012. Immersive environment: An emerging future of telecommunications. *IEEE Multimedia* 19 (1), 80–86.

Allen, T., 2011. *Introduction to Discrete Event Simulation and Agent-based Modeling*. Springer.

Alregib, G., 2009. Immersive communications: Why now? *IEEE COMSOC MMTC E-Letter* 4 (3), 9–10.

Altunbasak, Y., Apostolopoulos, J., Chou, P., Juang, B., 2011. Realizing the vision of immersive communication. *IEEE Signal Processing Magazine* 28 (1), 18–19.

Armstrong, M., Flynn, D., Hammond, M., Jolly, S., Salmon, R., 2008. *High Frame-Rate Television*. BBC Research White Paper WHP 169, BBC.

Athuraliya, S., Li, V., Low, S., Yin, Q., 2001. REM: Active Queue Management. *IEEE Network* 15 (3), 48–53.

Bailenson, J. N., Blascovich, J., Beall, A. C., Noveck, B., 2006. Courtroom applications of virtual environments, immersive virtual environments, and collaborative virtual environments. *Law & Policy* 28 (2), 249–270.

Baker, H. H., Bhatti, N., Tanguay, D., Sobel, I., Gelb, D., Goss, M. E., Culbertson, W. B., Malzbender, T., 2005. Understanding performance in Coliseum, an immersive videoconferencing system. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)* 1 (2), 190–210.

Bako, C., 2004. JPEG 2000 image compression. *Analog Dialogue* 38, 1–5.

Bandoh, Y., Takamura, S., Jozawa, H., Yashima, Y., 2010. *High-Quality Visual Experience – Creation, Processing and Interactivity of High-Resolution and High-Dimensional*

Video Signals. Springer, Ch. *Mathematical Modeling for High Frame-Rate Video Signal*, pp. 197–215.

Bartholomew, R., 2008. Globally distributed software development using an immersive virtual environment. In: *Proc. IEEE International Conference on Electro/Information Technology (EIT)*. pp. 355–360.

Birtwistle, G., 2003. *DEMOS - A System for Discrete Event Modelling on Simula*. School of Computer Science, University of Sheffield.

Bourke, P., Felinto, D., 2010. Blender and immersive gaming in a hemispherical dome. In: *Proc. International Conference on Computer Games, Multimedia & Allied Technology*. pp. 280–284.

Brock, N., Daniels, M., Morris, S., Otto, P., 2011. A collaborative computing model for audio post-production. *Future Generation Computer Systems* 27 (7), 935–943.

Burrows, M., Wheeler, D., 1994. *A block sorting lossless data compression algorithm*. Tech. Rep. 124, Digital Equipment Corporation.

Buso, N., Allochio, C., Fall 2012. *LOLA – LOw LAtency audio visual streaming system*. http://www.internet2.edu/presentations/fall12/20121002-ALLOCCHIO-LOLA-Plenary.pdf, internet2 member meeting.

Callet, P. L., Möller, S., Perkis, A., March 2013. *Qualinet White Paper on Definitions of Quality of Experience (Version 1.2)*. European Network on Quality of Experience in Multimedia Systems and Services (COST Action IC 1003).

Chafe, C., Gurevich, M., Leslie, G., Tyan, S., 2004. Effect of time delay on ensemble accuracy. In: *Proc. International Symposium on Musical Acoustics*.

Conca, D. P., 2012. *Telepresence quality*. Master's thesis, ITEM, NTNU.

Cruz-Neira, C., Sandin, D., DeFanti, T., Kenyon, R., Hart, J., 1992. The CAVE: audio visual experience automatic virtual environment. *Communications of the ACM* 35 (6), 64–72.

DeFanti, T., Acevedo, D., Ainsworth, R., Brown, M., Cutchin, S., Dawe, G., Doerr, K., Johnson, A., Knox, C., Kooima, R., Kuester, F., Leigh, J., Long, L., Otto, P., Petrovic, V., Ponto, K., Prudhomme, A., Rao, R., Renambot, L., Sandin, D., Schulze, J., Smarr, L., Srinivasan, M., Weber, P., Wickham, G., 2011. The future of the CAVE. *Central European Journal of Engineering* 1 (1), 16–37.

Delaney, D., Ward, T., McLoone, S., 2006. On consistency and network latency in distributed interactive applications: a survey–part I. *Presence: Teleoperators and Virtual Environments* 15 (4), 218–234.

Delp, E., Mitchell, O., 1979. Image compression using block truncation coding. *IEEE Transactions on Communications* 27 (9), 1335–1342.

Domoxoudis, S., Kouremenos, S., Loumos, V., Drigas, A., 2013. Characteristics of video traffic from videoconference applications: From H.261 to H.264. *Journal of Computer Networks and Communications* 2013 (Article ID 614157).

Dumas, V., Guillemin, F., Robert, P., 2002. A Markovian analysis of Additive-Increase Multiplicative-Decrease (AIMD) algorithms. *Advances in Applied Probability* 34 (1), 85–111.

Ebara, Y., Kukimoto, N., Leigh, J., Koyamada, K., 2007. Tele-immersive collaboration using high-resolution video in tiled displays environment. In: *Proc. International Conference on Advanced Information Networking and Applications Workshops (AINAW)*. pp. 953–958.

Eisert, P., 2003. Immersive 3-D video conferencing: challenges, concepts, and implementations. In: *Proc. SPIE Visual Communications and Image Processing (VCIP)*. pp. 69–79.

ETH Zurich, 2003. blue-c. http://blue-c.ethz.ch/.

European Commission 2020, 2010. *Europe 2020: A European strategy for smart, sustainable and inclusive growth*. http://ec.europa.eu/eu2020/index_en.htm.

EVL UIC, 2012a. CAVE2 Hybrid Reality Environment Trailer 1. http://www.youtube.com/watch?v=d5XDbzy7vuE.

EVL UIC, 2012b. CAVE2 Hybrid Reality Environment Trailer 2. http://www.youtube.com/watch?v=vK74PP4kHHM.

EVL UIC, 2012c. Sage. http://www.sagecommons.org/.

Feldmann, I., Schreer, O., Kauff, P., Schafer, R., Zuo, F., Belt, H., , Escoda, O., 2009. Immersive multi-user 3D video communication. In: *Proc. International Broadcast Conference*.

Floyd, S., Jacobson, V., 1997. Random Early Detection gateways for congestion avoidance. *IEEE/ACM Transactions on Networking* 1 (4), 397–413.

Fuo, C., Wang, M., 2012. Motion generation and virtual simulation in a digital environment. *International Journal of Production Research* 50 (22), 6519–6529.

Gross, M., Würmlin, S., Naef, M., Lamboray, E., Spagno, C., Kunz, A., Koller-Meier, E., Svoboda, T., Van Gool, L., Lang, S., Strehlke, K., Moere, A. V., Staadt, O., 2003. blue-c: a spatially immersive display and 3D video portal for telepresence. *ACM Transactions on Graphics (TOG)* 22 (3), 819–827.

Halak, J., Krsek, M., Ubik, S., Zejdl, P., Nevrela, F., 2011. Real-time long-distance transfer of uncompressed 4K video for remote collaboration. *Future Generation Computer Systems* 27 (7), 886–892.

Handley, M., Floyd, S., Pahdye, J., Widmer, J., 2003. RFC 3448: TCP Friendly Rate Control (TFRC): Protocol Specification. IETF.

Harders, M., Spaelter, U., Tuchschmid, S., Szekely, G., 2006. Highly-realistic, immersive training environment for hysteroscopy. In: *Proc. Medicine Meets Virtual Reality (MMVR)*. pp. 176–181.

Hollerbach, J., 2007. *Haptic Rendering: Foundations, Algorithms, and Applications*. A.K. Peters, Ch. *Locomotion Interfaces and Rendering*, pp. 83–91.

Hollot, C., Misra, V., Towsley, D., Gong, W.-B., 2001. On designing improved controllers for AQM routers supporting TCP flows. In: *Proc. IEEE INFOCOM (Volume 3)*. pp. 1726–1734.

Holub, P., Matela, J., Pulec, M., Srom, M., 2012. Ultragrid: low-latency high-quality video transmissions on commodity hardware. In: *Proc. ACM International Conference on Multimedia*. pp. 1457–1460.

Holub, P., Matyska, L., Liska, M., Hejtmanek, L., Denemark, J., Rebok, T., Hutanu, A., Paruchuri, R., Radil, J., Hladka, E., 2006. High-definition multimedia for multiparty low-latency interactive communication. *Future Generation Computer Systems* 22 (8), 856–861.

Holub, P., Srom, M., Pulec, M., Matela, J., Jirman, M., 2013. GPU-accelerated DXT and JPEG compression schemes for low-latency network transmissions of HD, 2K, and 4K video. *Future Generation Computer Systems* 29 (8), 1991–2006.

Hua, H., Brown, L. D., Gao, C., 2004. Scape: Supporting stereoscopic collaboration in augmented and projective environments. *IEEE Computer Graphics and Applications* 24 (1), 66–75.

Hurley, P., Boudec, J.-Y. L., Thiran, P., Kara, M., 2001. ABE: providing a low-delay service within best effort. *IEEE Network* 15 (3), 60–69.

IETF, 1998. RFC 2408 Internet Security Association and Key Management Protocol (ISAKMP).

IETF, 1999. RFC 2640 Internet Protocol version 6 (IPv6).

IETF, 2005a. RFC 4302 IP security (IPsec).

IETF, 2005b. RFC 4306 Internet Key Exchange (IKE).

IETF, 2007. Datagram congestion control protocol (DCCP).

Ikeda, S., Sato, T., Kanbara, M., Yokoya, N., 2004. An immersive telepresence system with a locomotion interface using high-resolution omnidirectional movies. In: *Proc. International Conference on Pattern Recognition (ICPR)*. pp. 396–399.

Ishida, T., Miyakawa, A., Shibata, Y., 2008. Collaborative and multimodal communications system using immersive virtual reality environment over ultrahigh-speed network. In: *Proc. International Conference on Distributed Computing Systems Workshops*. pp. 72–77.

ITU-T, May 2003. *Recommendation G.114 – One-way transmission time, general recommendations on the transmission quality for an entire international telephone connection.*

Iversen, V., 2007. *Teletraffic Engineering Handbook*. ITU.

Jaynes, C., Seales, W. B., Calvert, K., Fei, Z., Griffioen, J., 2003. The Metaverse: a networked collection of inexpensive, self-configuring, immersive environments. In: *Proc. Workshop on Virtual Environments*. pp. 115–124.

Jo, J., Hong, W., Lee, S., Kim, D., Kim, J., Byeon, O., 2006. Interactive 3D HD video transport for e-science collaboration over UCLP-enabled GLORIAD lightpath. *Future Generation Computer Systems* 22 (8), 884–891.

Jung, S. H., Bajcsy, R., 2006. A framework for constructing real-time immersive environments for training physical activities. *Journal of Multimedia* 1 (7), 9–17.

Katabi, D., Handly, M., Rohrs, C., 2002. Congestion control for high-bandwidths-delay product networks. *ACM SIGCOMM* 32 (4), 89–102.

Kim, J., Kim, H., Tay, B. K., Muniyandi, M., Srinivasan, M. A., Jordan, J., Mortensen, J., Oliveira, M., Slater, M., 2004. Transatlantic touch: a study of haptic collaboration over long distance. *Presence: Teleoperators and Virtual Environments* 13 (3), 328–337.

Kim, K., Low, S., 2003. Analysis and design of AQM based on state-space models for stabilizing TCP. In: *Proc. American Control Conference (Volume 1)*. pp. 260–265.

Kim, N., Woo, W., Kim, G. J., Park, C.-M., 2006. 3-D virtual studio for natural inter-"acting". *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans* 36 (4), 758–773.

Kitamura, M., Shirai, D., Kaneko, K., Murooka, T., Sawabe, T., Fujii, T., Takahara, A., 2011. Beyond 4K: 8K 60p live video streaming to multiple sites. *Future Generation Computer Systems* 27 (7), 952–959.

Kurillo, G., Bajcsy, R., 2012. 3D teleimmersion for collaboration and interaction of geographically distributed users. *Virtual Reality* 17 (1), 29–43.

Kurillo, G., Bajcsy, R., Nahrsted, K., Kreylos, O., 2008. Immersive 3D environment for remote collaboration and training of physical activities. In: *Proc. IEEE Virtual Reality Conference*. pp. 269–270.

Lazaris, A., Koutsakis, P., 2010. Modeling multiplexed traffic from H.264/AVC videoconference streams. *Computer Communications* 33 (2010), 1235–1242.

Lee, S. Y., Ahn, S. C., Kim, H. G., Lim, M. T., 2006. Real-time 3D video avatar in mixed reality: An implementation for immersive telecommunication. *Simulation Gaming* 37 (4), 491–506.

Leong, C. W., Xing, Y., Georganas, N. D., 2008. Tele-immersive systems. In: *Proc. IEEE International Workshop on Haptic Audio Visual Environments and their Applications (HAVE)*. pp. 81–86.

Lie, A., 2007. P-AQM: Low-delay max-min fairness streaming of scalable real-time CBR and VBR media. In: *Proc. IASTED International Conference on Internet and Multimedia Systems and Applications*. pp. 140–149.

Lie, A., 2008. *Enhancing rate adaptive IP streaming media performance with the use of active queue management*. Ph.D. thesis, ITEM, NTNU.

Lie, A., Aamo, O., Rønningen, L., 2004a. On the use of classical control system based AQM for rate adaptive streaming media. In: *Proc. Nordic Teletraffic Seminar*.

Lie, A., Aamo, O., Rønningen, L., 2004b. Optimization of active queue management based on proportional control system. In: *Proc. IASTED Communications, Internet, and Information Technology*. pp. 69–74.

Lie, A., Aamo, O. M., Rønningen, L., 2005. A performance comparison study of DCCP and a method with non-binary congestion metrics for streaming media rate control. In: *Proc. International Teletraffic Congress*. pp. 153–162.

Lien, J.-M., Kurillo, G., Bajcsy, R., 2009. Multi-camera tele-immersion system with real-time model driven data compression: A new model-based compression method for massive dynamic point data. *The Visual Computer: International Journal of Computer Graphics* 26 (1), 3–15.

Lu, Y., Zhao, Y., Kuipers, F., Van Mieghem, P., 2010. Measurement study of multi-party video conferencing. In: *Proc. IFIP TC 6 international conference on Networking*. pp. 96–108.

Lunenburger, L., Wellner, M., Banz, R., Colombo, G., 2007. Combining immersive virtual environments with robot-aided gait training. In: *Proc. IEEE International Conference on Rehabilitation Robotics*. pp. 421–424.

Maimone, A., Bidwell, J., Peng, K., Fuchs, H., 2012. Enhanced personal autostereoscopic telepresence system using commodity depth cameras. *Computers & Graphics* 36 (7), 791–807.

Maimone, A., Fuchs, H., 2011. Encumbrance-free telepresence system with real-time 3d capture and display using commodity depth cameras. In: *Proc. IEEE international symposium on mixed and augmented reality (ISMAR)*. pp. 137–146.

Mouchtaris, A., Tsakalides, P., 2009. The ASPIRE project - sensor networks for immersive multimedia environments. *ERCIM News* 78, 38–39.

Naveen, K., Venkatram, V., Vaidya, C., Nicholas, S., Allan, S., Charles, Z., Gideon, G., Jason, L., Andrew, J., 2004. SAGE: the Scalable Adaptive Graphics Environment. In: *Proc. Workshop on Advanced Collaborative Environments*.

Nechvatal, J., 2009. *Immersive Ideals/Critical Distances: A Study of the Affinity Between Artistic Ideologies Based in Virtual Reality and Previous Immersive Idioms*. LAP Lambert Academic Publishing.

Patel, K., Bailenson, J. N., Jung, S. H., Diankov, R., Bajcsy, R., 2006. The effects of fully immersive virtual reality on the learning of physical tasks. In: *Proc. Annual International Workshop on Presence*. pp. 87–94.

Perkins, C., Gharai, L., 2004. Real-time collaborative environments and the grid. In: *Proc. Workshop on Advanced Collaborative Environments*.

Perkins, C., Gharai, L., Lehman, T., Mankin, A., 2002. Experiments with delivery of HDTV over IP networks. In: *Proc. International Packet Video Workshop*.

Petit, B., Dupeux, T., Bossavit, B., Legaux, J., Raffin, B., Melin, E., Franco, J.-S., Assenmacher, I., Boyer, E., 2010. A 3d data intensive tele-immersive grid. In: *Proc. ACM International Conference on Multimedia*. pp. 1315–1318.

Ramakrishnan, K., Floyd, S., Black, D., 2001. RFC 3168 The Addition of Explicit Congestion Notification (ECN) to IP. IETF.

Ramamurthy, P., Blundell, M., Bastien, C., Zhang, Y., 2012. Computer simulation of real-world vehicle-pedestrian impacts. *International Journal of Crashworthiness* 16 (4), 351–363.

Renambot, L., Jeong, B., Hur, H., Johnson, A., Leigh, J., 2009. Enabling high resolution collaborative visualization in display rich virtual organizations. *Future Generation Computer Systems* 25 (2), 161–168.

Renambot, L., Jeong, B., Leigh, J., 2007. Real-time compression for high-resolution content. In: *Proc. Access Grid Retreat*.

Rhee, S.-m., Ziegler, R., Park, J., Naef, M., Gross, M., Kim, M. H., 2007. Low-cost telepresence for collaborative virtual environments. *IEEE Transactions on Visualization and Computer Graphics* 13 (1), 156–166.

Roelofs, G., 2003. *PNG: The Definitive Guide*. OReilly.

Rønningen, L. A., 1982. Input traffic shaping. In: *Proc. International Teletraffic Congress*.

Rønningen, L. A., 1999. *The Combined Digital Satellite Broadcast and Internet System*. Tech. rep., Telenor Satellite Services.

Rønningen, L. A., 2005. Adaptive scene and traffic control in DMP. In: *Proc. IADAT International Conference on Telecommunications and Computer Networks*.

Rønningen, L. A., 2011a. *DMP Processing Architectures based on FPGA and PCIe*. Tech. rep., ITEM, NTNU.

Rønningen, L. A., 2011b. *The Distributed Multimedia Plays Architecture (version 3.20)*. Tech. rep., ITEM, NTNU.

Rønningen, L. A., 2012. *Collaboration Spaces, Camera Arrays and Sparse Aperture*. Tech. rep., ITEM, NTNU.

Rønningen, L. A., Chilwan, A., 2012. Packet handling and quality shaping for real-time services in futuristic Internet. In: *in Proc. International Conference on Frontiers of Information Technology*. pp. 1–6.

Rønningen, L. A., Heiberg, E., 2009. Perception of time variable quality of scene objects. In: *Proc. SPIE Image Quality and System Performance VI*. pp. 72420X–72420X–12.

Santa-Cruz, D., Ebrahimi, T., Askelof, J., Larsson, M., Christopoulos, C., 2000. JPEG 2000 still image coding versus other standards. In: *Proc. SPIE Applications of Digital Image Processing XXIII, vol. 4115*.

Sawchuk, A. A., Chew, E., Zimmermann, R., Papadopoulos, C., Kyriakakis, C., 2003. From remote media immersion to distributed immersive performance. In: *Proc. ACM SIGMM Workshop on Experiential Telepresence*. pp. 110–120.

Schelkens, P., Skodras, A., Ebrahimi, T., 2009. *The JPEG 2000 Suite*. Wiley, Series: Wiley-IS&T Series in Imaging Science and Technology.

Schreer, O., Feldmann, I., Atzpadin, N., Eisert, P., Kauff, P., Belt, H., 2008. 3DPresence - a system concept for multi-user and multi-party immersive 3D videoconferencing. In: *Proc. European Conference on Visual Media Production (CVMP)*. pp. 1–8.

Sheppard, R. M., Kamali, M., Rivas, R., Tamai, M., Yang, Z., Wu, W., Nahrstedt, K., 2008. Advancing interactive collaborative mediums through tele-immersive dance (TED): a symbiotic creativity and design environment for art and computer science. In: *Proc. ACM International Conference on Multimedia*. pp. 579–588.

Shimizu, T., Shirai, D., Takahashi, H., Murooka, T., Obana, K., Tonomura, Y., Inoue, T., Yamaguchi, T., Fujii, T., Ohta, N., Ono, S., Aoyama, T., Herr, L., van Osdol, N., Wang, X., Brown, M., DeFanti, T., Feld, R., Balser, J., Morris, S., Henthorn, T., Dawe, G., Otto, P., Smarr, L., 2006. International real-time streaming of 4K digital cinema. *Future Generation Computer Systems* 22 (8), 929–939.

Shirai, D., Kawano, T., Fujii, T., Kaneko, K., Ohta, N., Ono, S., Arai, S., Ogoshi, T., 2009. Real time switching and streaming transmission of uncompressed 4K motion pictures. *Future Generation Computer Systems* 25 (2), 192–197.

Shirai, D., Kitamura, M., Fujii, T., Takahara, A., Kaneko, K., Ohta, N., 2011. Multi-point 4K/2K layered video streaming for remote collaboration. *Future Generation Computer Systems* 27 (7), 986–990.

Taubman, D., 2000. High performance scalable image compression with EBCOT. *IEEE Transactions on Image Processing* 9 (7), 1158–1170.

Taubman, D. S., Marcellin, M. W., 2001. *JPEG 2000: Image Compression Fundamentals, Standards and Practice*. Kluwer Academic Publishers.

The Climate Group, 2008. *Smart 2020: Enabling the low carbon economy in the information age, The Smart 2020 Report*. http://www.smart2020.org.

Turner, J., 2002. New directions in communications (or which way to the information age?). *IEEE Communications Magazine* 40 (5), 50–57.

van Waveren, J., 2006. *Real-time DXT compression*. Tech. rep., Id Software, Inc.

Vasudevan, R., Kurillo, G., Lobaton, E., Bernardin, T., Kreylos, O., Bajcsy, R., Nahrstedt, K., 2011. High-quality visualization for geographically distributed 3-D teleimmersive applications. *IEEE Transactions on Multimedia* 13 (3), 573–584.

Willert, M., Ohl, S., Staadt, O., 2012. Reducing bandwidth consumption in parallel networked telepresence environments. In: *Proc. ACM SIGGRAPH International Conference on Virtual-Reality Continuum and its Applications in Industry*. pp. 247–254.

Wolff, R., Roberts, D. J., Steed, A., Otto, O., 2007. A review of telecollaboration technologies with respect to closely coupled collaboration. *International Journal of Computer Applications in Technology* 29 (1), 11–26.

Wu, W., Yang, Z., Jing, D., Nahrstedt, K., 2008. Implementing a distributed 3d teleimmersive system. In: *Proc. IEEE International Symposium on Multimedia*. pp. 477–484.

Yang, Z., Nahrstedt, K., Cui, Y., Yu, B., Liang, J., Jung, S. H., Bajcsy, R., 2005. TEEVE: The next generation architecture for tele-immersive environments. In: *Proc. IEEE International Symposium on Multimedia (ISM)*. pp. 112–119.

Yang, Z., Wu, W., Nahrstedt, K., Kurillo, G., Bajcsy, R., 2007. Viewcast: View dissemination and management for multiparty 3d tele-immersive environments. In: *Proc. ACM International Conference on Multimedia*. pp. 882–891.

Zhang, S., Ho, W. C., 2012. Tele-immersive interaction with intelligent virtual agents based on real-time 3D modeling. *Journal of Multimedia* 7 (1), 57–65.

Zimmermann, R., Chew, E., Ay, S. A., Pawar, M., 2008. Distributed musical performances: Architecture and stream management. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)* 4 (2), 14:1–14:23.

PART II

Included Papers

PAPER **A**

Modeling and simulating motions of human bodies in a futuristic distributed tele-immersive collaboration system for synthesizing transient input traffic

Mauritz Panggabean, Leif Arne Rønningen, and Harald Øverby

This paper, in the original version, has been published in *Simulation Modelling Practice and Theory*, vol. 31, 2012, pp. 132–148 (Elsevier, 2012 Impact Factor: 1.159).

Digital Object Identifier: 10.1016/j.simpat.2012.10.010

Abstract

In this paper we model human body as a discrete-event system that enables applying DES to forward kinematics of stochastic and deterministic motions of a human body. Human gait cycles for walking and running are examples of the latter. Fast and flexible DES is made possible by LI for both motion types. It gives close approximation to FFS that model natural gait cycles. We use the simulator as the building block in simulating a futuristic tele-immersive collaboration system with arbitrary multi-actor collaboration scenario from remote locations. The silhouette area from the visualization of simulated moving human bodies is the main feature in synthesizing transient traffic that would result from such collaboration system. It will be useful as the input traffic for DES of arbitrary networks in future work. We demonstrate how to apply the simulation framework to simulate a scenario of collaborative dancing and singing that involves four performers with both motion types from two different places. Through the scenario we also show how the simulator works as a novel transient-traffic generator.

A.1 Introduction

A vision of a futuristic tele-immersive collaboration system [Rønningen et al. (2010)] sets the background and context of the work reported in this paper. It would be a network of interconnected CSs at remote places in and through which people could work together and perceive as if they were all in the same place. To achieve the near-natural quality of experience, all the surfaces of the CS would be composed of seamless arrays of high-end display panels with speakers and sensors such as cameras and microphones. The autostereoscopic multiview 3D displays would have high spatial resolution due to the life-size dimensions of the collaborators. Their bodies and the other important objects would be segmented using background subtraction to be displayed on the corresponding surface of the CS or on public displays fed by transmission that is not sensitive to delay. The background on the displays would be able to be customized and depicted from a local digital library. The selection would depend on the preferred theme and setting of the collaboration, such as a serene beach, a busy business district, or simply a stage. The environment would become more powerful when the display panels support multi-touch sensing. These features allow the system to facilitate various forms of delay-sensitive real-time full-duplex collaborations which are more complex and creative than those in standard teleconference. Other functions, such as immersive games and cinema, can also be supported. More detailed description about the system will follow later with an exemplary scenario of collaborative singing and dancing.

At the moment the exact system as we envision does not exist yet. At least four main challenges discussed in the next section stand on the path towards its realization. Nevertheless, it is a foreseeable vision that is worthy of in-depth study, given the rapid advances in electronics that produce faster, smaller and more powerful devices with cheaper prices and lower power consumption. It is evident from recent constructions of similar systems reported, for example, in [Maimone et al. (2012); Vasudevan et al. (2011); Zhou et al. (2011); DeFanti et al. (2011)]. The first briefly covers the progress in this

endeavor since 1998. The immense complexity of the system makes analytical approach highly improbable for its design and study. Thus this work focuses on simulation of such system which we find still a void in this area. We choose DES from various simulation methods [Law and Kelton (2000)] due to its powerful ability in emulating complex dynamics of a discrete event system [Ingalls (2001)].

It leads us to introduce the two main objectives of this work. First, we aim at constructing a framework to model and simulate arbitrary scenario in such complex collaboration system using DES. As the collaboration is essentially a complex human interaction, this objective implies modeling and simulating motions of human bodies. Moreover, the forms of collaborations to be supported in the system would require guaranteed maximum delay which is much lower than that in teleconference system today. Rønningen (2011, 2007) proposed a novel futuristic network architecture called DMP that is designed to deliver such guarantee with graceful VQ degradation. In-depth study and test of DMP using DES ideally needs input traffic from the envisioned collaboration system. However, such input traffic must be synthesized because the system is still nonexistent. Thus the second objective is to synthesize such traffic using the modeled and simulated collaboration system from the first objective. The traffic will be used later as the appropriate input for DES of DMP as future work. As explained later, the second objective justifies important simplifications for the first one. The phases undertaken in this work to achieve the objectives are summarized in Figure A.1.

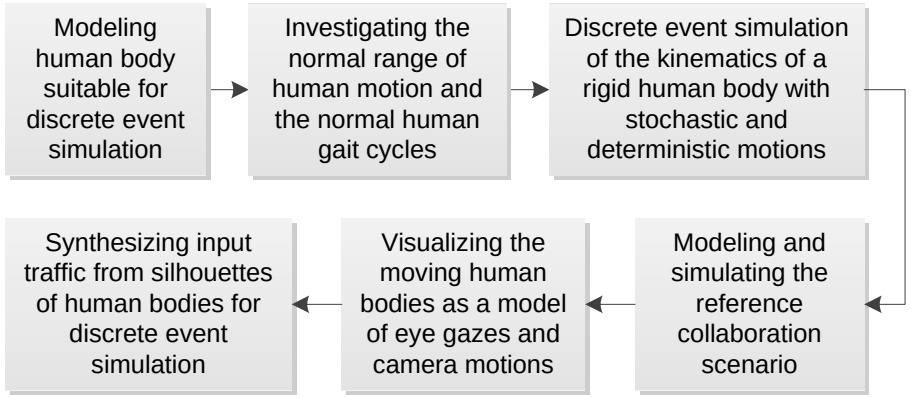


Figure A.1: The overview of the phases in this work to achieve its two objectives.

The novel contributions from addressing the two objectives are the following. With respect to the first one, we propose a comprehensive simulation framework that can be used to model and simulate arbitrary complex collaboration scenario for the envisioned system. It consists of instances of moving human body that are independently animated via forward kinematics using object-oriented DES that produces uniquely reproducible motions. We combine both stochastic human motion, using the natural ROM for human from literature, and the deterministic motion in the form of human gait cycles for walking and running. LI and FFS are used in simulating the forward kinematics of motions, where the latter is devoted for modeling the gait cycles. Exploiting valid simplifications from the second objective, LI matches the proposed model of human body and enables fast DES. We show that LI offers close approximation to the natural

functions in FFS and higher flexibility for modeling and simulating gait cycles and hand swing with arbitrary ROM. Then we provide an exemplary application that demonstrates how the simulator serves as the building block in modeling, simulating and visualizing arbitrary scenario of multi-actor interaction in the envisioned collaboration system. Thus the properties and performance of such complex system can be studied using the simulation framework despite its nonexistence.

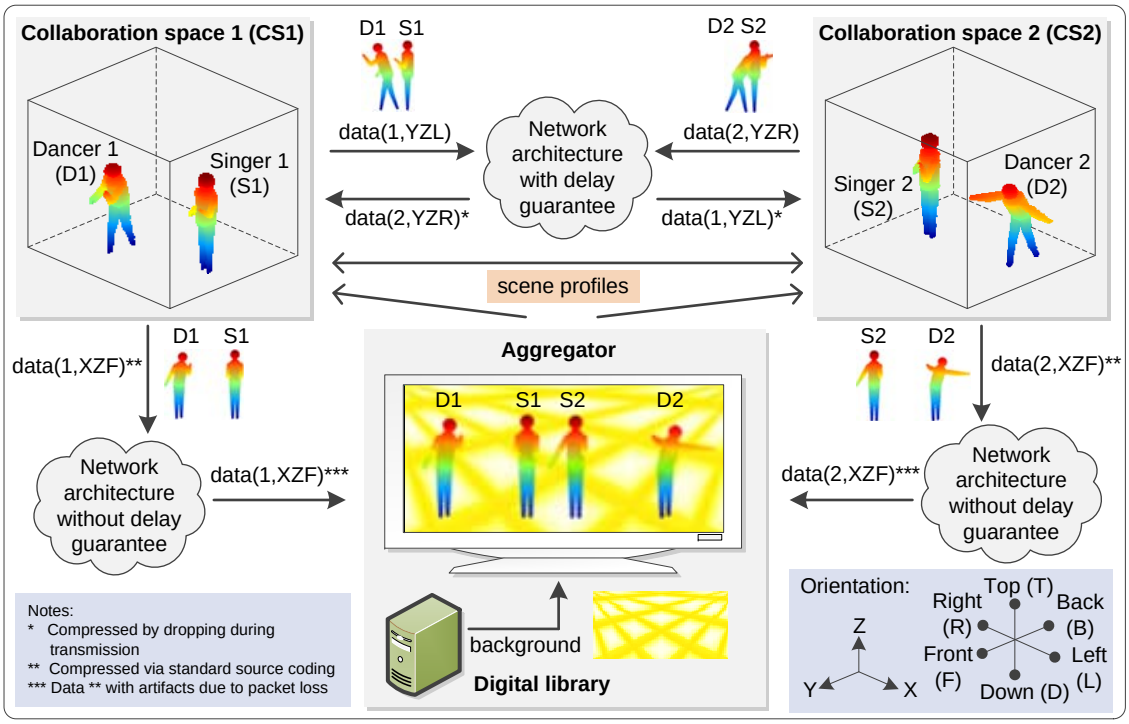
With respect to the second objective, our empirical results show that the silhouette area of the simulated and visualized moving human bodies is sufficient as the main feature to synthesize transient traffic that would be produced from the collaboration scenario. The synthesized traffic will be used in future work as the input traffic for DES of arbitrary network architecture that supports the collaboration system, particularly the DMP in our case. This means that the simulator framework also functions as a novel transient-traffic generator for DES. The advances made by this work for the state of the art reported in literature are exposed later in the relevant sections.

This paper adopts the following structure. Section A.2 starts with an example of collaboration scenario as the reference in this work. Section A.3 covers a model of human body as a discrete-event system, the ROM for stochastic human motion, and the basic kinematics of rigid human body in 3D space. Human gait cycles for walking and running as the deterministic human motion are addressed in Section A.4. All these underlie the DES algorithms of stochastic and deterministic human motions detailed in Sections A.5. Section A.6 shows how the silhouette areas of moving human bodies can serve as the main feature for synthesizing the transient traffic from a CS. Some results from simulation and visualization of both motion types are presented and discussed in Section F.3. Section B.4 covers the application of the simulator to achieve the two objectives on the reference collaboration scenario. Concluding remarks and some ideas for future outlook in Section F.4 complete the paper.

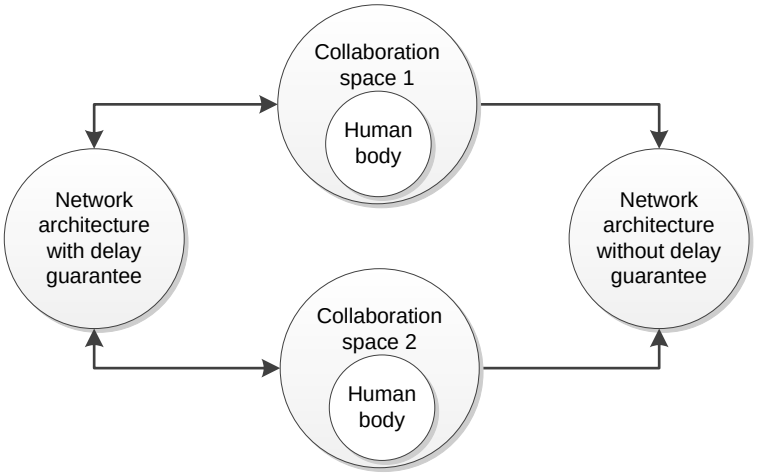
A.2 Reference collaboration scenario

Figure A.2 (a) shows a collaboration scenario of the envisioned system where two pairs of singer and dancer perform together from two different places, each in their own CS. S1 and D1 denote the singer and dancer in collaboration space 1 (CS1), respectively, and hence S2 and D2 in CS2. Seen from the front on the aggregator display, from left to right, are D1, S1, S2 and D2. The aggregator models public display via broadcasting, thus non-sensitive to delay. Both singers interact with each other as if they were in same place side by side. The interactions are typically be indicated by visual and physical cues such as eye gazes and gestures. Thus when a singer, for example S1, looks at the other one, then S2 and D2 must be displayed on the corresponding surface, i.e. YZL in CS1. In other words, the eye gaze of S1 determines which camera must be active on which surface in CS2, and vice versa. It makes detection and tracking of eye gazes very crucial as they are part of the control information distributed among the connected CS.

On the other hand, when S1 and S2 look at the virtual audience on the front surface, the transmission of visual data between CS1 and CS2 is no longer necessary. In addition, the dimension of S2 and D2 must be adjusted to be realistic according to their distance



a



b

Figure A.2: The reference system with a scenario of a real-time delay-sensitive artistic collaboration between dancers and singers from two remote locations (a). The interconnected instances of elementary entities that construct the reference system (b).

and position with respect to S1. The depth is provided by the multiview autostereoscopic 3D displays.

Here we assume that the dancers perform more varied motions than the singers. The dancers can walk a few steps forward and backward randomly while the singers are immobile. The occurrences and durations of the occlusions between the pair in a CS will

logically yield the transient traffic coming from, for instance, the YZL and YZR surfaces. This scenario is chosen as the reference and example for application in Section B.4 because it exhibits stochastic motion as 'stochastic' dances and deterministic motion as walking cycles by the dancers. All these motions are more complex and sensitive to EED than those supported in today's teleconference systems.

At least four major technical challenges arise from the system description above. First, the amount of data for exchange within the system can be very high. Second, the data size can fluctuate frequently as transient periods, even from zero to maximum. Moreover, the real-time nature of the collaborations to be performed in the system poses the third major challenge. Studies have shown that smooth synchronization among performers in real-time delay-sensitive collaborations demands very low EED. For example, Chafe et al. (2004) reported that the optimal EED for synchronizing rhythmic clapping hands from different places is 11.5ms. They disclosed that longer delays will produce increasingly severe tempo deceleration while shorter ones yield a modest yet surprising acceleration.

Percussions are rhythmically very similar to clapping hands, so collaborative musicians playing such instruments will require the same delay for audiovisual data. The study on collaborative dancing by Yang et al. (2006) also indicates the importance of guaranteeing a maximum delay, particularly for video data, to enable good synchronization between dancers because it depends on visual cues. The same also applies to collaborative singing and remote conducting [Rønningen and Wittner (2011)]. Finally, the envisioned near-natural QoE will be impossible without graceful VQ degradation perceived by the performers in the system. This is the fourth major challenge.

Any collaboration scenario on the system essentially consists of four entities, see Figure A.2 (b). The instances of the entities are interconnected in ways that construct the system. The moving body of a human performer inside a CS is evidently the basis for simulating the system. Let us proceed with modeling and animating human body.

A.3 Human body and the motion as discrete event system

Many models of human body have been reported in literature. The complexity of a humanoid model depends on the goal for which it is proposed and used. Observation on reported work with models of human body would reveal that they mostly address two major areas in computer vision and computer graphics. The first is related to capture, tracking, analysis and recognition of human motion from images and videos, for instance those recently reviewed in [Aggarwal and Ryoo (2011); Ji and Liu (2010)]. The second concerns with animation of virtual human as realistically as possible, for example the complex emulation of ballet dances in [LaViers et al. (2011a); LaViers and Egerstedt (2011); LaViers et al. (2011b)] and all the related work in computer graphics such as those for computer games. To our best knowledge, work that address exactly the two objectives aforementioned have not been reported. However, there are work on simulation of human motion with different environments and goals, such as in virtual simulation for better manufacturing processes [Fuo and Wang (2012)] and for

simulating real-world vehicle-pedestrian impacts [Ramamurthy et al. (2012)].

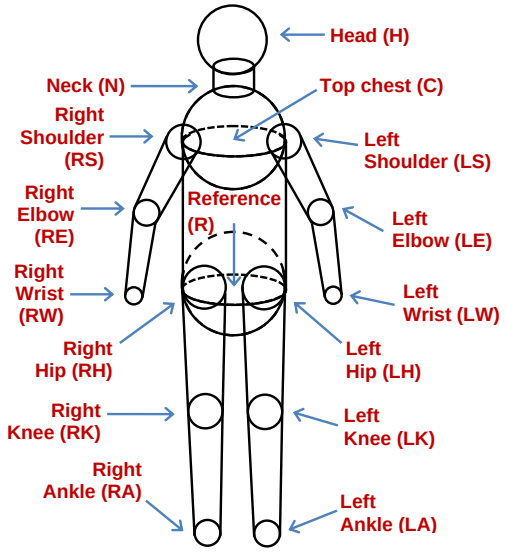
Zhmakin (2011) listed and discussed at least seven approaches to animating human body: kinematics, dynamics, motion controllers, animation of deformable objects, motion planning, autonomous character behaviour, and optimization methods. The problem and objectives that we address determine the most appropriate approach to use. Our second objective implies constructing a novel generator of transient traffic that will be used for later DES. DEMOS [Birtwistle (2003)] is an excellent and fast DES tool that has been used extensively for simulating various networks in wired, wireless and optical domains, including the DMP. It is then preferred to implement the traffic generator also in DEMOS. This means that the approach must be compatible to DES and fast to compute. Moreover the motion generated by the approach must be acceptable with respect to the objectives.

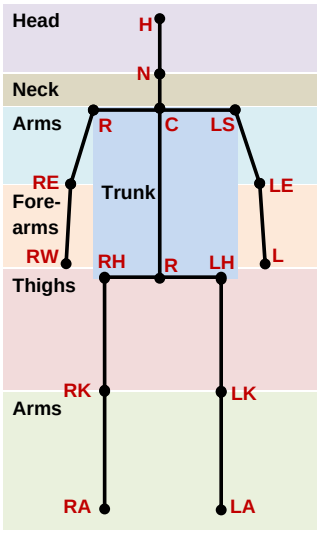
As explained shortly, the state of the human body in kinematics as the system can be simply defined by the position and orientation of the joints. This makes kinematics suitable to DES by viewing human body as a discrete event system. Then we have to choose either forward or inverse kinematics. Forward kinematics changes the angles of the joints which alter the transformation matrix that produces the new positions. The process is reversed in inverse kinematics: given the current and next positions of the joints, the matrix must be computed. It makes the computations much more intensive and time consuming than those in forward kinematics, but it yields more general and realistic motions [Granieri et al. (1995)]. However, as shown in Section A.6, the problem in this work tolerates the less realistic motions resulting from forward kinematics. After modeling human body as a discrete event system for DES, we discuss the human ROM and the forward kinematics of rigid human body in 3D space.

A.3.1 A model of human body and the range of motion for DES

Figure A.3 shows the frontal side of the model of human body suitable for DES as a simplified version of the model by [Hatze (1980)]. The model describes a human body as a discrete event system that consists of ten *links* or *limbs*: the head and neck combined for simplification, trunk, arms, forearms, thighs and legs. The feet, palms and fingers are excluded for simplification since their silhouette areas are not significant, cf. Section A.6. Each limb would be visualized later either as a sphere or a cylinder with two radii and a length that are defined a priori as simulation parameters. A contact at a *joint* connects two links, except at the terminal joints. Thus the model entails sixteen essential joints that belong to and move with the links to which they are attached. In this work the positions and orientations resulting from the simulated forward kinematics in 3D space belong to the joints instead of the links. The surface area centered at a joint is modelled with a sphere which radius also becomes that of the link cylinder connected to the joint.

The first step to make the simulation realistic and accurate is to constrain the simulated human motion within the normal ROM for human. *Range of motion* is the angular or linear distance that can be normally achieved by a movable human link with its motion given proper attachment to another link via the connecting joint. Each joint has movement axes around which it moves. The relationship of the muscles to the axes





Links	Connected joints
Head	N – H
Neck	C – N
Trunk	C, R LS, RS LH, RH center hip as reference (R)
Arms (shoulder - elbow)	LS – LE RS – RE
Forearms (elbow - wrist)	LE – LW RE – RW
Thighs (hip - knee)	LH – LK RH – RK
Leg (knee - ankle)	LK – LA RK – RA

Figure A.3: The frontal side of the model of human body as discrete event system with the connected spheres and cylinders for visualization later (left). The skeleton of the model with the essential links and joints for DES of forward kinematics (middle). The list of the included links and the attached joints (right).

determines the direction of the movement. Opposite movements that can occur around each axis at a joint include: *bending - extending* (*flexion - extension*) as in elbows, *rolling inward - rolling outward* (*inner rotation - outer rotation*) as in shoulders, as well as *pushing out - pulling in* (*abduction - adduction*) and *forward motion - backward motion* (*flexion - extension*) as in hips [Faller et al. (2004)]. The normal ROM included in this work for the human links with the opposite movements are illustrated in Figure A.4 [Faller et al. (2004); National Aeronautics and Space Administration (NASA) (1995)]. The minimum and maximum values of the angles for the ROM with respect to forward kinematics are detailed in Figure A.5. Note that all angles are measured against the vertical, which is indicated by 0 degree. Negative signs refer to the counter direction from that of the positive values.

A.3.2 Forward kinematics of rigid human body in motion

The distance between any two points on every rigid body does not change over time as the object moves. Thus it is sufficient to model a rigid body as a point and its motion as the motion of one point. The humanoid model above also considers human body as a rigid body. The diagram on the top side of Figure A.5 illustrates the forward kinematics of rigid bodies [Jazar (2010)]. Consider point A_T as the position of a rigid body S at $t = T$ in the global coordinate frame $OXYZ$ or G. If from $t = 0$ to $t = T$ the rigid body S consecutively rotates α, β and γ degrees respectively around the X, Y, and Z axes of the global coordinate frame, then the coordinate of A_T is determined by the following transformation:

$$^{A}\mathbf{r}_T = {}^{G}R_S\, {}^{A}\mathbf{r}_0 \tag{A.1}$$

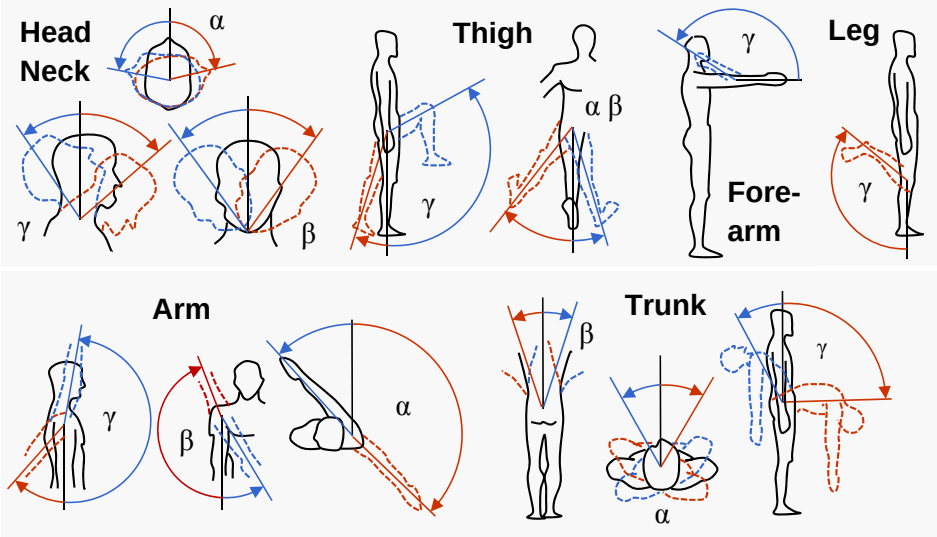


Figure A.4: The human ROM for head, neck, trunk, arms, forearms, thighs and legs [Faller et al. (2004); National Aeronautics and Space Administration (NASA) (1995)]. The angles α, β and γ for each link refer to the corresponding angles in mathematical models of forward kinematics. Opposite movements are indicated by in the positive (blue) and negative (red) signs of the angles.

where ${}^{G}R_S = R_{X,\gamma}R_{Y,\beta}R_{Z,\alpha}$ is the successive global rotation matrix, ${}^{A}\mathbf{r}_T = [x_T, y_T, z_T]^{\mathsf{T}}$, ${}^{A}\mathbf{r}_0 = [x_0, y_0, z_0]^{\mathsf{T}}$,

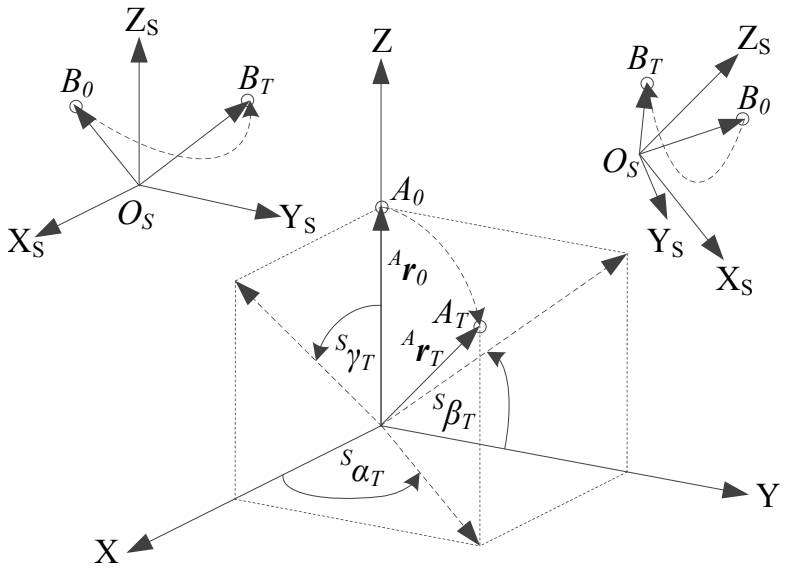
$$R_{Z,\alpha} = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}, R_{Y,\beta} = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix},$$

$$R_{X,\gamma} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{bmatrix}, \tag{A.2}$$

$${}^{G}R_S = \begin{bmatrix} \cos\alpha\cos\beta & -\cos\beta\sin\beta & \sin\beta \\ \cos\gamma\sin\alpha + \cos\alpha\sin\beta\sin\gamma & \cos\alpha\cos\gamma - \sin\alpha\sin\beta\sin\gamma & -\cos\beta\sin\gamma \\ \sin\alpha\sin\gamma - \cos\alpha\cos\gamma\sin\beta & \cos\alpha\sin\gamma + \cos\gamma\sin\alpha\sin\beta & \cos\beta\cos\gamma \end{bmatrix}.$$

${}^{G}R_S$ means the rotation of the local coordinate frame $S(oxyz)$ of the rigid body S that brings the axes of $oxyz$ on the corresponding global axes of $OXYZ$. The rotation matrix also rotates any other rigid body as long as it rotates with respect to the global reference frame. Consider a rigid body V modeled as point B that moves from $t = 0$ to $t = T$. The motion is shown by a trajectory with respect to the coordinate frame of the rigid body S. It illustrates the change to the rigid body V and its local coordinate frame from $t = 0$ to $t = T$. It is caused by the changes from $({}^{S}\alpha_0, {}^{S}\beta_0, {}^{S}\gamma_0)$ to $({}^{S}\alpha_T, {}^{S}\beta_T, {}^{S}\gamma_T)$. In this work, for $t_1 \le t \le t_2$, α_t is computed as the following LI [Granieri et al. (1995)]:

$$\alpha_t = \alpha_{t_1} + \frac{t_2 - t}{t_2 - t_1}(\alpha_{t_2} - \alpha_{t_1}). \tag{A.3}$$



Links	Joints	α	β	γ
Head	H	-70, 70	-63, 63	-60, 30
Trunk	N, C, LS, RS LH, RH, R	-30, 30	-40, 40	-90, 35
Arms	LE	-30, 130	-40, 170	-40, 170
	RE	-130, 30	-170, 40	-40, 170
Forearms	LW, RW	0	0	0, 150
Thighs	LK	-50, 45	-30, 45	-15, 90
	RK	-45, 50	-45, 30	-15, 90
Legs	LA, RA	0	0	-145, 0

Figure A.5: An illustration of forward kinematics of rigid bodies (top). The minimum and maximum ranges of α, β and γ in degrees for the simulated joints from the ROM adapted from Faller et al. (2004); National Aeronautics and Space Administration (NASA) (1995) (bottom).

It applies also for computing β_t and γ_t. LI will be used in the simulation, as detailed later in Section A.5. Furthermore, if V is attached to S yet both move independently, then the translational transformation must be included as an additional component. Thus, the coordinate of point B at $t = T$ in the global coordinate frame G equals

$$ {}^{G}\mathbf{r}_T^B = {}^{G}R_S\, {}^{S}R_V\, {}^{V}\mathbf{r}_0^B + {}^{G}\mathbf{r}_T^A. \tag{A.4} $$

Hence V is a child of S or S is the parent of V. Having covered the basics for stochastic motion, we continue with those for human gait cycles as deterministic human motion.

A.4 Human gait cycles as deterministic human motion

In this paper, gait denotes the manner of human walking and running. The two types are denoted by parameter $g = 1$ and $g = 2$, respectively. Observations on how a physically healthy person walks and runs reveals that both are periodic repetitions of similar patterns, namely *gait cycles*. The model of human body in Figure A.3 consists of the upper-half and lower-half parts since they can move independently. The head, neck, trunk, arms, forearms form the upper-half part, while the rest are for the lower half. The gait cycle applies to the lower-half part of human body. Although the arms and forearms might follow periodic motions while one walks or runs, their motion pattern is hard to generalize unlike gait cycle, because of no direct repetitious contact to earth.

A gait cycle begins with the contact of one foot with the ground and ends when the same foot contacts the ground again. This event is called *initial contact* (IC). Both walking and running gait cycles comprise two phases: *stance* and *swing*. Stance starts at IC and ends when the toe is off the ground, an event called *toe off* (TO). TO marks the beginning of the swing phase that ends at IC. Figure A.6 subdivides the gait cycles into sub-phases [Novacheck (1998)].

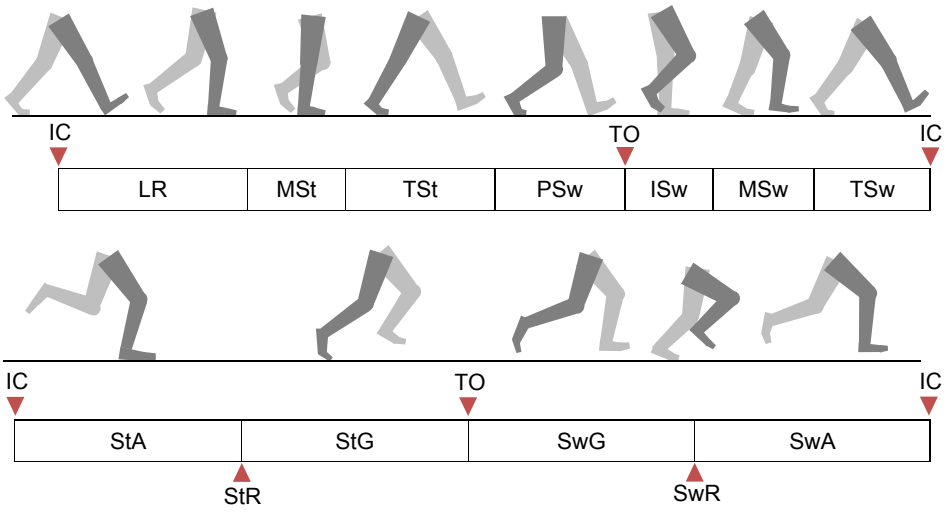
Figure A.6 (b;left) describes the gait cycles of the left and the right feet when walking and running [Novacheck (1998)]. Here a walking gait cycle is formed by approximately 60% stance and 40% swing, while that of running consists of approximately 35% stance and 65% swing [Novacheck (1998); Franz et al. (2009)]. Note that the stances of the left and the right feet overlap during walking. On the contrary, during a running cycle, one actually floats in the air two times. This is a main difference between walking and running. As for the cycle periods, the average speed and cycle length of adult persons are 1.28 ± 0.17 m/s and 1.35 ± 0.22 m during walking, and 3.17 ± 0.4 m/s and 2.30 ± 0.29 m during running, respectively [Franz et al. (2009)]. These yield 1.0 second as the approximate cycle period for walking and 0.7 second for running. These periods are used in our simulation that gives the results exemplified the next section.

Figure A.6 (b;right) depicts the three important angles in this study: the thigh extension χ, the thigh flexion ψ, and the leg flexion ω. As explained in Subsection A.3.1, here these angles are equivalent to hip extension, hip flexion, and knee flexion, respectively. The normal values of the angles from extensive measurements [Novacheck (1998); Franz et al. (2009)] can be fitted to well-known mathematical functions to enable simulation. With only a few parameters, the data are very closely approximated by the following *finite Fourier series* (FFS)

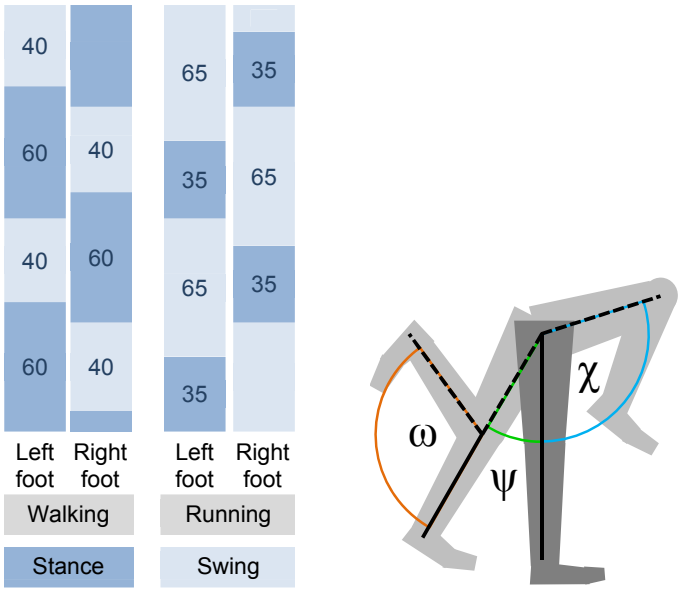
$$f(x,g) = a_{g,0} + \sum_{i=1}^{I} [a_{g,i}\cos(ix\theta_g) + b_{g,i}\sin(ix\theta_g)] \tag{A.5}$$

where $I = 4$, $0 \le x \le 100$. We will use FFS in our simulation due to its relatively high accuracy and simplicity. Together with the Fourier parameter values, Figure A.7 depicts the resulting plots of the ROM of the three angles in normal human gait cycles from [Novacheck (1998); Franz et al. (2009)] for walking and running.

When a person walks or runs, almost all parts of the body move. However the FFS in Equation (A.5) models only the motion of the thighs (knees) and the legs (ankles), due to



(a)



(b)

Figure A.6: **(a)** The gait cycle for walking (top) and running (bottom) of a physically healthy person with their components as sub-phases [Novacheck (1998)]: initial contact (IC), toe off (TO), *loading response* (LR), *midstance* (MSt), *terminal stance* (TSt), *preswing* (PSw), *initial swing* (ISw), *midswing* (MSw), *terminal swing* (TSw), *stance stance reversal* (StR), and *swing reversal* (SwR). **(b)** Comparing the gait cycle of the left and the right feet during walking and running [Novacheck (1998)] (left). The three important angles in our simulation of human gait cycles: the thigh extension χ, the thigh flexion ψ, and the leg flexion ω (right).

two reasons. First, the motion of the lower-half part of the body in walking and running is in principle similar for most healthy people. Second, the thighs and legs contribute quite significantly to the areas of the whole body. Including feet and fingers makes the calculations more complex with insignificant effect to the output transient traffic.

Despite high accuracy of the FFS in modeling the natural human gait cycles, it lacks flexibility from DES point of view. Changes to the curves in Figure A.7 might require a totally different set of parameter values. Moreover, it is a fact that hand swing usually accompany gait cycles. However, unlike gait cycles, the upper-half part of the body may move differently for different people, e.g. the way they swing their hands while walking or running, making it harder to generalize and model them mathematically. These argue for a need to closely approximate gait cycles with hand swing for DES. LI is a good candidate to start with for forward kinematics. Two sets of limbs are involved in gait cycles with hand swing: [LS, LW, RK, RA] and [RS, RW, LK, LA]. Normally when one set moves forward as indicated, for example, by $\gamma^{RK} > 0$, the other set moves backward, here by $\gamma^{LK} < 0$. As in stochastic case, LI needs minimum and maximum angles as ROM for the two sets. Table A.1 presents the sets of ROM for knees and ankles that correspond to Figure A.7, as well as for arms and wrists to include the hand swing.

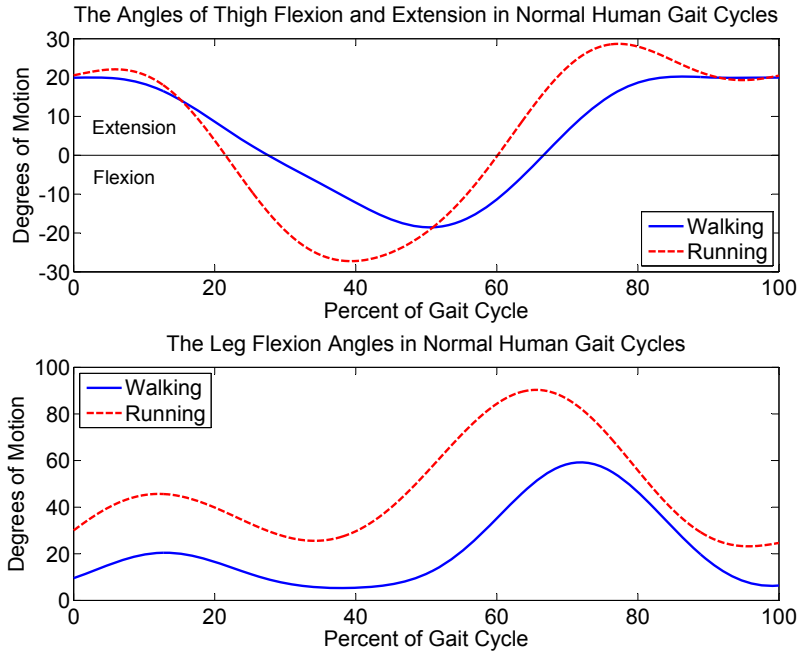
Table A.1: The ROM sets in degrees for simulating gait cycles with hand swing using LI

Joint set 1	Joint set 2	$\gamma_{\text{walk,forward}}$	$\gamma_{\text{walk,backward}}$	$\gamma_{\text{run,forward}}$	$\gamma_{\text{run,backward}}$
LE	RE	−10	10	−30	5
LW	RW	10	20	90	120
RK	LK	20	−20	25	−25
RA	LA	−5	−15	−25	−35

Note that other purposes than motion synthesis may need different models of human body and cyclic motion. For example, tracking of such motion in video data aims for highest accuracy despite the difficulties such as severe self-occlusion and non-linear dynamics of the limbs. Thus the purpose of tracking, which is in essence analysis and more difficult than our objectives, entails more complex model of human cyclic motion. Chang et al. (2004), for example, modelled human body as a hybrid graph for tracking a cyclic motion. The complex cyclic motion is decomposed into component motions and the phase coupling between them is maintained. Assuming the phase locking relationship for a given motion pattern is known *a priori*, it can be used to track unknown component motion from the known. Although tracking is not relevant to our synthesis, the phase relationship implies the dependency between limbs shown by the parent-child hierarchy in forward kinematics.

A.5 DES and visualization of stochastic and deterministic human motion

A state in this simulation denotes the set of instantaneous positions of the sixteen joints. A *transition event* refers to the change of a limb's position to a new one indicated by the change in the corresponding angle. The duration of the transition is denoted by T_{next}. It is a fact that a limb might stay at its new current position for a certain period before moving to a new one. Therefore in this simulation, after arriving at its new position, a limb stays at it for a random period of time T_{set} before moving to the next position.



FFS parameter	Knee (LK, RK)		Ankle (LA, RA)	
	Walk	Run	Walk	Run
θ	0.06532	0.06271	0.06004	0.04354
a_0	4.15	5.003	21.95	38.79
a_1	19.54	20.89	-5.308	-21.06
b_1	-1.552	-15.84	-16.85	13
a_2	-2.968	-4.627	-9.026	14.49
b_2	-1.479	7.347	13.07	-1.549
a_3	-0.4995	-0.742	2.51	-4.813
b_3	1.469	2.677	3.237	13.92
a_4	-0.2677	0.05066	-0.58	2.608
b_4	0.1403	-0.2478	-0.4924	0.1292

Figure A.7: The ROM for thighs (top) and legs (middle) in normal human gait cycles for walking and running and the parameter values of the FFS function (bottom). Changes in the plots will alter the values, and vice versa.

This is called the *stationary event* despite no change in the state. Thus the transition and stationary events for a joint always occur consecutively during the simulation. The events from the joints are considered fully independent from each other and seen as a Poisson process, a fundamental concept in various stochastic fields, including queuing theory [Iversen (2007)]. Thus, the time between the events of a limb can be assumed to follow negative exponential distribution (n.e.d).

DEMOS [Birtwistle (2003)] is the chosen simulation tool for this work because it fulfills all criteria of a good tool for DES. In addition to its simplicity and fast operation, the support of object-oriented simulation in DEMOS enables creating every joint as an independent object. DEMOS also provides built-in random number generators, including those based on n.e.d and uniform distribution. Thus the duration of the events in T_{next} and T_{set}, i.e. the motion speed, can be controlled by setting the respective mean values λ_{next} and λ_{set}. The next position for each joint, denoted by α_{next}, β_{next} and γ_{next}, is generated using uniform distribution with the ranges specified in Figure A.5.

Table A.2 summarizes all the input parameters for simulating the stochastic and deterministic motion using DEMOS. They must be defined prior to running the simulation simply by updating the values in the input parameter file read by the simulator when it is started. The initial positions, lengths and the radii of the links for simulation and visualization are detailed in Table A.3. Here ${}^{\text{K}}d_{\text{J}}$ denotes the distance of joint J from the attached joint K and vector ${}^{\text{J}}\mathbf{r}_0^{\mathsf{T}}$ represents the initial position of joint J with respect to its local coordinate frame. The radii is not relevant for simulation as it will be used only for visualizing the simulated motion of human body. Note that the trunk is assumed to be rigid. Thus joints R, LH, and RH are set as static points of reference for the other dynamic joints. This also implies that joints B, N, LS and RS of the same person move as one with the same $\alpha_{t_{\text{next}}}$, $\beta_{t_{\text{next}}}$ and $\gamma_{t_{\text{next}}}$. The other joints move independently.

The simulator consists of two parts that work independently and concurrently, both for stochastic and deterministic motions. The simulators for both motion types work similarly, each for one of the two parts of the body. For stochastic motion they are detailed in Algorithms A.1 and A.2. In line 5 of Algorithm A.2, Equation (A.3) is used where the indices 1 and 2 refer to the indices *last* and *next*. Equation (A.1) is used in line 6 of Algorithm A.2 where $\mathbf{r}_0[p,j] = \mathbf{r}_{t_{\text{last}}}[p,j]$. Line 7 in Algorithm A.2 is especially important when a joint is a child. If the joint is connected to another joint as a parent which also may have a parent, then they all must be taken into account in the transformation. For example, LW is a child of LE and consequently LE is a child of LS. Analogous to Equation (A.4), ${}^{G}\mathbf{r}_F$ is given by

$$ {}^{G}\mathbf{r}_F = {}^{G}R_{\text{LS}}{}^{\text{LS}}R_{\text{LE}}{}^{\text{LE}}R_{\text{LW}}{}^{\text{LW}}\mathbf{r}_0 + {}^{G}\mathbf{r}_F^{\text{LE}} \tag{A.6} $$

where the last part incorporates the necessary translation to make the left hand attached to LE. Therefore the correct parent-child hierarchy must be followed in Algorithm A.2 to yield the correct results. For example, LS comes before LE and LW follows LE. Algorithms A.3 and A.4 detail the simulation for gait cycles using LI and FFS.

Using ${}^{G}\mathbf{r}_{t_F}$ produced by the simulation for each joint, the motion of human body can be visualized in 3D space from arbitrary view defined by azimuth and elevation angles that model camera motion. Spheres and cylinders in the humanoid model in Figure A.3 are followed with radius ${}^{\text{J}}r$ and distance ${}^{\text{K}}d_{\text{J}}$, the cylinder length for joint J. If

Algorithm A.1 Simulate the kinematics of each independent joint$[p, j]$ with stochastic motion

1: $\alpha_{last}[p, j] \leftarrow 0$
2: $\beta_{last}[p, j] \leftarrow 0$
3: $\gamma_{last}[p, j] \leftarrow 0$
4: **while** $t \leq T_{sim}$ **do**
5: sample $T_{next}[p, j]$ from n.e.d with mean $\lambda_{next}[p, j]$
6: sample $\alpha_{next}[p, j]$, $\beta_{next}[p, j]$ and $\gamma_{next}[p, j]$
7: $t_{last}[p, j] \leftarrow t$,
8: $t_{next}[p, j] \leftarrow t + T_{next}[p, j]$
9: hold for $T_{next}[p, j]$
10: $\alpha_{t_{last}}[p, j] \leftarrow \alpha_{t_{next}}[p, j]$
11: $\beta_{t_{last}}[p, j] \leftarrow \beta_{t_{next}}[p, j]$
12: $\gamma_{t_{last}}[p, j] \leftarrow \gamma_{t_{next}}[p, j]$
13: sample $T_{set}[p, j]$ from n.e.d with mean $\lambda_{set}[p, j]$
14: hold for $T_{set}[p, j]$
15: $\mathbf{r}_0[p, j] \leftarrow {}^{G}\mathbf{r}_t[p, j]$ from line 7 in Algorithm A.2
16: **end while**

a joint J has a sphere, it is created and displayed at the output points from simulation ${}^{G}r_{F}^{\mathrm{J}}$ as the center. Since the simulator works based on the normal ROM for humans, one can save all the time-consuming and expensive effort to hire people to perform body motions, record and segment the bodies, and produced the input traffic for DES. Experimental results in will later demonstrate this huge benefit. However, prior to that, we need to explain some important aspects for the second objective, as the contents from Section A.3 until now are devoted for addressing the first.

A.6 Silhouette areas of visualized moving human bodies for transient-traffic synthesis

Generally input traffic for DES is modelled according to the underlying probabilistic mechanism of the simulated system. For instance, the time between two consecutive events in a Poisson process follows n.e.d. It makes the distribution used often to model inter-arrival and service times in DES, as they are typically present in queue-based systems. A taxonomy of various input models for simulation can be found, for example, in [Law and Kelton (2000)]. The use of a certain distribution to generate input traffic for DES assumes that the statistical characteristics do not change over time, meaning the process is stationary. However, time and event processes in the real world are never always stationary. Transient periods are always present in such a process, for example at the beginning before it reaches the steady state.

This paper, with the second objective, extends previous work on transient traffic synthesis in [Rønningen et al. (2010); Rønningen (2011, 2007)]. In addition to manual inspection of some movies, step-rate generators are employed using four models of traffic sources with traffic slopes. Merging a number of streams from each model at random returns highly transient traffic.

Table A.2: Summary of all simulation parameters of stochastic and deterministic human motion using DEMOS. The top group of rows include the general parameters, while those relevant only for stochastic and deterministic motion are listed in the middle and bottom groups, respectively.

Parameter	Notation	Description
Simulation time	T_{sim}	The duration of the simulation as the chosen criterion for termination
Seed number	M	A unique M always yield unique and repeatable results
Frame rate	F	Frames per second to simulate the recording speed of the camera in a CS
Number of persons	P	Independently simulated persons, each denoted as person$[p]$, $p = 1, 2, ..., P$
Number of joints	J	Simulated joints per person, each denoted as joint$[p, j]$, $j = 1, 2, ..., J$, $J = 16$
Initial joint position	$\mathbf{r}_0[p, j]$	A vector denoting the initial position of each joint$[p, j]$, as detailed in Table A.3
Length of limb	d	Length of the end limb or limb connected by two consecutive joints, see Table A.3
Mean for $T_{next}[p, j]$	λ_{next}	Mean of n.e.d to sample $T_{next}[p, j]$ for each joint$[p, j]$
Mean for $T_{set}[p, j]$	λ_{set}	Mean of n.e.d to sample $T_{set}[p, j]$ for each joint$[p, j]$
ROM for $\alpha_{next}[p, j]$	$\alpha_{min}, \alpha_{max}$	Range of uniform distribution to sample $\alpha_{next}[p, j]$ for each joint$[p, j]$
ROM for $\beta_{next}[p, j]$	β_{min}, β_{max}	Range of uniform distribution to sample $\beta_{next}[p, j]$ for each joint$[p, j]$
ROM for $\gamma_{next}[p, j]$	$\gamma_{min}, \gamma_{max}$	Range of uniform distribution to sample $\gamma_{next}[p, j]$ for each joint$[p, j]$
Gait type	g	Walking ($g = 1$) or running ($g = 2$)
Cycle period	C_g	The periods of gait cycle for walking and running are recommended in Section A.4
Knee parameters	$\theta_g^K, a_{g,i}^K, b_{g,i}^K$	Parameters of the FFS in Equation (A.5) for knee, $i = 1, ..., 4$
Ankle parameters	$\theta_g^A, a_{g,i}^A, b_{g,i}^A$	Parameters of the FFS in Equation (A.5) for ankle, $i = 1, ..., 4$
ROM$_f$ for gait cycle	$\gamma_{f,g}[p, j]$	A set of γ angles for *forward* motion in half of gait cycle using LI
ROM$_b$ for gait cycle	$\gamma_{b,g}[p, j]$	A set of γ angles for *backward* motion in half of gait cycle using LI

Table A.3: The radii, lengths and initial positions of the links indicated in the first column for simulation and visualization. O, r and d in the last row refers to $[0,0,0]$, ${}^{\mathrm{R}}d_{\mathrm{B}}$, and the corresponding ${}^{\mathrm{K}}d_{\mathrm{J}}$, respectively.

J	R	B	N	H	LH	RH	LS	RS
K	-	R	B	N	R	R	B	B
${}^{\mathrm{K}}d_{\mathrm{J}}$	-	15	5	4	2		5	
${}^{\mathrm{J}}r$	5		-	4	3		2	
${}^{\mathrm{J}}\mathbf{r}_0^{\mathsf{T}}$	O	$[0,0,d]$			$[-d,0,0]$	$[d,0,0]$	$[-d,0,r]$	$[d,0,r]$

J	LE	RE	LW	RW	LK	RK	LA	RA
K	LS	RS	LE	RE	LH	RH	LK	RK
${}^{\mathrm{K}}d_{\mathrm{J}}$	10				15			
${}^{\mathrm{J}}r$	2		1		2		1	
${}^{\mathrm{J}}\mathbf{r}_0^{\mathsf{T}}$	$[0,0,-d]$							

Algorithm A.2 Output the position ${}^{G}\mathbf{r}_t^{(p,j)}$ every $1/F$ for independent joint$[p,j]$ with stochastic motion

while $t \leq T_{sim}$ **do**
  hold for $1/F$
  for every person$[p]$ **do**
    for every joint$[p,j]$ with stochastic motion under the correct parent-child hierarchy **do**
      compute $\alpha_t[p,j]$, $\beta_t[p,j]$, $\gamma_t[p,j]$ (Equation (A.3))
      compute ${}^{A[p,j]}\mathbf{r}_t^{(p,j)}$ using Equation (A.1)
      compute ${}^{G}\mathbf{r}_t[p,j]$ using Equation (A.4)
      append ${}^{G}\mathbf{r}_t[p,j]$ in the output files
    end for
  end for
end while

However they do not address the traffic from a CS that essentially centers on the motion of human bodies. Apart from these work, to our best knowledge, the study on transient traffic synthesis for DES is still in infancy, particularly with respect to our goals. Achieving the second objective will reduce that void in the field of DES.

The transient traffic synthesized from arbitrary simulated collaboration scenario will be in a curve of bitrate against time. Therefore we have to investigate how to produce the bitrate from the visualization of the simulated human motion. The time axis is perfect because the positions are recorded according to frame rate F which essentially models the camera recording in a CS. This visualization every $1/F$ means intraframe processing of visual data. It fits our simulation purpose because delay-sensitive collaborations prefer avoiding interframe compression of video signals since it incurs additional delay.

Algorithm A.3 Simulate the human gait cycles for the relevant joints$[p, j]$ for LI or FFS

$\alpha_{last}[p, j] \leftarrow 0$
$\beta_{last}[p, j] \leftarrow 0$
$\gamma_{last}[p, j] \leftarrow 0$
while $t \leq T_{sim}$ **do**
    $\gamma[p, j] \leftarrow \gamma_{g,1}[p, j]$
    $t_{last} \leftarrow t$
    hold for $C_g/2$
    $\gamma_{last}[p, j] \leftarrow \gamma[p, j]$
    $\gamma[p, j] \leftarrow \gamma_{g,2}[p, j]$
    hold for $C_g/2$
    $\gamma_{last}[p, j] \leftarrow \gamma[p, j]$
    $\mathbf{r}_0[p, j] \leftarrow {}^{G}\mathbf{r}_t[p, j]$ from Algorithm A.4
end while

Algorithm A.4 Output ${}^{G}\mathbf{r}_t^{(p,j)}$ every $1/F$ for relevant joint$[p, j]$ with gait cycles using LI and FFS

while $t \leq T_{sim}$ **do**
    hold for $1/F$
    for every person$[p]$ **do**
        $x \leftarrow (t - t_{last})/C_g \times 100$
        compute γ_g^{LK} and γ_g^{RK} using Equation (A.5)
        if $x > 50$ **then**
            $x \leftarrow x - 50$
            compute γ_g^{LA} and γ_g^{RA} using Equation (A.5)
            $\gamma_g^{LA} \leftarrow -\gamma_g^{LA}$
            $\gamma_g^{RA} \leftarrow -\gamma_g^{RA}$
        end if
        $\gamma_{g,LI} \leftarrow x(\gamma[p, j] - \gamma_0[p, j])/50$ from Equation (A.3)
        compute ${}^{A[p,j]}\mathbf{r}_{t,LI}^{(p,j)}$ and ${}^{A[p,j]}\mathbf{r}_{t,FFS}^{(p,j)}$ using Equation (A.1)
        compute ${}^{G}\mathbf{r}_{t,LI}^{(p,j)}$ and ${}^{G}\mathbf{r}_{t,FFS}^{(p,j)}$ using Equation (A.4)
        append ${}^{G}\mathbf{r}_{t,LI}^{(p,j)}$ and ${}^{G}\mathbf{r}_{t,FFS}^{(p,j)}$ in the output files
    end for
end while

Thus in this paper a video data from a CS is modelled as a sequence of n frames that are processed as independent images of $W \times H$-pixel resolution.

As explained in Section F.1, here we work with images that render a segmented object with a background of uniform color. A frame image I is called image B if it contain only the background of uniform color. Figure A.8 (a) and (b) show two frames of one person whose left hand and head are in two different positions. However they both produce the same silhouette as in Figure A.8 (d) because her body occludes her left hand and the visual differences of the faces in the two images are not relevant. It yields the rest of the image as background as in Figure A.8 (e). The last two pictures depict how occlusions can be more elaborate when more persons are involved.

Let S_B, S_n^I, S_n^O and S_n^B denote the size in bits of image B, frame n, and frame n

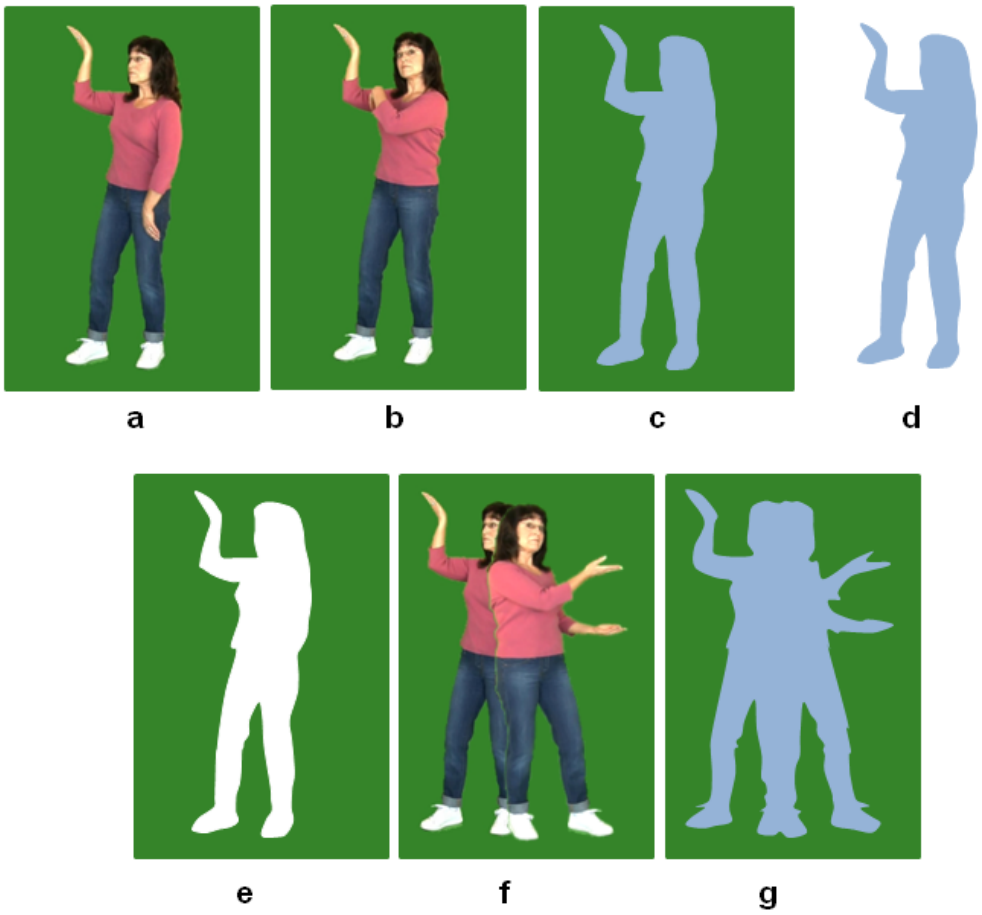
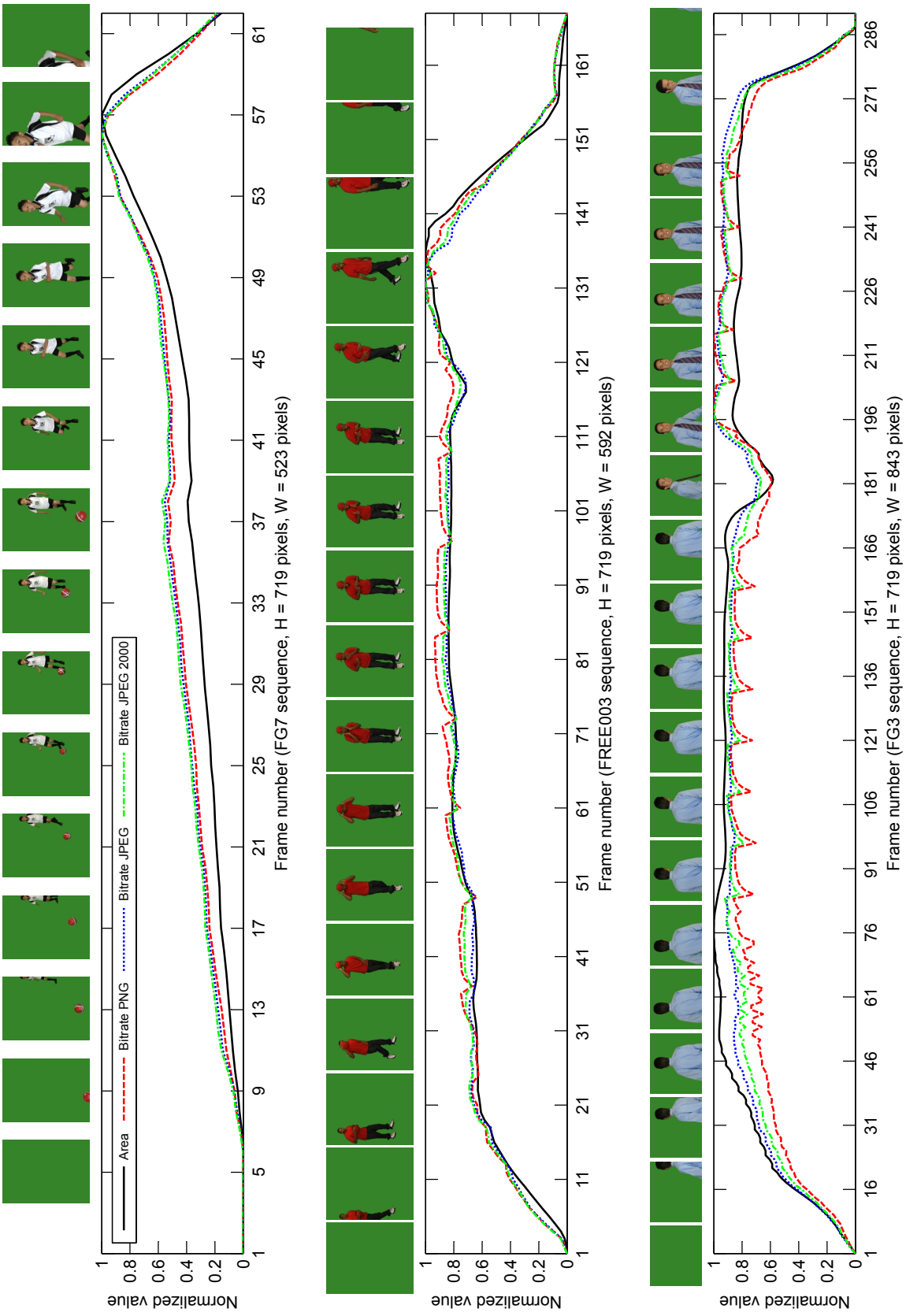


Figure A.8: Human body with background of uniform color from background subtraction with its silhouettes as frames from a video sequence. Two different frames of one person with different positions of the hands and head in (a) and (b) but with the same silhouette (c) due to occlusions. Such frame always comprises the area contributed by the silhouette of the object, i.e. the body of the person (d), and that remaining from the background (e). Two persons with occlusions make it more complicated (f,g).

contributed by the object O and the background B therein, respectively. The same notation applies for R_n^I and R_n^O where R represents the image bitrate in bits per color pixel (bpp). Figure A.8 (c) demonstrates that $S_n^I = S_n^O + S_n^B = S_n^O + (1 - a_n)\, S_B$ where a_n is the ratio of the image area in pixels occupied by the object's silhouette in frame n (A_n^O) and the total area of the image (HW). As the system exchanges only the data from the object, it can be computed simply as $S_n^O = S_n^I - (1 - a_n)\, S_B$ and all factors in the right-hand side are known from measurement. It translates into bitrate as $R_n^O = S_n^O / A_n^O$.

We investigate the relationship between A_n^O and R_n^O by experimenting with green-screen color video sequences freely offered at [TGFX (2012)]. The resolution of the sequences is 1280×720 pixels. Every sequence includes frame B to compute S_B. Figure A.9 plots a_n and $\hat{R}_n^O$ from five exemplary sequences over the frame number where $\hat{R}_n^O = \Delta\hat{R}_n^O / \max(\Delta\hat{R}_n^O)$. It normalizes R_n^O to be between 0 and 1 where $\Delta\hat{R}_n^O = R_n^O - R_{n,\min}^O$. PNG, JPEG and JPEG 2000 compressions with highest quality are employed.


Area
Bitrate PNG
Bitrate JPEG
Bitrate JPEG 2000
Normalized value
Frame number (FG7 sequence, H = 719 pixels, W = 523 pixels)
Frame number (FREE003 sequence, H = 719 pixels, W = 592 pixels)
Frame number (FG3 sequence, H = 719 pixels, W = 843 pixels)

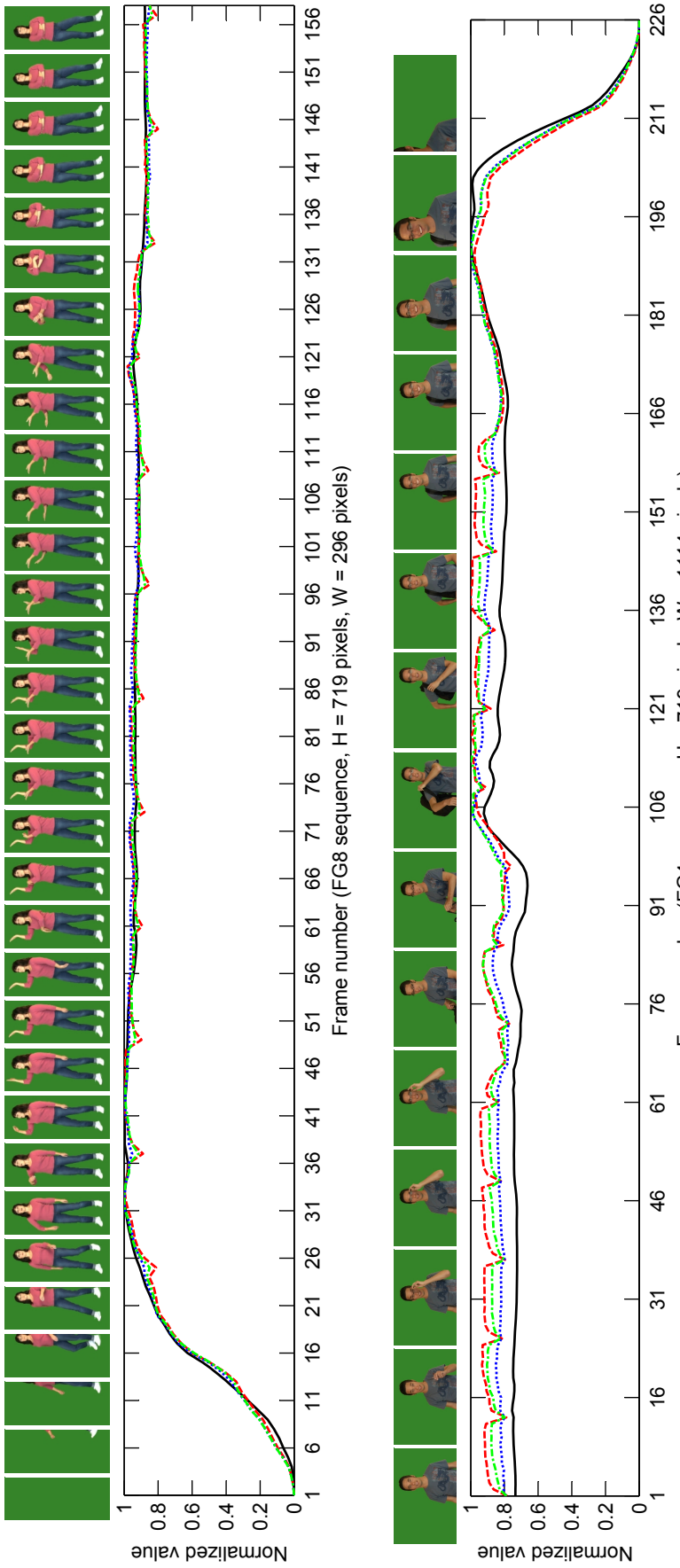


Figure A.9: The normalized silhouette areas and bitrates from PNG, JPEG and JPEG 2000 for FG7, FREE003, FG3, FG8 and FG4 sequences cropped and converted from TGFX (2012) (top to bottom). All sequences are originally in 1280 × 720-pixel resolution. The frame numbers shown are accompanied by the respective frame snapshots for comparison and evaluation by readers. Images are to be seen on screen for best VQ.

It is clear that $\hat{R}_n^O$ faithfully follows a_n. Therefore the synthesized a_n from visualizing the simulated human motion is a good approximation to synthesize the wanted trace of input traffic $\bar{S}_n^O$ in bits for DES that is expressed by

$$\bar{S}_n^O = C\,\bar{R}^O\,\bar{a}_n\bar{W}\bar{H} \tag{A.7}$$

where $\bar{a}_n$ is the normalized silhouette area of the simulated human bodies in frame n and C denotes the percentage of the targeted resolution $\bar{W} \times \bar{H}$ covered by the object with the maximum $\bar{a}_n$ from the corresponding surface. $\bar{R}$, $\bar{H}$ and $\bar{W}$ are scalar values selected by users to take compression and image resolution into account. $\bar{R}$ can be set as a scalar since R_n^O from measurement is generally stationary. The five sequences have R_n^O between 4 and 15 bpp on average.

Differences exist between $\hat{R}_n^O$ and a_n. It is mainly caused by the more bits allocated to areas with high-frequency information. For instance, the human face and the texture of the tie in the FG3 sequence contribute to the accrual in bitrate after frame 181. These differences and the causing factors are interesting for future exploration. Nevertheless, as the shape of a_n is very similar or close to that of $\hat{R}_n^O$, the silhouette area is the most dominant factor which makes is the core of our simulation. Together with the occlusions, it offers a major advantage for simplification in traffic synthesis: it nullifies the need to produce strictly faithful imitation of limb motion in simulation, such as the complex emulation of ballet dances in [LaViers et al. (2011a); LaViers and Egerstedt (2011); LaViers et al. (2011b)]. It explains why less realistic results from forward kinematics are tolerable in this work in favor of its simplicity, speed and fitness to DES.

A.7 Simulation results and discussion

Here we exhibit and discuss exemplary simulation results both for stochastic and deterministic motions. Figure A.10 depicts the normalized silhouette areas from the first 300 frames synthesized with $F = 30$ fps, $\lambda_{\text{next}} = 250$ms, $\lambda_{\text{set}} = 150$ms, and $M = 1945$. They follow the views corresponding to the XZF and YZL surfaces in Figure A.2 (a).

The area computed in each plot is normalized to be within 0 and 1 according to the corresponding maximum value. The transient traffic results from scaling the normalized silhouette areas with a scalar value that takes important factors into account, such as image resolutions and the compression ratio of the image-compression technique used. On top of frame number 1, 10, 20, ..., 300 for each plot in Figure A.10, the corresponding snapshot of the simulated human-body motion is provided. Readers are invited to validate the results in three ways. First, whether the resulting positions of human body fall within the ROM by inspecting the sequences of human-body motion. Second, whether motion sequences are smooth and streamlined. Third, the resulting normalized silhouette areas are correct by comparing the values in the plot and the body positions above them.

The normalized silhouette areas for gait cycles are depicted in Figure A.11. The simulations with three different views are conducted with $F = 30$ fps where the body and the head are static. Each plot covers two simulation methods for comparison. The view from XZF surface is denoted by 0° and -90° of elevation and azimuth angles.

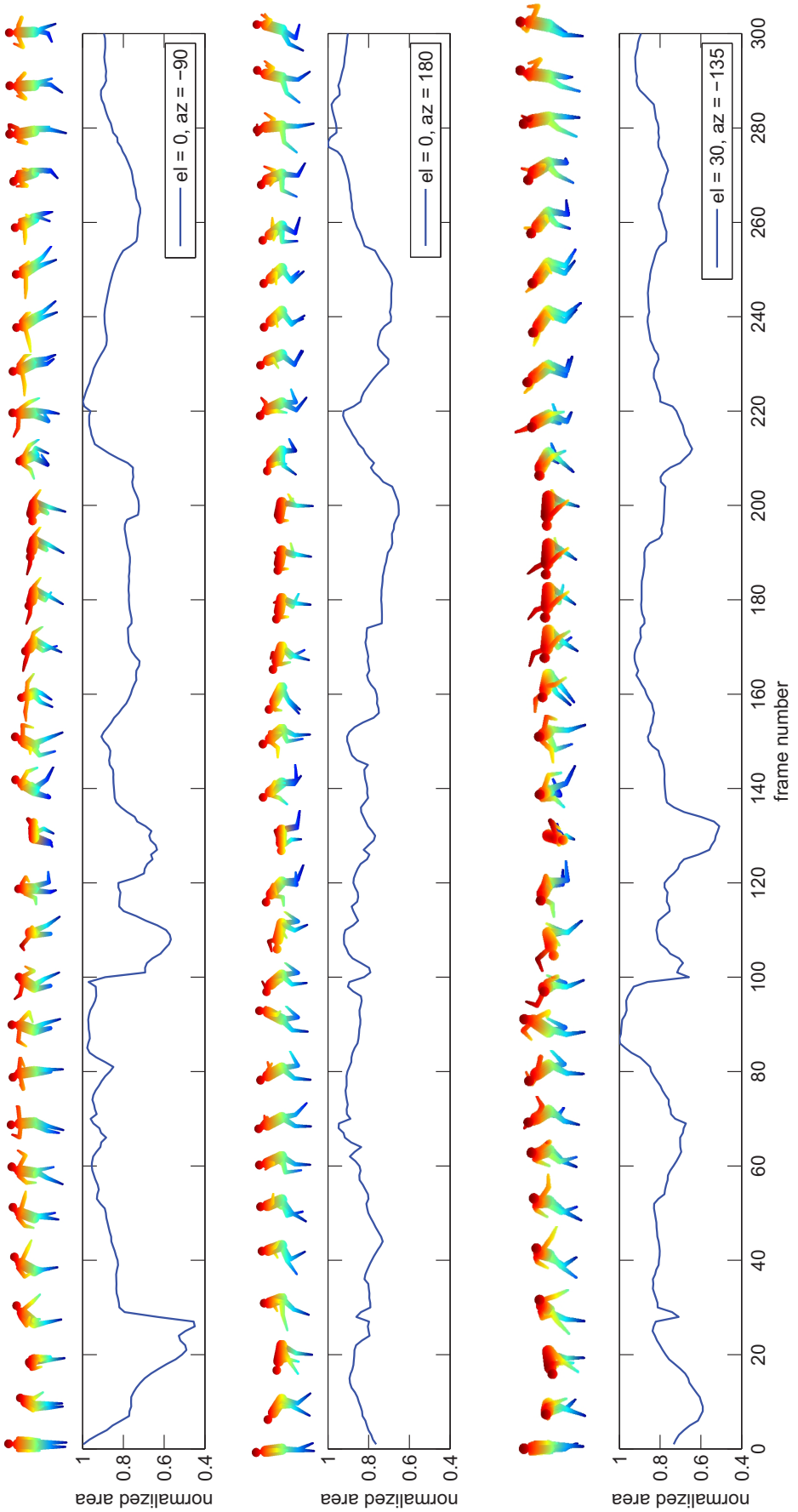


Figure A.10: The normalized silhouette area as the final output of the simulation with three different views. Each plot is accompanied with the snapshots of the simulated human-body motion for frame number 1, 10, 20, ..., 300. The input parameters are F =30 fps, λ_{next} =250ms, λ_{set} =150ms, and M =1945.

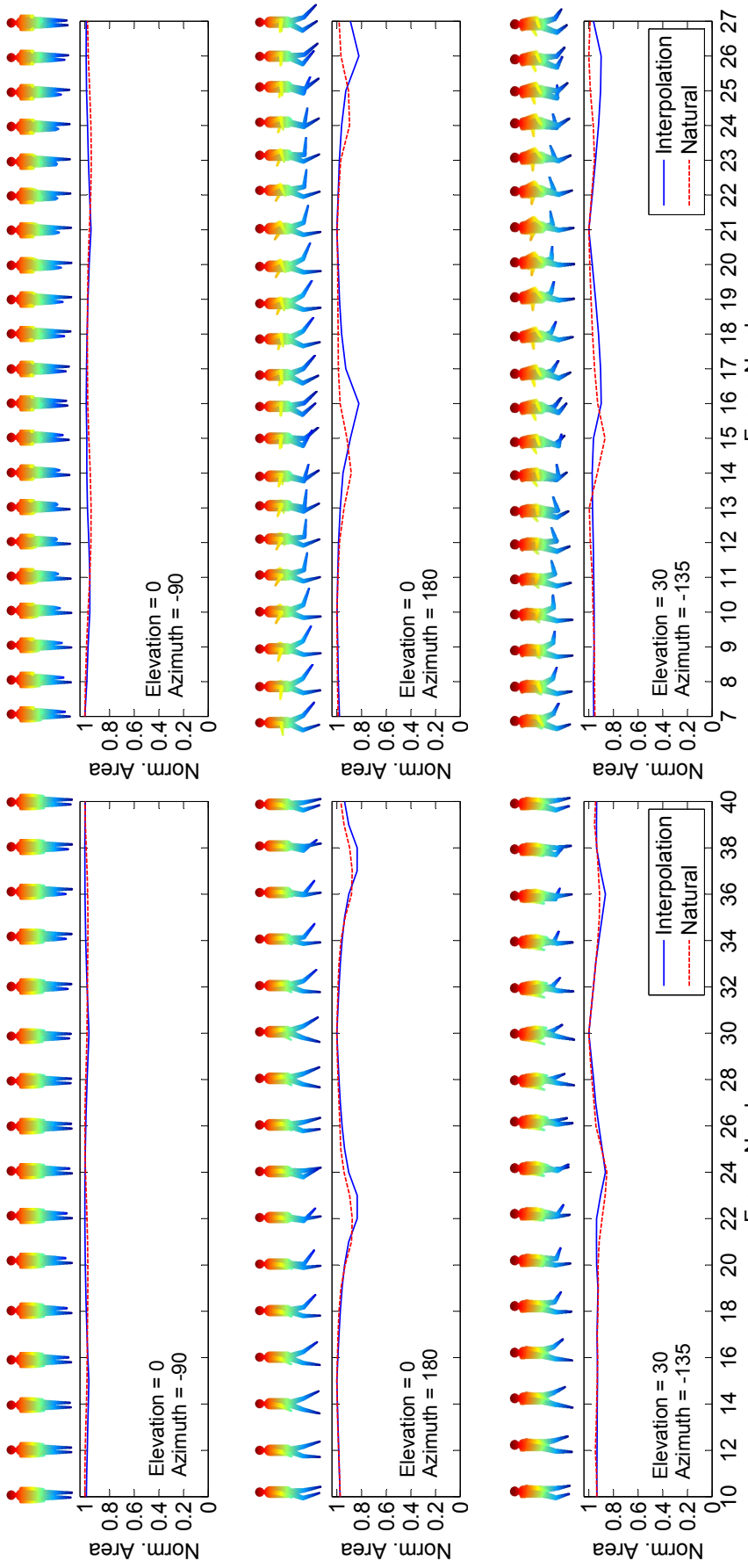


Figure A.11: The normalized silhouette area from a gait cycle for walking (left) and running (right) as the output of the simulation with three different views and $F = 30$ fps. Two simulation methods are conducted: natural cycle using FS and LI of two alternating sets of ROMs for the lower limbs. Each plot is accompanied with the snapshots of the body motion from the first method. Hand swings are also included in the simulation.

Analogously 0° and 180° define the view from YZL surface. First, the gait is simulated using the FFS in Equation (A.2) with the parameter values in Figure A.7. Second, the gait cycles using the model are approximated by LI in Equation (A.3). The two sets of limbs mentioned above moves in alternating manner according to the two sets of ROMs presented in Table A.1. The transition from forward to backward and vice versa is governed by LI. Note that only the ROM for γ are relevant, i.e. the rotation around the X axis. Readers can use them to validate the correctness of the results. It is obvious from Figure A.11 that LI well approximates the FFS model.

A.8 Exemplary application

In this section we revisit the reference system with the collaboration scenario pictured in Figure A.2 (a). We simulate the motion of the singer and the dancer in CS1 and CS2 with $F = 30$ fps for sufficient exhibition of the motions within the allowed space for this paper. The visualization produces the normalized silhouette areas that are scaled to synthesize the wanted transient traffic. Simulating the delay-guaranteeing network architecture needs input traffic from YZL surface in CS1 and YZR surface in CS2. The XZF surfaces in CS1 and CS2 become the source of input traffic for simulating the network architecture without delay guarantee.

A.8.1 Simulating and visualizing the reference scenario

A transient period in the input traffic mainly occurs during the occlusions between the singer and the dancer. The fluctuations in the traffic will be due to the motion of their limbs. To incur frequent transient traffic from the YZL and YZR surfaces for display on this paper, the dancers are allowed to be only one step forward or backward from their initial position. This gives three possible positions for the dancers. When they are at their initial position, they walk one step forward or backward with probability 0.5. The motion of one-step walking is simulated using LI with the ROM in Table A.1. LI is more flexible than FFS to model and simulate such motion.

We assume the gait period to follow n.e.d with mean $\lambda_{\text{period}} = 500$ms. It is forced as the period if the generated random period is less than 500ms. A dancer stays at a position for a duration that also follows n.e.d with mean $\lambda_{\text{stay}} = 1000$ms. As for the upper-half limbs, the time for the next move and the duration to stay at a position are also assumed to follow n.e.d with the mean $\lambda_{\text{next}} = \lambda_{\text{now}} = 250$ms. As clear from the ROM presented in Table A.4, the singers move within a much more limited range than those of the dancers in this collaboration scenario.

For correct visualization, the coordinates of the simulated limbs must first be compensated along the X, Y and Z axes. The dancer and singer in a CS are initially at the same position but separated by distance $d = 50$ along the X axis, based on the lengths and the initial positions of the arm and the hand in Table A.3. Their bodies are always oriented to face the XZF surface. All the coordinates along the X axis must be modified with the distance d, depending on the position of the singer and the dancer.

To visualize the forward or backward motion, the coordinates along the Y axis for all the limbs of the dancers must be compensated accordingly. Finally, to enable

Table A.4: The minimum and maximum ranges of α, β and γ in degrees for the singers (S) and the dancers (D).

Joints	α_S	β_S	γ_S	α_D	β_D	γ_D
N, B, RS, LS	$-30,30$	0	0	$-30,30$	$-40,40$	$-90,35$
H	0	0	0	0	0	0
RE	0	$-45,0$	$0,45$	0	$-170,40$	$40,170$
LE	0	$0,45$	$0,45$	0	$-40,170$	$40,170$
RW, LW	0	0	$0,150$	0	0	$0,150$
RK, LK	0	0	0	0	0	$-20,20$
RA, LA	0	0	0	0	0	$-5,15$

natural look of one-step walking, the coordinates along the Z axis must be updated so that the body is always in contact with the ground. The wanted surface, or the active camera array, is determined by specifying the azimuth and the elevation angles during visualization. The effects from perspectives with respect to the position and distances are not considered in this work.

After correct visualization, the normalized silhouette areas are produced from the two addressed surfaces in CS1 as shown in Figure A.12 and CS2 as in Figure A.13. The seed numbers used are 356, 1980, 281, and 1945, for S1, D1, S2, and D2, respectively. The areas are normalized using the maximum areas from the corresponding surface in CS1 and CS2. Snapshots of the frames from three different views are on top of the plots. The views from top to bottom are those from the YZL/YZR surface, the XZF surface, and 45° between the XZF and YZL surfaces. The snapshots from the first view are given every ten frames. As expected, one can see transient periods in the normalized area from the YZL and YZR surfaces whereas the normalized area from the XZF surface is relatively much more stable. Here we assume that the singers sing and look at each other, except from frame 60 to 180, indicated by immediate zero traffic. Thus eye contact is another cause of transient periods with very steep transition.

Frame rate F is a very important parameter. Standard movies are recorded with F = 24 fps, but it becomes insufficient for larger displays and increased pictured resolution due to the loss of detail on fast moving objects on 50/60Hz displays. Armstrong et al. (2008) asserted that high-frame rate capture and display must be seriously considered to achieve significant improvements in VQ with the increasing spatial resolutions of proposed television standards. They also recommended F = 300 fps for easy down-conversion to dynamic resolutions such as 50, 60, or 100 Hz. The near-natural quality requires high-frame rate cameras and displays.

A.8.2 Scaling normalized silhouette areas into synthetic traces of input traffic

The normalized silhouette areas in Figures A.12 and A.13 are transformed into synthetic traces of input traffic for DES using Equation (A.7). The values of the three parameters involved must be defined.

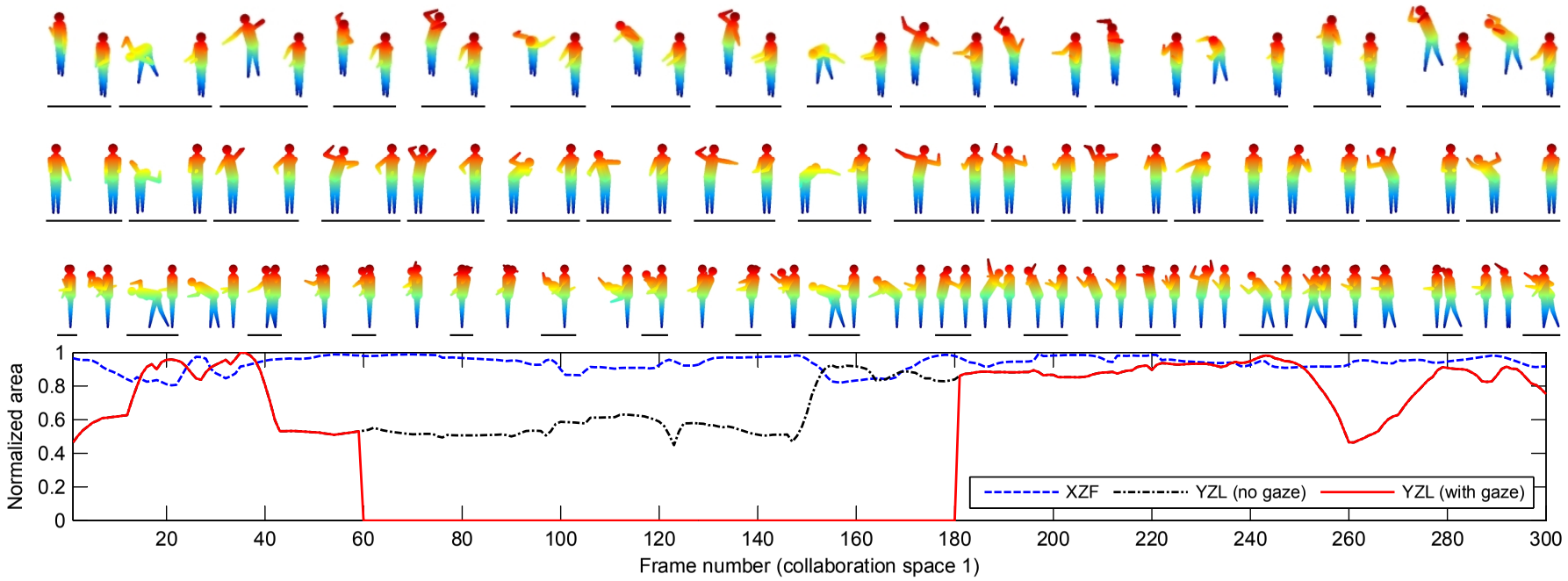


Figure A.12: The normalized area from YZL and XZF surfaces of CS1 with $M = 356$ for S1 and $M = 1980$ for D1. Frame snapshots in three rows from top to bottom are those from the YZL surface, the XZF surface, and 45° between the XZF and YZL surfaces, respectively.

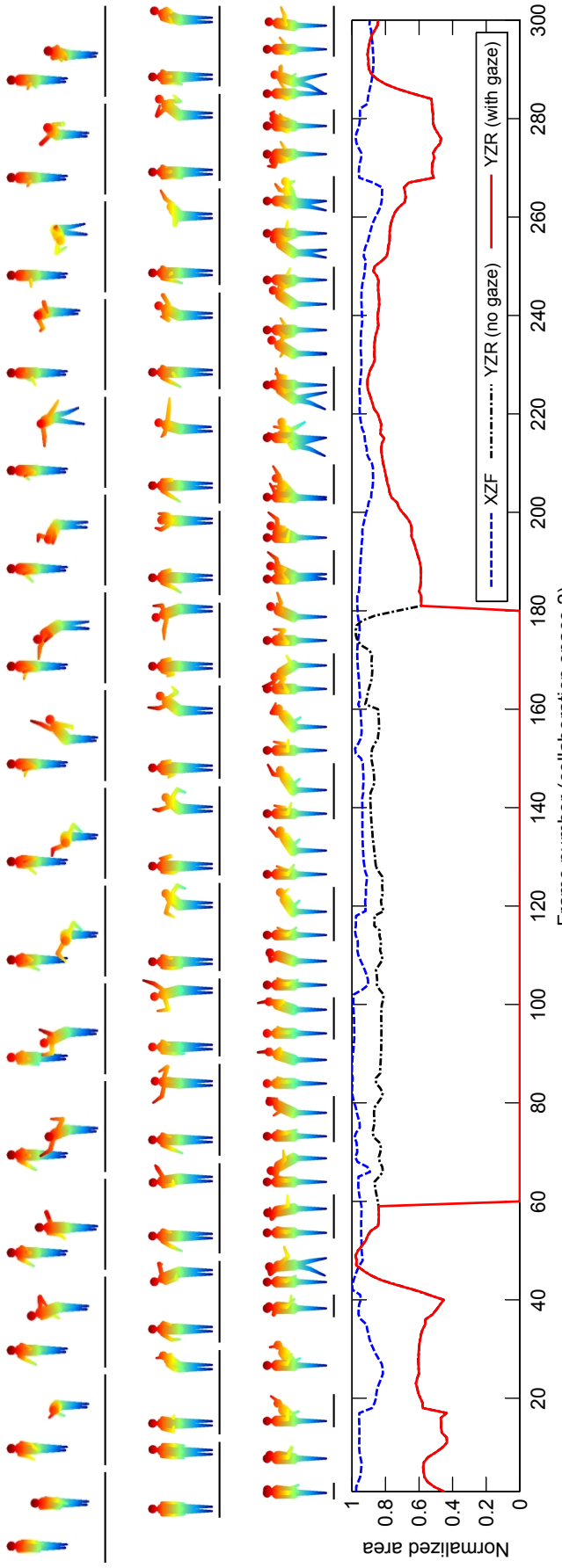


Figure A.13: The normalized area from YZL and XZF surfaces of CS1 with $M = 281$ for S2 and $M = 1945$ for D2. Frame snapshots in three rows from top to bottom are those from the YZR surface, the XZF surface, and 45° between the XZF and YZL surfaces, respectively.

DMP architecture accommodates either uncompressed or compressed data as the input traffic. $\bar{R}^O$ equals 24 bpp for an uncompressed RGB color image. Suitable compression schemes for DMP is still under investigation. However, if the information of the color image is partly dropped prior to entering the network, than $\bar{R}^O$ also drops slightly. The second parameter to specify is the targeted resolution $\bar{W} \times \bar{H}$. Then the bitrate $\bar{R}$, in bits per second (bps) or Bytes per second (Bps), of the synthesized sequence with N frames can be computed from $\bar{R} = \frac{F}{N} \sum_{n=1}^{N} \bar{S}_n^O = \bar{R}_{\max}^O \sum_{n=1}^{N} \bar{a}_n$ where $\bar{S}_n^O$ comes from Equation (A.7) and $\bar{R}_{\max}^O = F\,C\,\bar{R}^O\,\bar{W}\,\bar{H}/N$.

To display object with life-size dimension, the size of YZL or YZR surface must be sufficient for two persons with the allowed ROM. If 60-inch display panels are tiled on the surface where each panel is 52.29in × 29.42in (1.33m × 0.75m), both surfaces must be designed as at least two columns and three rows of such display panels, as illustrated in Figure A.14 (a). It covers a total area of 2.66 × 2.25 m^2. Given the 1920 × 1080-pixel resolution with 16:9 aspect ratio per panel, the total resolution of the surface is $\bar{W} = 3840$ and $\bar{H} = 3240$ pixels.

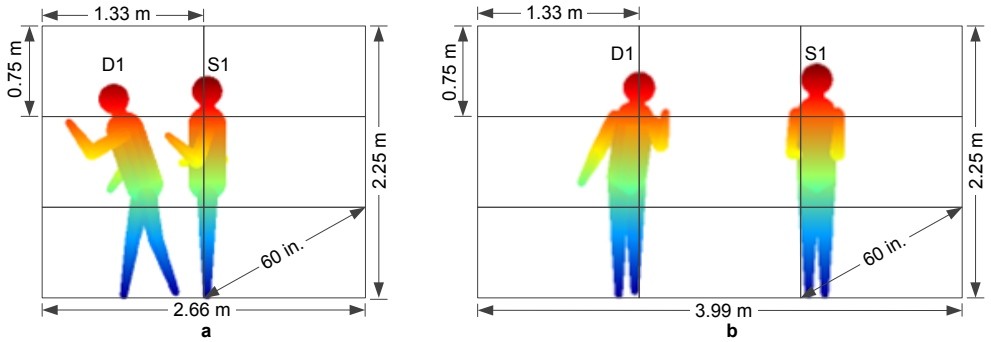


Figure A.14: Configurations of 60-inch display panels for YZL (a) and XZF (b) surfaces in CS1.

Assume that the largest silhouette area from both surfaces in Figures A.12 and A.13 occupies 20% of the resolution ($C = 0.2$) with $\bar{R}^O = 20$ bpp. Therefore, if both CS maintain exchanging data despite no eye contact from S1 or S2, then $\bar{R}$ from the 300-frame traffic from these surfaces are 1.015 Gbps (129.9 MBps) from CS1 and 1.07 Gbps (137.04 MBps) from CS2, respectively. For both CS, $\bar{R}_{\max}^O = 4.746$ Mbps. These rates will be much higher when the sense of depth is enabled with 3D display panels. One can see how a collaboration scenario affects the source and magnitude of the input traffic. For example, changing the scenario so that the singers and dancers face and interact with each other via XZF surfaces will increase the input traffic. It is because, as shown in Figure A.14 (b), the XZF surface employs at least one additional column of screens and the silhouette area is larger than those from the YZL and YZR surfaces.

The input traffic for the network architecture without delay guarantee is much lower because the aggregator is only one television. Assuming it is one of the display panels above, each CS contributes to half of the resolution, i.e. 960 × 1080 pixels. Let us assume that the largest silhouette area from both XZF surfaces in CS1 and CS2 takes 20% of the resolution. By choosing $\bar{R} = 5$ bpp, the traffic results from scaling the normalized silhouette area in Figures A.12 and A.13 with $\bar{R}_{\max}^O = 101.25$ kbps. The video bitrate for the traffic is 27.78 Mbps (3.47 MBps) from CS1 and CS2 due to no occlusion between the

pair of singer and dancer. The aggregator must inform CS1 and CS2 about its technical specifications including the resolution. This set of information is called *scene profile* and supported in the DMP architecture. All connected CS exchange it with each other prior to transmitting any data, as shown in Figure A.2 (a).

A.9 Conclusions and future outlook

We proposed a comprehensive simulation framework for modeling and simulating arbitrary complex collaboration scenario in the envisioned system. Instances of moving human body are independently animated in it via forward kinematics based on a model as discrete event system. The simulation was implemented in excellent object-oriented DEMOS tool. Besides stochastic human motion using the natural *range of motion* for human, we also covered deterministic motion in the form of human gait cycles for walking and running. Linear interpolation and finite Fourier series FFS were used in simulating the forward kinematics. LI not only matches the humanoid model and enables fast, but also closely approximates the natural functions in FFS. Moreover it is more flexible than FFS for gait cycles and hand swing with arbitrary ROM. We demonstrated how the simulator serves as the building block in modeling, simulating and visualizing arbitrary scenario of multi-actor interaction in the envisioned system. Thus the properties and performance of such complex system can be studied using the simulation framework despite its nonexistence.

We also showed how the simulator framework also functions as a novel transient-traffic generator for DES using the silhouette areas from the visualization of the simulated human motion. Abrupt changes in bitrates shown in the synthetic traces generated confirm the transient properties that are expected from the real traffic when the collaboration system is fully implemented. Moreover the periods in the synthetic traces where no packet is transmitted due to changes in eye gaze of the performer make the traces resemble the ON/OFF sources in [Willinger et al. (1997)]. There the i.i.d. and independent ON- and OFF-periods strictly alternate and do not need to have the same distribution. They reported that such an ON/OFF source exhibits the Noah Effect (i.e. have high variability or infinite variance) that will produce aggregate network traffic with self-similarity [Leland et al. (1994)]. This indicates the possibility that self-similarity will also characterize the real network from the envisioned collaboration. First steps to investigate it may include analyzing synthetic traces generated from our simulation by following the work of Garrett and Willinger (1994). The results of this future work will be valuable for constructing a traffic model that correctly synthesizes traces of transient traffic from moving human bodies for DES without visualization.

Using real-life ROM from literature in our simulation framework shows our utmost care in its accuracy and validity. However these need to be strengthened by comparing with real traffic and measurement from the envisioned collaboration system. Until the system can be physically constructed, methods to approximate it and the measurements for comparison and validation will be undertaken as future work. The accuracy and naturalness of the animated human motion can be improved by employing more complex animation methods in DES, for example starting with real-time reproducible inverse kinematics techniques [Tolani et al. (2000)]. Investigating other factors in addition to

silhouette area that affect the synthesis of the input traffic would be interesting, for example areas of high-frequency content such as human faces and textured regions. Other ideas include incorporating variable high-frame rates [Armstrong et al. (2008)] and exploitation of temporal redundancies in the traffic synthesis.

References

Aggarwal, J., Ryoo, M., 2011. Human activity analysis: a review. *ACM Computing Surveys* 43 (3), 1–43.

Armstrong, M., Flynn, D., Hammond, M., Jolly, S., Salmon, R., 2008. *High Frame-Rate Television*. BBC Research White Paper WHP 169, BBC.

Birtwistle, G., 2003. *DEMOS - A System for Discrete Event Modelling on Simula*. School of Computer Science, University of Sheffield.

Chafe, C., Gurevich, M., Leslie, G., Tyan, S., 2004. Effect of time delay on ensemble accuracy. In: *Proc. International Symposium on Musical Acoustics*.

Chang, C., Ansari, R., Khokhar, A., 2004. Cyclic articulated human motion tracking by sequential ancestral simulation. In: *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 45–52.

DeFanti, T., Acevedo, D., Ainsworth, R., Brown, M., Cutchin, S., Dawe, G., Doerr, K., Johnson, A., Knox, C., Kooima, R., Kuester, F., Leigh, J., Long, L., Otto, P., Petrovic, V., Ponto, K., Prudhomme, A., Rao, R., Renambot, L., Sandin, D., Schulze, J., Smarr, L., Srinivasan, M., Weber, P., Wickham, G., 2011. The future of the CAVE. *Central European Journal of Engineering* 1 (1), 16–37.

Faller, A., Schunke, M., Schunke, G., 2004. *The Human Body: An Introduction to Structure and Function*. Georg Thieme Verlag.

Franz, J., Paylo, K., Dicharry, J., Riley, P., Kerrigan, D., 2009. Changes in the coordination of hip and pelvis kinematics with mode of locomotion. *Gait & Posture* 29 (3), 494–498.

Fuo, C., Wang, M., 2012. Motion generation and virtual simulation in a digital environment. *International Journal of Production Research* 50 (22), 6519–6529.

Garrett, M., Willinger, W., 1994. Analysis, modeling and generation of self-similar VBR video traffic. *ACM SIGCOMM Computer Communication Review* 24 (4), 269–280.

Granieri, J., Crabtree, J., Badler, N., 1995. Production and playback of human figure motion for visual simulation. *ACM Transactions Modeling and Computer Simulation* 5 (3), 222–241.

Hatze, H., 1980. A mathematical model for the computational determination of parameter values of anthropomorphic segments. *Journal of Biomechanics* 13 (10), 833–843.

Ingalls, R., 2001. Introduction to simulation. In: *Proc. Winter Simulation Conference (WSC)*. pp. 7–16.

Iversen, V., 2007. *Teletraffic Engineering Handbook*. ITU.

Jazar, R., 2010. *Theory of Applied Robotics: Kinematics, Dynamics, and Control*. Springer.

Ji, X., Liu, H., 2010. Advances in view-invariant human motion analysis: a review. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews* 40 (1), 13–24.

LaViers, A., Chen, Y., Belta, C., Egerstedt, M., 2011a. Automatic sequencing of ballet poses. *IEEE Robotics and Automation Magazine* 18 (3), 87–95.

LaViers, A., Egerstedt, M., 2011. The ballet automaton: A formal model for human motions. In: *Proc. American Control Conference*. pp. 3837–3842.

LaViers, A., Egerstedt, M., Chen, Y., Belta, C., 2011b. Automatic generation of balletic motions. In: *Proc. IEEE/ACM International Conference on Cyber-Physical Systems*. pp. 13–21.

Law, A., Kelton, W., 2000. *Simulation Model and Analysis*. McGraw-Hill.

Leland, W., Taqqu, M., Willinger, W., Wilson, D., 1994. On the self-similar nature of Ethernet traffic (extended version). *IEEE/ACM Transactions on Networking* 2 (1), 1–15.

Maimone, A., Bidwell, J., Peng, K., Fuchs, H., 2012. Enhanced personal autostereoscopic telepresence system using commodity depth cameras. *Computers & Graphics* 36 (7), 791–807.

National Aeronautics and Space Administration (NASA), 1995. Anthropometry and biomechanics (Vol. 1, Section 3) in Man-Systems Integration Standards (NASA-STD-3000) Revision B. http://msis.jsc.nasa.gov/sections/section03.htm.

Novacheck, T., 1998. The biomechanics of running. *Gait & Posture* 7 (1), 77–95.

Ramamurthy, P., Blundell, M., Bastien, C., Zhang, Y., 2012. Computer simulation of real-world vehicle-pedestrian impacts. *International Journal of Crashworthiness* 16 (4), 351–363.

Rønningen, L. A., 2007. A protocol stack for futuristic multimedia. In: *Proc. International Conference On Signal Processing and Communication Systems*. pp. 1–9.

Rønningen, L. A., 2011. *The Distributed Multimedia Plays Architecture (version 3.20)*. Tech. rep., ITEM, NTNU.

Rønningen, L. A., Panggabean, M., Tamer, Ö., 2010. Toward futuristic near-natural collaborations on Distributed Multimedia Plays architecture. In: *Proc. IEEE International Symposium on Signal Proccessing and Information Technology (ISSPIT)*. pp. 102–107.

Rønningen, L. A., Wittner, O., 2011. *Experiments on remote conducting between Trondheim and Lisbon*, ITEM, NTNU.

TGFX, 2012. http://www.timelinegfx.com/.

Tolani, D., Goswami, A., Badler, N., 2000. Real-time inverse kinematics techniques for anthropomorphic limbs. *Graphical Models* 62 (5), 353–388.

Vasudevan, R., Kurillo, G., Lobaton, E., Bernardin, T., Kreylos, O., Bajcsy, R., Nahrstedt, K., 2011. High-quality visualization for geographically distributed 3-D teleimmersive applications. *IEEE Transactions on Multimedia* 13 (3), 573–584.

Willinger, W., Taqqu, M., Sherman, R., Wilson, D., 1997. Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level. *IEEE/ACM Transactions on Networking* 5 (1), 71–86.

Yang, Z., Yu, B., Wu, W., Nahrstedt, K., Diankov, R., Bajscy, R., 2006. A study of collaborative dancing in tele-immersive environments. In: *Proc. IEEE International Symposium on Multimedia*. pp. 177–184.

Zhmakin, A., 2011. Mathematics of human motion: from animation towards simulation (a view form the outside). *CoRR* abs/1102.4992.

Zhou, Z., Chen, X., Zhang, L., Chang, X., 2011. Internet-wide multi-party tele-immersion framework for remote 3-D collaboration. In: *Proc. IEEE International Symposium on VR Innovation*. pp. 183–188.

PAPER **B**

Synthesizing transient traffic of temporal visual signals for discrete event simulation

Mauritz Panggabean and Leif Arne Rønningen

This paper, in the original version, has been published in the Proceedings of 3rd International Congress on Ultra Modern Telecommunications and Control Systems (ICUMT) 2011, organized by IEEE in Budapest, Hungary on October 5-7, 2011.

Abstract

This paper presents the analysis and modeling of transient traffic source of temporal visual signals for DES. Transient nature implies the inability of applying established traffic theory that assumes stochastic processes. The focus is on the transient traffic of foreground object due to its motions as well as camera panning and zoom. We show that such transient traffic can be well modeled as power functions. Implemented for discrete event simulation in DEMOS, the model with a number of parameters can synthesize transient traffic of temporal visual signal in piecewise manner. Our simulation results reveal that arbitrary form of transient traffic can be synthesized in discrete event simulation as a series of the power function model with different parameter values. An exemplary application of the model in the simulation of a complex queuing system is given within our current research in futuristic multi-camera tele-immersive collaboration system.

B.1 Introduction

The audiovisual traffic generated as an input to a queuing system often has non-linear and transient behavior. This means that established stochastic theory assuming stationary processed cannot be applied in the study of such phenomena. However, aggregated control packet streams may be modeled using established queuing theory [Iversen (2007)].

In this work we assume that an input video consists of scene with important objects that attract the most viewer's attention through eye gazes. We are interested in the data traffic that represents only one particular important object and not the entire content of the video. Thus, instead of the term *video* in general meaning, we use the term *temporal visual signals* to refer to such visual information.

Seen from the sources, audiovisual data are generated as packets which rate can be constant for some time, and then instantly step to another value. We are not generally interested in mean values, but rather the transient slopes and their variations, and extreme values and their duration. Therefore, it is best to analyze the traffic in piecewise manner where each piece of traffic will be either stationary or transient, as illustrated in Figure B.1. The pictured traffic can be segmented into nine sub-traffic which mostly appear with transient properties. Each transient piece can be analyzed and modeled.

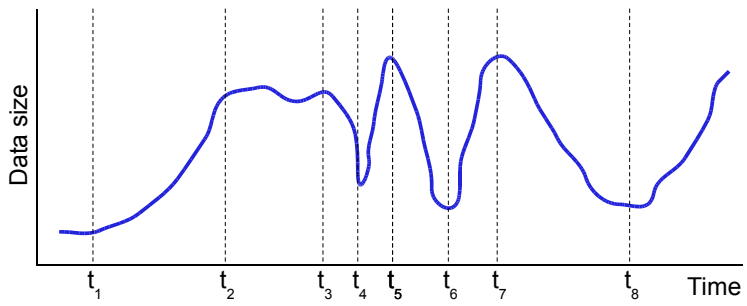


Figure B.1: An illustration of piecewise analysis of a transient traffic.

The objective of this work is investigate a correct model that in general can represent any transient traffic with just a few parameters. The model will be implemented in a tool for discrete event simulation to synthesize arbitrary transient traffic of temporal visual signals. It must have real physical meaning, for example, a person entering and disappearing from the scene, the act of camera zooming in an object, or an object performing particular motions in the scene.

This paper adopts the following structure. Section B.2 presents the analysis and modeling phases of the work. The results from implementing the model in DES are covered and discussed in Section F.3. An application of the model in DES of a complex queuing system is exemplified in Section B.4 which include some previous work that form the basis for this paper. Final remarks come in Section F.4 that concludes the paper.

B.2 From transient signals to simulation models

We focus on the transient traffic from *uncompressed* visual signals of a person as the object in three video clips with the following scenarios where the camera is always static.

1. Video clip PANNING shows the object entering the scene from the left side until disappearing on the right side, equivalent to the panning motion of the camera.

2. In the video clip ZOOM, the object gradually moves closer to the camera, analogous to a zooming-in camera.

3. The object performs some motions with the limbs at the center of the scene in video clip MOTION.

The three video clips are recorded using an off-the-shelf camcorder with 1920×1080-pixel resolution and 30Hz frame rate. A person is featured with the upper-half body in front of a background with the same color to ease the object segmentation. The clips are transcoded into de-interlaced color AVI format at 1280×720-pixel resolution to simplify the color-based object segmentation. Matlab is used on a PC with a 2.99GHz processor and 8.00GB RAM that processes 10-second input video clips.

From each video clip, a number of transient parts are selected for analysis and modeling. This yields sequences PANNING, ZOOM, and MOTION which consist of 94, 293 and 301 frames, respectively. Figure B.2 depicts exemplary frames of the three cut-out sequences and the resulting traffic from the sequences are shown in Figure B.3. The actual traffic assumes temporal *uncompressed* three-channel color visual signals. The object consists of arrays of 8×8-pixel blocks and each pixel value is encoded with eight bits.

Let us analyze some transient parts from the actual traffic in Figure B.3. Figure B.4 plots them together with the results of curve fitting to the data points using Matlab, each with two functions: power function $f(x) = ax^b$ and linear function $f(x) = cx + d$. Note that the frame numbers and the data sizes per frame are normalized to be between 0 and 1 prior to curve fitting to simplify the analysis. The stair-case shapes present in the transient traffic of sequence PANNING and MOTION imply the necessity of much higher

Figure B.2: Some exemplary frames for sequence PANNING (frame number 20, 25, 30, 35, 40, and 45), ZOOM (frame number 1, 50, 100, 150, 200, and 250), and MOTION (frame number 1, 25, 55, 118, 127, and 145). The frame numbers are from left to right in every row.

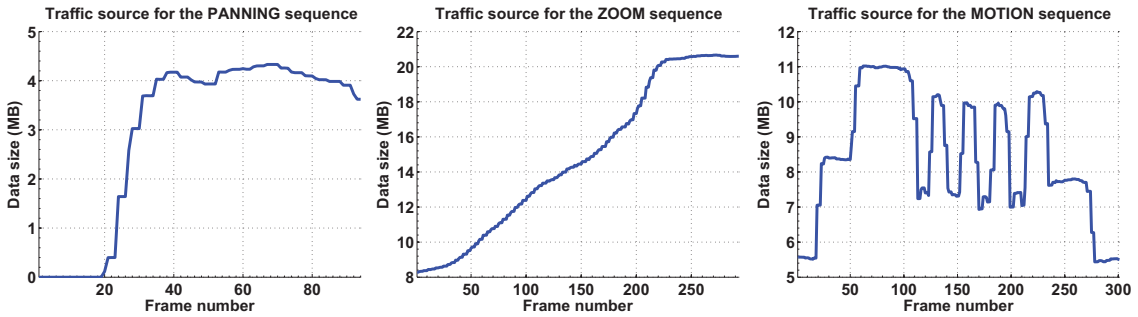


Figure B.3: Actual traffic of uncompressed temporal color visual signals for PANNING, ZOOM, and MOTION sequences (left to right) with transient parts.

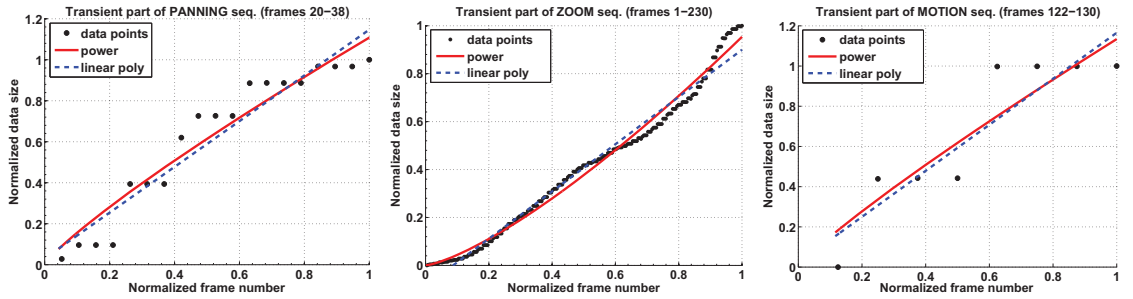


Figure B.4: Transient parts from the actual traffic of PANNING, ZOOM, and MOTION sequences (left to right), with the fitted curves of power and linear functions.

frame rates to capture more data points with higher accuracy, due to motions of high speed.

The parameter values and the root mean square errors (RMSEs) for the goodness of fit for the fitted curves in Figure B.4 are as follows. The numbers in brackets are the low and high bounds with 95% confidence.

Sequence PANNING

- a = 1.108 (1.007, 1.21), b = 0.8518 (0.6703, 1.033), RMSE = 0.09655

- c = 1.116 (0.9418, 1.291), d = 0.03048 (-0.07434, 0.1353), RMSE = 0.104

Sequence ZOOM

- a = 0.9533 (0.942, 0.9646), b = 1.339 (1.308, 1.371), RMSE = 0.03234
- c = 0.9854 (0.9665, 1.004), d = -0.08577 (-0.09675, -0.07479), RMSE = 0.04202

Sequence MOTION

- a = 1.134 (0.8431, 1.425), b = 0.8762 (0.3309, 1.421), RMSE = 0.1677
- c = 1.145 (0.6255, 1.665), d = 0.02072 (-0.3072, 0.3487), RMSE = 0.172

Evidently the power function $f(x) = ax^b$ gives smaller RMSE values than those of the linear polynomial function, although the differences are small. In fact, linear functions are special cases of power functions where $b = 1$. Our goal is to construct a correct model to generate synthetic transient traffic of temporal visual signals for DES. As presented in the next section, our results show that the power function is a good candidate for the goal.

B.3 Simulation results and discussion

The power function $f(x) = ax^b + d + e$ is implemented to synthesize temporal visual signals for DES by using DEMOS [Birtwistle (2003)]. A fast and powerful system for DES, DEMOS gives much freedom to users to improve and improvise, including changing the built-in functions or adding new features, if necessary. The following are the parameters required by the model for synthetic traffic generation in increasing manner as in Figure B.4.

1. Parameter b to define the bending of the curve, where $0.5 < b < 1$.
2. Assuming that the synthetic transient part to generate has D_{min} and D_{max} as the respective minimal and maximal data size, set parameters a and d as $a = D_{max} - D_{min}$ and $d = D_{min}$.
3. The values of e define the random smoothness of the curve at generated points at x, where e is a random real value that is uniformly distributed between parameters e_{min} and e_{max}, $e_{min} \leq e_{max}$. The generation of random value with well-known distributions is a built-in feature in DEMOS. Note that $e < a$.
4. Frame rate F and simulation time S in seconds.

Given all the parameters above, the model can generate the synthetic data rate per frame $f(x)$ with increasing trend every $t = 1/F$ seconds where $x = t/S$. For synthetic transient traffic with decreasing trend, simply use the modified power function $f(x) = a(1 - x^b) + d + e$.

We employ the model to synthesize transient traffic of temporal visual signals that resemble some transient parts of the actual traffic shown in Figure B.3. The synthetic

traffic of the signals are depicted in Figure B.5 where $e_{min} = 0$, $e_{max} = 0.5$ and $S = 1$ in all cases. The other parameter values for the traffic generation for the three sequences are as follows: $a = 4$, $b = 0.65$, $c = 0$ and $F = 19$ (sequence PANNING); $a = 12$, $b = 1.34$, $c = 8$ and $F = 230$ (sequence ZOOM); $a = 3$, $b = 0.75$, $c = 7$ and $F = 9$ (sequence MOTION, increasing part); $a = 2.5$, $b = 2.5$, $c = 7.5$ and $F = 11$ (sequence MOTION, decreasing part).

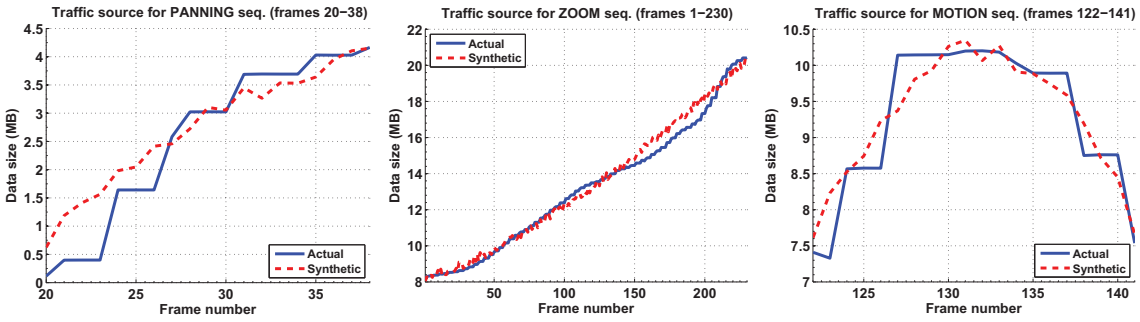


Figure B.5: Actual and synthetic traffic sources for PANNING, ZOOM, and MOTION sequences (left to right) where $e_{min} = 0$, $e_{max} = 0.5$ and $S = 1$. The other parameter values: $a = 4$, $b = 0.65$, $c = 0$ and $F = 19$ (sequence PANNING); $a = 12$, $b = 1.34$, $c = 8$ and $F = 230$ (sequence ZOOM); $a = 3$, $b = 0.75$, $c = 7$ and $F = 9$ (sequence MOTION, increasing part); $a = 2.5$, $b = 2.5$, $c = 7.5$ and $F = 11$ (sequence MOTION, decreasing part).

One can see from the actual and synthetic traffic that the power function performs very well as the model for generating synthetic transient traffic of temporal uncompressed visual signals. It will be very useful for DES that needs such signals as the input. The model serves as the building block to generate any piecewise synthetic transient traffic for simulation. The next section presents an exemplary application of the model in our research for futuristic tele-immersive collaboration systems. It includes extensive discrete-event simulation of a novel complex queuing system.

B.4 An exemplary application

NTNU has recently started an ambitious project to build advanced environments that will support various real-time artistic collaborations such as music, singing, dancing and drama [Rønningen et al. (2010)]. Using cutting edge technologies such as autostereoscopic multiview 3D displays in all the surfaces, the environments will function also as laboratories in which interesting multidisciplinary research questions in science, technology and arts will be studied. Table B.1 lists the main technical requirements for the envisioned collaborations related to visual aspects.

Guaranteeing maximal EED is essential to facilitate good synchronization in real-time artistic collaborations from distributed places, as shown by, for example, Chafe et al. (2004) and Yang et al. (2006). The delay can be as low as 11.5ms, including propagation and all processing delays. We estimate the same need in other unexplored forms of artistic collaborations, such as real-time singing practice of several locally distributed singers and remote conducting between a conductor and a choir in different locations.

Table B.1: Technical requirements for the envisioned tele-immersive collaboration related to visual aspects.

Nr.	Main technical requirements
1.	Guaranteed maximum EED ≤ 10-20ms (cf. 150ms for videoconference [ITU-T (2003)])
2.	Near-natural video quality
3.	Auto-stereoscopic multiview 3D vision
4.	High spatial and temporal resolution due to life-size dimension of objects i.e. mostly humans
5.	Accurate representation of physical presence cues e.g. eye contact and gesture
6.	Quality allowed to vary with time due to different technical specifications among collaboration spaces
7.	Graceful quality degradation due to traffic overload or failure

The Internet and the source-coding approach are not designed to address and deliver such a guarantee for video data. Thus we argue that network nodes should be designed such that they are able to intelligently drop video packets due to immediate traffic conditions with graceful VQ degradation. Given the basic definition of data compression as data reduction necessary for storage or transmission of an information, we call this approach *compression-by-network* (CbN). Figure B.6 depicts how it differs from the source-coding approach with raw input digital signals.

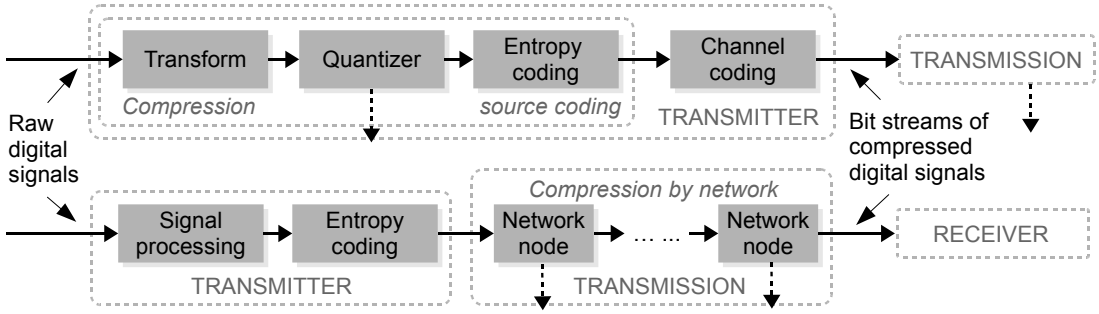


Figure B.6: A comparative overview of the standard source coding (top) and the CbN (bottom) approaches in lossy compression of digital signals. Arrows with dashed lines denote the reduction or loss of information. Channel coding in CbN is also assumed.

CbN is designed to operate on the DMP architecture [Rønningen et al. (2010)], which involves novel queuing systems with advanced control feedback mechanisms. The input to CbN will be uncompressed temporal visual signals that are assumed to be produced by multi-array of cameras in a collaboration space. Simulation will be the most appropriate approach to study and investigate such a system of high complexity, before the physically built system allows extensive measurements. The model proposed in this paper will certainly be useful for such simulations. More details and results from this research are available as reported work, for example, by Rønningen (2011) and Dai (2008). Unfortunately, to our best knowledge, we have not found other reported work on modeling and synthesis of input transient traffic for DES.

B.5 Conclusion

We have presented our analysis of transient traffic of temporal visual signals that represents a person in the video scene as the important object. Our study covers three main cases, i.e. panning and zooming modes of the camera, as well as motions of the object. Our objective is to construct a correct model of such transient traffic. The model implementation will be used for synthesizing such traffic in DES. Our study reveals that power function can perform as such a model. The results from implementing the model in DEMOS show that the model can be easily employed as a building block to synthesize any transient traffic in piecewise manner. Finally, an example of the model application for DES of a complex queuing system is given.

References

Birtwistle, G., 2003. *DEMOS - A System for Discrete Event Modelling on Simula*. School of Computer Science, University of Sheffield.

Chafe, C., Gurevich, M., Leslie, G., Tyan, S., 2004. Effect of time delay on ensemble accuracy. In: *Proc. International Symposium on Musical Acoustics*.

Dai, K., 2008. *Performance analysis of the dropping scheme of DMP network nodes*. Master's thesis, ITEM, NTNU.

ITU-T, May 2003. *Recommendation G.114 – One-way transmission time, general recommendations on the transmission quality for an entire international telephone connection*.

Iversen, V., 2007. *Teletraffic Engineering Handbook*. ITU.

Rønningen, L. A., 2011. *The Distributed Multimedia Plays Architecture (version 3.20)*. Tech. rep., ITEM, NTNU.

Rønningen, L. A., Panggabean, M., Tamer, Ö., 2010. Toward futuristic near-natural collaborations on Distributed Multimedia Plays architecture. In: *Proc. IEEE International Symposium on Signal Proccessing and Information Technology (ISSPIT)*. pp. 102–107.

Yang, Z., Yu, B., Wu, W., Nahrstedt, K., Diankov, R., Bajscy, R., 2006. A study of collaborative dancing in tele-immersive environments. In: *Proc. IEEE International Symposium on Multimedia*. pp. 177–184.

PAPER **C**

Parameterization of windowed kriging for compression-by-network of natural images

Mauritz Panggabean and Leif Arne Rønningen

This paper, in the original version, has been published in the Proceedings of 7th International Symposium on Image and Signal Processing and Analysis (ISPA) 2011, organized by IEEE and EURASIP in Dubrovnik, Croatia on September 4-6, 2011.

Abstract

Artistic elements in immersive collaborations that depend on visual cues, such as collaborative percussions and dancing, require a maximal EED for both audio and video data to ensure harmonious synchronization. Such delays have been shown to be extremely low in milliseconds. To guarantee such delay for video data with graceful quality degradation, we envision that network nodes can reduce video data intelligently in changing traffic conditions. Thus we call this approach CbN in contrast to the source coding that comes with no such guarantee. WK interpolation has been introduced as a technique that matches the novel approach. This paper presents an empirical study on the parameters of the technique for interpolating grayscale natural images in terms of image quality and processing time. Recommended values for the parameters are given together with interesting insights as the basis to extend the application of the technique for interpolating chroma images.

C.1 Introduction

Along with greater interest in green technology and rapid technological advances, real-time multi-party collaboration from distributed places via tele-immersive environment has been an increasingly active research area. NTNU has recently started an ambitious project to build such environments that will support various real-time artistic collaborations such as music, singing, dancing and drama [Rønningen et al. (2010)]. Using cutting edge technologies such as autostereoscopic multiview 3D displays in all the surfaces, the environments will function also as laboratories in which interesting multidisciplinary research questions in science, technology and arts will be studied. Table B.1 lists the main technical requirements on the important aspects for the envisioned collaborations.

One such question is on the effect of delay in artistic collaborations. Chafe et al. (2004) reported that the optimal delay for synchronizing rhythmic clapping hands from different places is 11.5ms, including propagation and all processing delays. Longer delays will produce increasingly severe tempo deceleration while shorter ones yield a modest yet surprising acceleration. Percussions are rhythmically very similar to clapping hands, so collaborative musicians playing such instruments will require the same delay for both audio and video data. A study on collaborative dancing by Yang et al. (2006) also indicates the importance of constant delay, particularly for video data, to yield good synchronization between dancers as it depends on visual cues.

We estimate the same need in collaborative singing and remote conducting. This implies that, with tremendous video traffic from the tele-immersive environments envisioned above, artistic collaborations that are real-time, multiparty and distributed are only possible if the maximal EED can be guaranteed with graceful VQ degradation. Unfortunately the Internet and the source-coding approach are not designed to address and deliver such a guarantee for video data. To provide it, we argue that network nodes should be designed such that they are able to intelligently drop video packets due to immediate traffic conditions but also with graceful VQ degradation. Given the basic definition of data compression as data reduction necessary for storage or transmission of

an information, we call this approach CbN as it differs from the source-coding approach for raw input digital signals, as depicted in Figure B.6.

Some fundamental differences between CbN and source coding are evident from Figure B.6. First, the quantization at the transmitter is responsible for lossy compression in source coding while data reduction by CbN occurs in each network node and also at the transmitter. Second, the latter implies that the transmitter in a CbN system has the freedom to reduce data or not because the network can do it later, if necessary, by keeping the VQ degradation graceful. Whenever the queueing delay of the input bit streams in a network node of a CbN system is more than the maximum allowed, the node can drop some of them, fully or partially, to guarantee the maximal EED. The time for data reduction in each network node must also be minimized, affecting how the entropy coding must be designed later. This means that CbN offers fully adaptive control to the network with respect to delay and quality. Pursuing minimal computational delay in CbN also favors object-based and parallel processing while avoiding the exploitation of temporal redundancies. On the contrary, the network in source coding transmits only what is passed on by the transmitter. Thus loss of packets during transmission in source coding can cause annoying artifacts in the end. Third, guaranteeing delay in CbN may yield logical increase in bitrate compared to that in source coding.

We have proposed a novel network architecture for CbN, namely DMP [Rønningen et al. (2010)]. Its basic design goal is to simplify and extend the quality of existing collaborative systems. Standards for video conferencing systems, such as the H.323, need a large number of different protocols to work properly. DMP reduces the number of protocols to two and correspondingly reduces the number of architectural layers to three. To handle the high data rates, data processing has to be performed by hardware employing parallel transmission and computations. Figure C.1 depicts the queueing model of dropping and prioritizing packets in a network node of a CbN system on DMP architecture.

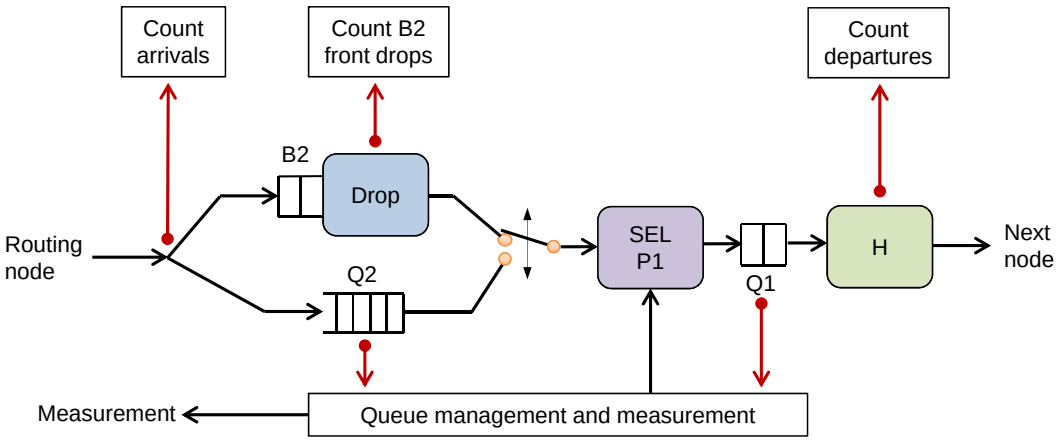


Figure C.1: The queueing model of dropping and prioritizing packets in a network node of a CbN system on DMP architecture

Figure C.2 illustrates an overview of a CbN system for color images using optimal interpolation at the receiver. First, the input original color image is converted into

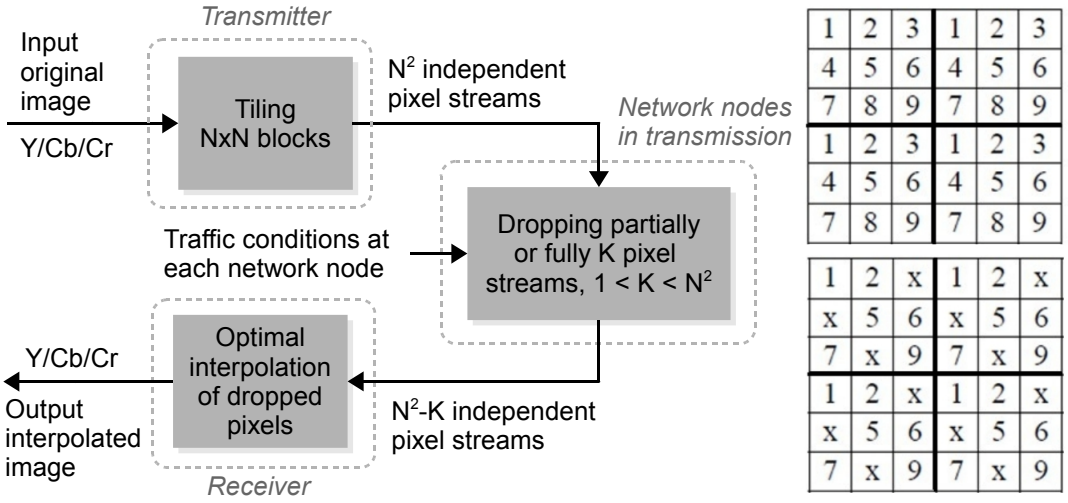


Figure C.2: An overview of a proposed CbN system for color images using optimal interpolation by direct transmission of pixel values with entropy coding (left). Tiling 3×3 blocks in an image (right-top); dropping stream number 3, 4 and 8 (right-bottom). Each pixel value of the dropped streams denoted by × will be optimally interpolated from the remaining pixels at the receiver.

YCbCr color space yielding three images, each for a channel. Then $N \times N$ blocks are tiled to each image, giving N^2 independent bit streams of pixel values after entropy and channel coding. The sender directly transmits all these pixel streams into the network. Values of the pixels dropped during transmission will be estimated by applying optimal interpolation in the sense of mean square error to the received bit streams at the receiver. Our initial study has shown that kriging with window mechanism works well in interpolating the luminance, hence called WK interpolation [Panggabean et al. (2010)]. Not much study related to kriging in image processing or compression has been reported, which the most recent can be found in [Panagiotopoulou and Anastassopoulos (2007); Ruiz-Alzola et al. (2005)].

This paper presents the parameterization of WK to interpolate the intensity values in the dropped pixels of grayscale natural images, following the model in Figure C.2. It offers a thorough and comprehensive study on WK with five standard test images as sample results based on our initial work in [Panggabean et al. (2010)]. This paper is organized as follows. Section D.2 presents the basic concepts of kriging and the proposed WK interpolation, since they are uncommon to use for image processing and compression. Experimental results and discussion follow in Section D.3 and finally Section F.4 concludes this paper.

C.2 WK interpolation

This section presents some formal basics of kriging and WK interpolation. Let $z(\mathbf{s}_1)$, $z(\mathbf{s}_2), ..., z(\mathbf{s}_n)$ be denoted as a set of n observations where $\mathbf{s}_i = (x_i, y_i)$ is a vector of a point coordinate in spatial region R. The observations are realizations of an underlying random function $Z(\mathbf{s})$ modeled as consisting of a trend component, $m(\mathbf{s})$, and a residual

component, $r(\mathbf{s})$, hence $Z(\mathbf{s}) = m(\mathbf{s}) + r(\mathbf{s})$. $Z(\mathbf{s})$ is defined as *second-order stationary* if, for a vector $\mathbf{h}$ linking any two points in R, $E[Z(\mathbf{s}+\mathbf{h})] = E[Z(\mathbf{s})]$, i.e. a finite constant independent of $\mathbf{s}$, and $cov[Z(\mathbf{s}+\mathbf{h}), Z(\mathbf{s})] = C(\mathbf{h})$ that depends only on $\mathbf{h}$. A second-order stationary $Z(\mathbf{s})$ is also *intrinsically stationary*, i.e. $E[Z(\mathbf{s}+\mathbf{h}) - Z(\mathbf{s})] = 0$ and $var[Z(\mathbf{s}+\mathbf{h}) - Z(\mathbf{s})] = 2\gamma(\mathbf{h})$, where $\gamma(\mathbf{h})$ is called the *semivariogram* of $Z(\mathbf{s})$. It can be obtained from the covariance function that $\gamma(\mathbf{h}) = C(0) - C(\mathbf{h})$. Experimental semivariogram $\hat{\gamma}(\mathbf{h})$ can be computed from input samples and then fitted to authorized theoretical semivariogram models such as spherical, exponential, and Gaussian.

In *simple kriging*, $m(\mathbf{s})$ is assumed a known mean constant over the entire domain. In *ordinary kriging* (OK), $m(\mathbf{s})$ is unknown and constant only in the local neighborhood of each estimation point $\mathbf{s}_0$. *Universal kriging* (UK) applies when $m(\mathbf{s})$ is an unknown non-constant deterministic component and $r(\mathbf{s})$ is a residual random function assumed to be second-order stationary. Thus OK is a special case of UK.

Now we briefly show estimation of values at unobserved points via kriging with OK. Given second-order stationary $Z(\mathbf{s})$, OK is the best linear unbiased estimator because it uses the linear weighted estimator $\hat{Z}(\mathbf{s}_0) = \sum_{i=1}^{n} w_i Z(\mathbf{s})$ with unbiasedness ensured by the condition $\sum_{i=1}^{n} w_i = 1$ aiming at minimizing the error variance. The estimator variance is given by $\sigma_E^2 = var[\hat{Z}(\mathbf{s}_0) - Z(\mathbf{s}_0)] = C(0) + \mathbf{w}^T K\mathbf{w} - 2\mathbf{c}_0^T\mathbf{w}$ where $\mathbf{c}_0 = [C(\mathbf{s}_0 - \mathbf{s}_1), ..., C(\mathbf{s}_0 - \mathbf{s}_n)]^T$ and $\mathbf{w} = [w_1, ..., w_n]^T$. Minimizing σ_E^2 yields

$$\begin{pmatrix} K & \mathbf{1} \\ \mathbf{1}^T & 0 \end{pmatrix} \begin{pmatrix} \mathbf{w} \\ \mu \end{pmatrix} = \begin{pmatrix} \mathbf{c}_0 \\ 1 \end{pmatrix}$$

as the OK system where $K_{i,j} = C(\mathbf{s}_i - \mathbf{s}_j)$ and μ is the Lagrange multiplier. Under the assumption of intrinsic stationarity, the semiavariogram can be used instead of the covariance matrix since $\gamma(\mathbf{h}) = C(0) - C(\mathbf{h})$. Hence it modifies the OK system to be

$$\begin{pmatrix} \Gamma & \mathbf{1} \\ \mathbf{1}^T & 0 \end{pmatrix} \begin{pmatrix} \mathbf{w} \\ \mu \end{pmatrix} = \begin{pmatrix} \boldsymbol{\gamma}_0 \\ 1 \end{pmatrix}$$

where $\Gamma_{i,j} = \gamma(\mathbf{s}_i - \mathbf{s}_j)$ and $\gamma_{0,j} = \gamma(\mathbf{s}_0 - \mathbf{s}_j)$. Now the weights $\mathbf{w}$ can be obtained and used for estimating the values at unobserved points.

Kriging becomes more computationally expensive as the number of unobserved points to estimate increases, particulary in interpolating intensity values in images where data points to interpolate are much more than those in geostatistics. To enable kriging for images of high resolution in CbN, we propose the use of WK interpolation as follows [Panggabean et al. (2010)].

1. Tile $S \times S$ blocks on the input image of $W \times H$ resolution. Assume that W and H are evenly divisible by S for simplification.

2. Tile $N \times N$ pixel stream pattern on the image. Keep K streams, partially or fully, as the input samples for interpolation, as shown in Figure C.2, where $1 < K \leq N^2$. Set vector $\mathbf{k}$ which elements are the index numbers of the retained input streams. K equals the length of $\mathbf{k}$.

3. For each block, assign a $D \times D$ window that shares the same center point with the block. To avoid typical border artifact caused by WK [Panggabean et al. (2010)],

set $D/2 = S/2 + d$ where d denotes the number of additional pixels. Then employ kriging by using the samples in the window to interpolate each pixel value within the window.

4. Keep the output pixel values in the $S \times S$ area centered at the center point of each window and form the interpolated image by arranging the output blocks in their corresponding locations.

Focusing on the three WK parameters N, S, and $\mathbf{k}$ while varying the parameters for the exponential semivariogram, our objective is to study the effect of each parameter in terms of image quality and computational cost that will lead us to the optimal configuration. The next section presents and discusses the results.

C.3 Results and discussion

In this section we present and discuss our experimental results from five grayscale standard test images with 512×512-pixel resolution, namely LENNA, MANDRILL, BARBARA, PEPPER, and BOAT. We employ the DACE Matlab toolbox [Nielsen et al. (2009)] on a PC with a 2.80GHz processor and 8.00GB RAM for the kriging implementation. The toolbox consists of two most important operations: modeling the input data into the DACE model by the `dacefit` function and predicting the optimal values based on the model by the `predictor` function.

The toolbox supports universal or regression kriging as it models a random variable as a realization of predefined regression and correlation models, and a stochastic function. Exponential, gaussian, linear, spherical, cubic, and spline correlation models are provided. Empirical semivariograms of natural images fit the exponential model [Panggabean et al. (2010)] and this leads to the use of that model by choosing `correxp` in the `dacefit` function in the toolbox. Two parameters have to be set for the model: `theta0` and the functions for regression models, i.e. `regpoly0`, `regpoly1` and `regpoly2` for zero-, first-, and second-order polynomial, respectively. Before continuing with the WK parameters, first we examine `theta0` and the regression models for all the test images to select the best configuration in terms of image quality and processing time. In all cases we set $d = 3$ pixels to fix the border artifacts following [Panggabean et al. (2010)], as illustrated in Figure C.3.

Figure C.4 depicts the quality of the interpolated test images with `theta0` from 1 to 5 and `regpoly0` for the regression model where $N = 3$, $S = 32$, and $\mathbf{k} = [1]$. One can see that the order of the polynomial for regression model has negligible effect to image quality. Increasing `theta0` causes only little improvement in image quality. Thus it is difficult to plot the metric values from the test images as they are quite different. Therefore, we introduce ΔPSNR and ΔSSIM as the difference between the value of the metric and its highest value for a test image. We plot ΔPSNR and ΔSSIM for all test images to display the little increase clearly. The highest values are given in the caption and indicated by zeros on the vertical axis. It is obvious that `theta0` = 1 gives the best result and thus will be used in the following experiments. In addition, the processing times in any configuration of the two parameters are very similar.

Figure C.3: The image on the right shows the border artifact when $d = 0$, compared to the original (middle). Both are from the bounding box in the image on the left.

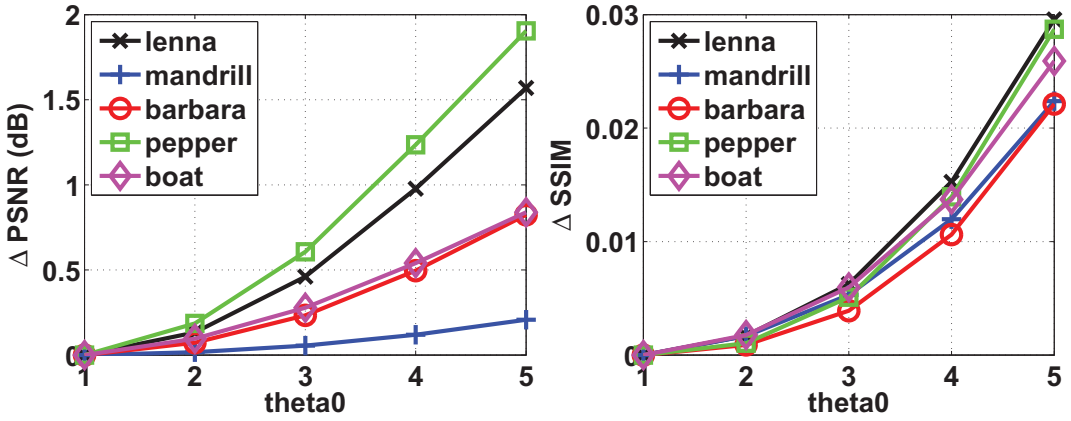


Figure C.4: The effect of varying `theta0` to image quality with `regpoly0` selected for the regression model. The highest values of (PSNR, SSIM) for LENNA, MANDRILL, BARBARA, PEPPER, and BOAT are (34.91, 0.909), (30.20, 0.675), (31.98, 0.779), (34.47, 0.919), (32.83, 0.837), respectively.

Having examined the parameters for the exponential correlation model, we continue by studying the effect of the three WK parameters S, N, and $\mathbf{k}$ to image quality and processing time for all test images with `theta0` = 1 and `regpoly0`. There are three tests in this step. First, we vary $S = [4,8,16,32]$ pixels while $N = 3$ pixels and $\mathbf{k} = [1]$. Second, with S selected from the first test and $\mathbf{k} = [1]$, the effect from N is examined where $N = [2,3,4]$ pixels. Finally, while $N = 3$ and $S = 16$, various configuration of retained pixel streams are studied by setting $\mathbf{k}$ = [(1), (5), (1,9), (3,7), (1,5,9), (3,5,7), (1,2,3), (2,4,8), (1,3,5,7,9)]. The results from the three tests for processing time and image quality are presented in Figures C.5 and C.6, respectively.

The use of the difference value instead of the actual ones in Figure C.6 shows that the resulting image quality is not affected very much by the three WK parameters, but they perform quite differently in terms of processing time, as shown in Figure C.5. Higher

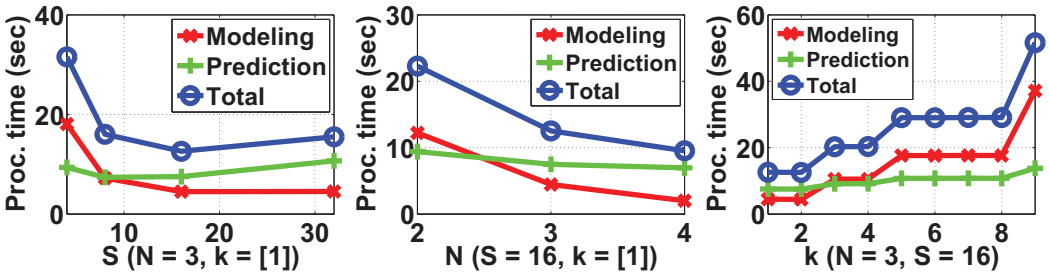


Figure C.5: The effect of varying S, N and **k** to the total processing time mainly devoted to modeling and prediction. The values of **k** from 1 to 9 in the right diagram refer to the nine configurations of **k** = [(1), (5), (1,9), (3,7), (1,5,9), (3,5,7), (1,2,3), (2,4,8), (1,3,5,7,9)], respectively.

S requires somewhat constant time for prediction and decreasing modeling time. It is also obvious that more time for modeling is needed with more input intensity values. Interestingly the same also happens if the inputs are too few, i.e. when $S = 4, N = 3$ and $K = 1$, which is similar to the modeling time when $S = 16, N = 3$ and $K = 3$. Note that WK also estimates the intensity values at the points of the input values. Replacing these estimated values with the original ones will cause another type of visual artifact similar to salt-and-pepper [Panggabean et al. (2010)]. Moreover, although observing Figure C.6 reveals that small improvement in image quality is generally achieved by smaller N as well as higher S and K, a number of unexpected anomalies occur e.g. the ΔPSNR of PEPPER and BARBARA images in the top left diagram. These indicate that, in terms of image quality, WK is not very robust in interpolating luminance information.

For the top row of Figure C.6 where $S = [4,8,16,32]$, $N = 3$ and **k** = [1], the highest values of (PSNR, SSIM) for LENNA, MANDRILL, BARBARA, PEPPER, and BOAT are (34.91, 0.909), (30.20, 0.675), (31.98, 0.779), (34.47, 0.919), (32.83, 0.837), respectively. The highest values for the other two tests are also similar. From these values, the order of test images in increasing image quality is MANDRILL, BARBARA, BOAT, PEPPER, and LENNA. The order consistently follows the level of frequency content in the images, e.g. there are very small areas in MANDRILL image which contents are of low frequency, contrary to PEPPER and LENNA images. From comparing the level of difference values in image quality in Figure C.6, the WK parameters in increasing impact to image quality can be ordered as S, **k**, and N. Taking processing time into account with this order in image quality, the recommended values of the WK parameters for processing luminance are $S = 16$ pixels, $K = 1$ (e.g. **k** = [1]), and $N = 3$ pixels after selecting `theta0` = 1 and `regpoly0` for regression model.

Now let us take a look on the perceptual quality of the output images. Figure C.7 displays the original test images in the top row with the corresponding interpolated images in the bottom row where $S = 16$ pixels, $N = 3$ pixels and **k** = [1], all with 512×512-pixel resolution. We focus on how WK interpolates areas in the image with textures, edges, plain content and objects such as faces. Figure C.8 presents sub-images cropped from the areas indicated by the bounding boxes in Figure C.7.

Sub-images 3, 7, 8, 9 and 10 in Figure C.8 display textures in the forms of fur, textile stripes pattern, and hair. Loss of details clearly occurs in the fur and the hair (sub-images 3 and 7), while the consistency of the textile pattern and the lines in sub-images

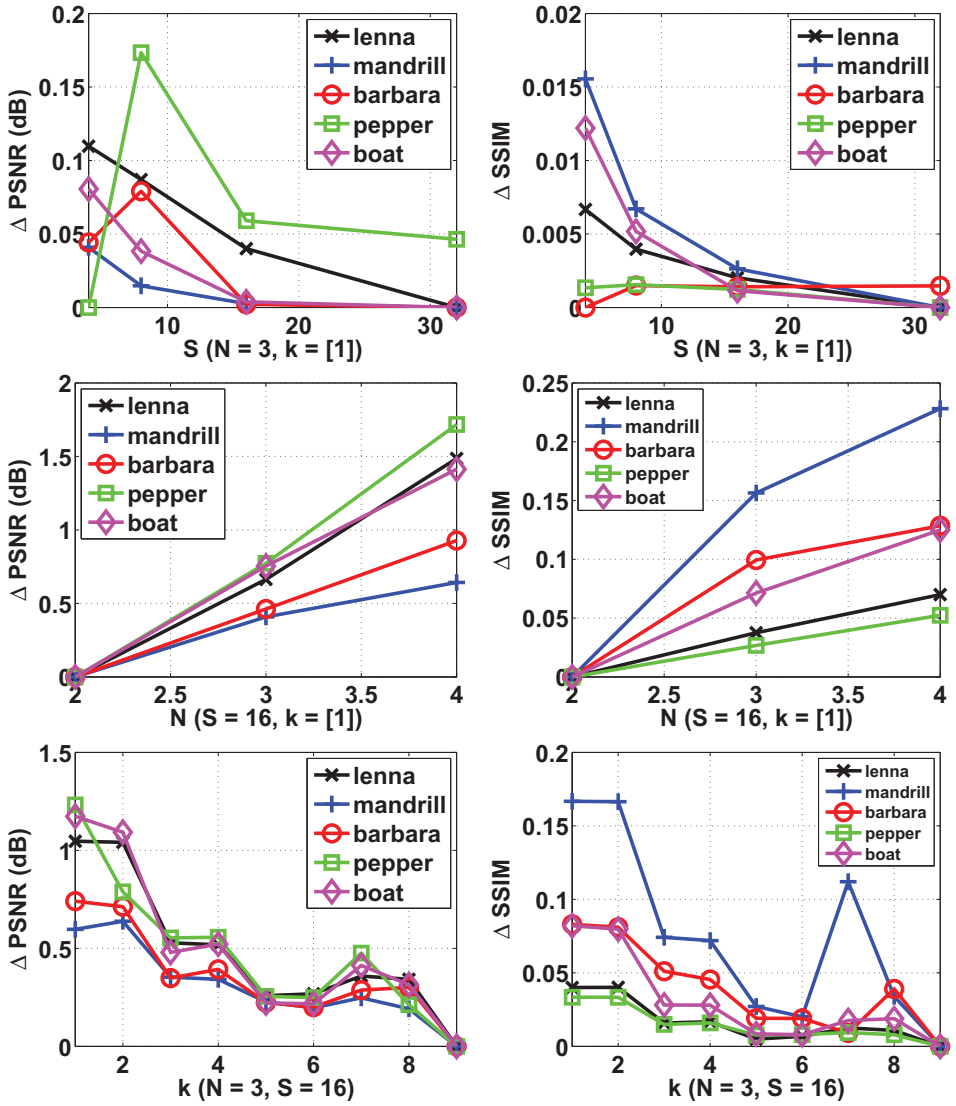


Figure C.6: The effect of varying S, N and $\mathbf{k}$ to image quality. For the top row, the highest values of (PSNR, SSIM) for the images are the same with those in Figure C.4. For the middle row, the highest values of (PSNR, SSIM) for LENNA, MANDRILL, BARBARA, PEPPER, and BOAT are (35.53, 0.944), (30.60, 0.829), (32.43, 0.878), (35.22, 0.945), (33.58, 0.907), respectively. For the bottom row, the highest values of (PSNR, SSIM) for the images are (35.92, 0.947), (30.79, 0.839), (32.71, 0.862), (35.68, 0.951), (34.00, 0.918), respectively. The values from 1 to 9 on the horizontal axes in the bottom row respectively refer to the nine configurations of $\mathbf{k}$ = [(1), (5), (1,9), (3,7), (1,5,9), (3,5,7), (1,2,3), (2,4,8), (1,3,5,7,9)].

8, 9, and 10 cannot be established. The edges shown e.g in sub-images 2, 4 and 11 suffer from staircase artifact. The text object in sub-image 5 becomes unreadable as basically it consists of edges. However WK can perform quite well in interpolating human faces in the bottom row, although it slightly blurs the images. Similar blur effect also occurs in more stationary and softer textures such as the skins, the clouds and the grain in sub-images 2, 4, 8, and 12.

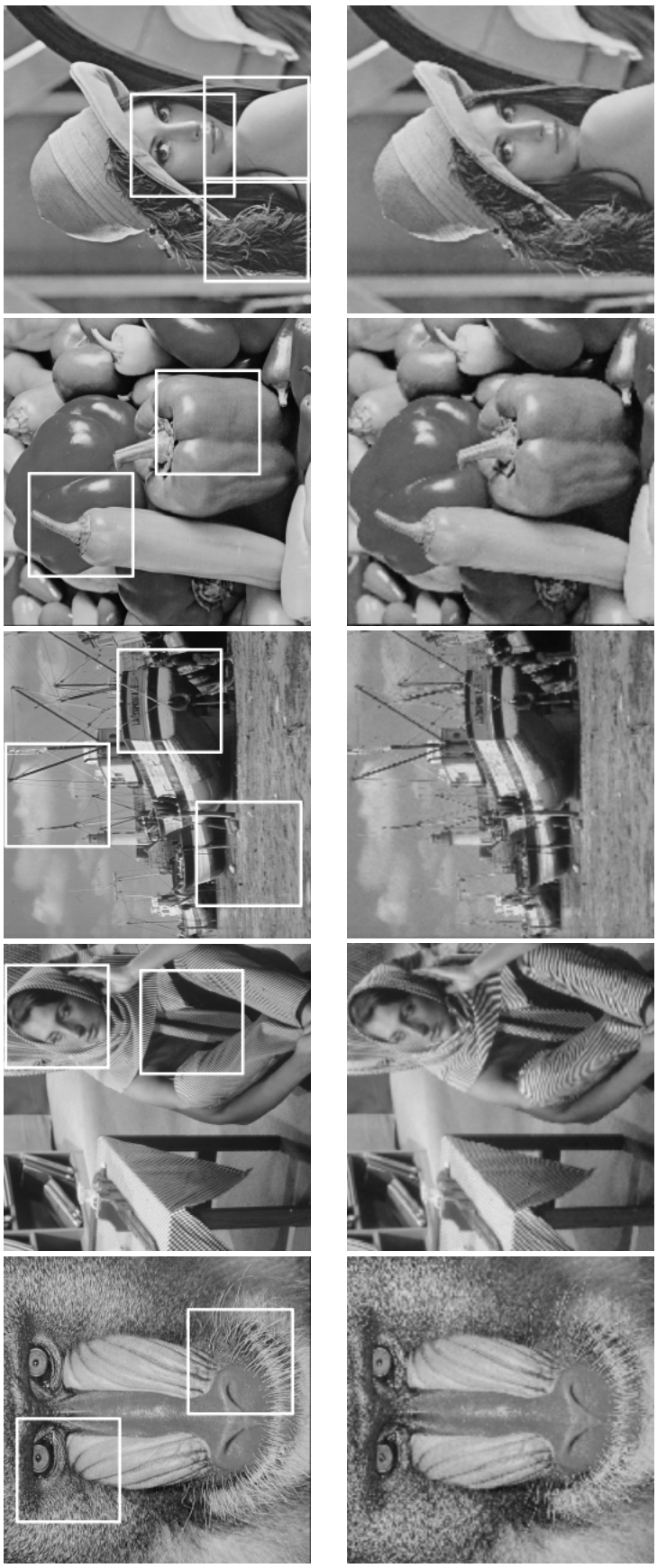


Figure C.7: Left to right: MANDRILL, BARBARA, BOAT, PEPPER, and LENNA test images. The top row shows the original images with bounding boxes of the images shown in Figure C.8 and the interpolated images in the bottom row where $S = 16$ pixels, $N = 3$ pixels and $\mathbf{k} = [1]$. The quality of the interpolated images (PSNR/SSIM) from left to right is (30.19, 0.673), (31.97, 0.779), (32.83, 0.836), (34.45, 0.918), (34.87, 0.907).

Figure C.8: Pairs of cropped areas from the original and interpolated images shown by bounding boxes in Figure C.7 for subjective assessment on screen by the readers.

These sample sub-images show that high frequency contents such as textures and edges are very challenging to interpolate by WK with compelling perceptual quality from merely 1/9 of the image pixels. Nevertheless, it is important to remember that optimal estimation in mean-square-error sense of the missing intensity values using only 1/9 of the image pixels. Therefore such results from WK are still worth appreciation. It is also worth noting that the interpolated areas with low activity such as the cloudy sky and the skins still appear with decent quality. As WK can interpolate such areas using merely 11% of the image pixels, it is very likely that WK can interpolate images or areas of low activity with much less samples.

These results lead us to our hypothesis that WK will be more suitable for interpolating chrominance information of color images as low activity areas are dominant in chroma images. As an example, we interpolate the chroma images (Cb and Cr channels) of LENNA color image at 512×512 pixels. With $S = 16$, $N = 3$ and $\mathbf{k} = [1]$, the image quality (PSNR/SSIM) for the interpolated Cb and Cr images is (41.61/0.953) and (41.78/0.952), respectively. These are much higher than the quality of the interpolated luminance image of LENNA. By exploiting the human visual system that is much more insensitive to changes in color than those in edges, the artifact caused by high number of reduced data due to high N can be overlooked. We have been pursuing more thorough and comprehensive study on employing WK for interpolating chroma images and investigating the effect to the quality of the output color images. Some initial results on these can be found in [Panggabean and Rønningen (2011)].

C.4 Conclusion

We have presented our study of grayscale image interpolation using WK for delay-bounded transmission with the CbN approach. It gives the recommended values for WK parameters. Although WK is suitable to the approach where the pixels values are directly transmitted and discarded during transmission, results from our study reveal that high frequency contents such as edges and textures are difficult to interpolate using WK while it performs well in low frequency contents. It is a fact that chroma images from the Cb and Cr channels have more low frequency content and much less edges and textures than those in luminance. Furthermore human eyes are more sensitive to changes in edges than those in chroma. These facts make WK promising for interpolating chroma images with the CbN approach and our preliminary reported work confirm this. Eventually our envisioned tele-immersive collaboration surfaces will certainly exchange color visual signals and it implies that a CbN-based system must be able to transmit and reduce the data of chroma information. WK can be a promising technique suitable for this purpose and our in-depth examination of this hypotesis has been in progress.

References

Chafe, C., Gurevich, M., Leslie, G., Tyan, S., 2004. Effect of time delay on ensemble accuracy. In: *Proc. International Symposium on Musical Acoustics*.

Nielsen, H., Lophaven, S., Søndergaard, J., 2009. DACE, a Matlab kriging toolbox. Technical University of Denmark, www2.imm.dtu.dk/~hbn/dace/.

Panagiotopoulou, A., Anastassopoulos, V., 2007. Super-resolution image reconstruction employing kriging interpolation technique. In: *Proc. IEEE International Workshop on Systems, Signals and Image Processing*. pp. 144–147.

Panggabean, M., Rønningen, L. A., 2011. Chroma interpolation using windowed kriging for color-image compression-by-network with guaranteed delay. In: *Proc. International Conference on Digital Signal Processing (DSP)*. pp. 1–6.

Panggabean, M., Tamer, Ö., Rønningen, L. A., 2010. Parallel image transmission and compression using windowed kriging interpolations. In: *Proc. IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*. pp. 315–320.

Rønningen, L. A., Panggabean, M., Tamer, Ö., 2010. Toward futuristic near-natural collaborations on Distributed Multimedia Plays architecture. In: *Proc. IEEE International Symposium on Signal Proccessing and Information Technology (ISSPIT)*. pp. 102–107.

Ruiz-Alzola, J., Alberola-Lopez, C., Westin, C.-F., 2005. Kriging filters for multidimensional signal processing. *Signal Processing* 85 (2), 413–439.

Yang, Z., Yu, B., Wu, W., Nahrstedt, K., Diankov, R., Bajscy, R., 2006. A study of collaborative dancing in tele-immersive environments. In: *Proc. IEEE International Symposium on Multimedia*. pp. 177–184.

PAPER **D**

Chroma interpolation using windowed kriging for color-image compression by network with guaranteed delay

Mauritz Panggabean and Leif Arne Rønningen

This paper, in the original version, has been published in the Proceedings of 17th International Conference on Digital Signal Processing (DSP) 2011, organized by IEEE and EURASIP in Corfu, Greece on July 6-8, 2011.

Digital Object Identifier: 10.1109/ICDSP.2011.6004935.

Abstract

Artistic elements in immersive collaborations, like collaborative percussions and dancing, depend on visual cues and thus require a same constant delay for both audio and video data to ensure harmonious synchronization. We envision the CbN approach where network nodes can reduce video data intelligently due to changing traffic conditions to guarantee maximum constant delay for video data with graceful quality degradation. To process color information with that approach, in this paper we propose using WK for chroma interpolation and investigate the effects. We define color compressibility as the maximum compression ratio of color information without noticeable artifact. Our results show that, based on compressibility, network nodes can discard 99% of chroma pixels and the receiver interpolates the chroma images using only the received 1% to yield the output color image without salient artifact. We discover that red and similar colors have lower compressibility than blue, green or brown.

D.1 Introduction

Along with greater interest in green technology and rapid technological advances, multiparty collaboration from distributed places via tele-immersive environment has been an increasingly active research area. NTNU has recently started an ambitious project to build such environments that will support various artistic collaborations such as music, sing-ing, dancing and drama [Rønningen et al. (2010)]. Using cutting edge technologies such as autostereoscopic multiview 3D displays in all the surfaces, the environments also will function as laboratories in which interesting multidisciplinary RQs in science, technology and arts will be addressed and studied.

One such question is on the effect of delay in artistic collaborations. Chafe et al. (2004) reports that the optimal delay for synchronizing rhythmic clapping hands from different places is 11.5 ms, including transmission and all processing delays. Longer delays will produce increasingly severe tempo deceleration while shorter ones yield a modest yet surprising acceleration. Percussions are rhythmically very similar to clapping hands, so collaborative musicians playing such instruments will require the same delay for both audio and video data. A study on collaborative dancing by Yang et al. (2006) also indicates the importance of constant delay, particularly for video data, to yield good synchronization between dancers as it depends on visual cues. We estimate the same need in collaborative singing and remote conducting.

Internet today is still unable to guarantee a constant maximum delay for video data. To provide such warranty, we argue that network nodes should be able to intelligently drop video packets despite changing traffic conditions but also with graceful quality degradation. We call this approach CbN. Compression by current coding standards is conducted only by the sender, hence CbN requires their modifications or novel approaches. However, very short delay as mentioned above implies minimizing or even avoiding video coding at the expense of high increase in bit rate. It also implies the need for intra-frame and object-based processing as well as parallel transmission. We have

proposed a novel network architecture namely DMP for CbN approach [Rønningen et al. (2010)].

A simple data representation for parallel transmission, as shown in Figure C.2 (right), is to assign $N \times N$ blocks directly to the pixels in a video frame, giving N^2 bit streams of pixel values. Objects in the image can also be segmented, processed and transmitted independently. The number of pixel streams in a segmented object may vary depending on its visual content. Thus network nodes can drop pixel streams after entropy coding to instantly reduce the bit rate according to immediate traffic conditions. The time for data reduction by network nodes must also be minimized, which affects how the entropy coding must be designed later. The dropped pixels will be estimated by applying optimal interpolation in the sense of mean square error to the received bit streams at the receiver.

Figure D.1 illustrates the overview of the end-to-end system. Searching for such interpolation leads us to *kriging*, a technique widely used in geostatistics. We have shown that kriging is a promising candidate to realize such system by adopting window mechanism, hence called WK interpolation [Panggabean et al. (2010)]. Interestingly not much study related to kriging in image processing or compression has been reported, which the most recent can be found in [Panagiotopoulou and Anastassopoulos (2007); Ruiz-Alzola et al. (2005)].

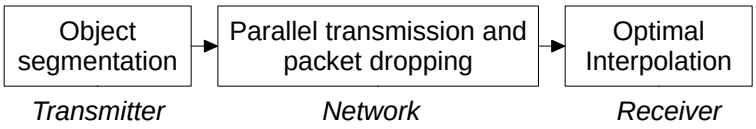


Figure D.1: Overview of CbN using kriging.

This paper presents the use of WK to interpolate the dropped pixels in chroma images (Cb and Cr channels) following the model in Figure C.2 (right). Then we study the VQ of the resulting color images that lead us to interesting insights. Thus this paper is organized as follows. Section D.2 presents the basic concepts of kriging and the proposed WK interpolation, since they are uncommon to use for image processing and compression. Experimental results will be presented and discussed in Section D.3 and finally Section F.4 concludes this paper.

D.2 WK, chroma interpolation and quality metrics

To give optimal estimate $\hat{z}(s_0)$ to the unknown value $z(s_0)$, kriging uses linear combinations of n known sample values at points s_i around s_0 as

$$\hat{z}(s_0) = \sum_{i=1}^{n} \lambda_i z(s_i) = \lambda_0^T . \mathbf{z}$$

where $\mathbf{z}$ is the vector of n observations at primary locations and λ_0 is the vector of kriging weights λ_i chosen such that the variance $\sigma_E^2(s_0) = \mathrm{var}(\hat{z}(s_0) - z(s_0))$ is at minimum and the interpolated value is unbiased, i.e. $E[\hat{z}(s_0) - z(s_0)] = 0$.

Kriging is based on the model of the random variable $Z(\mathbf{s}) = \mu + \varepsilon(\mathbf{s})$ where μ is the constant stationary function (global mean) and $\varepsilon(\mathbf{s})$ is the spatially correlated stochastic part of variation. *Ordinary kriging* (OK) is the most common type of kriging assuming that $\mu = m$ is unknown but constant only in the s_i used to estimate s_0. For *simple kriging* (SK) $\mu = m$ is known and constant over the entire domain. In this work OK will be used as it is more suitable with natural images than SK. The unknown local mean can be filtered in the interpolation by setting $\sum_{i=0}^{n} \lambda_i = 1$.

Minimizing $\sigma_E^2(z(s_0))$ gives the optimal weights from $\lambda_{OK}^T = \mathbf{C}^T\mathbf{V}^{-1}$ where $\mathbf{V} = \mathrm{var}(\mathbf{z})$ and $\mathbf{C} = \mathrm{cov}(z(s_0), \mathbf{z})$. Second-order stationarity is assumed for OK: $\mathrm{var}(Z(s_i)) = 0$, $E[z(s_i)] = m$, and $\mathrm{cov}(Z(s_1), Z(s_2)) = \mathrm{cov}(Z(s_3), Z(s_4))$ if $|s_1 - s_2| = |s_3 - s_4| = h$, hence distance/direction dependence. This implies that $\mathrm{cov}(Z(s), Z(s)) = \mathrm{var}(Z(s)) = C(0)$ and $\mathrm{cov}(Z(s), Z(s+h)) = C(h)$ where $C(h)$ is the *covariance* of the random variable $Z(s)$. It can be shown that $\gamma(h) = C(0) - C(h) = \frac{1}{2}E[(Z(s) - Z(s+h))^2]$ where $\gamma(h)$ is the *semivariance* of $Z(s)$. *Experimental semivariance* $\hat{\gamma}(\mathbf{h})$ can be computed from

$$\hat{\gamma}(\mathbf{h}) = \frac{1}{2N(\mathbf{h})} \sum_{n=1}^{N(\mathbf{h})} [z(\mathbf{s}_n + \mathbf{h}) - z(\mathbf{s}_n)]^2$$

where $\mathbf{h}$ is the lag vector representing separation between two spatial locations, $\mathbf{s}$ is the vector of spatial sample coordinates, and $N(\mathbf{h})$ denotes the number of sample pairs separated by $\mathbf{h}$. A variogram plots $\hat{\gamma}(\mathbf{h})$ against $\mathbf{h}$ as semivariogram or correlogram, which are dual. We can fit $\hat{\gamma}(\mathbf{h})$ with some variogram models such as linear, spherical, exponential, and Gaussian. After the parameters of the matched variogram model are estimated, the semivariances at all locations can be computed to solve the kriging weights. For many images, exponential models fit reasonably well with only two parameters [Ruiz-Alzola et al. (2005)]. Figure D.2 shows the $\hat{\gamma}(\mathbf{h})$ from Y, Cb and Cr channels of 128×128 Lena image from only one of nine pixel streams.

Kriging interpolation becomes more computationally expensive as the number of samples increases due to the forming of the covariance matrix and the matrix multiplication. The number of samples in Figure D.2 is 1849, already much higher than that typically found in geostatistical problems where kriging has been mainly used. To enable kriging for images of higher resolution with controlled quality degradation, we propose the use of WK interpolation for images with segmented objects as presented in Section C.2 [Panggabean et al. (2010); Samonig (2001)].

The VQ of the resulting color images is objectively quantified by using the weighted contributions of the YCbCr color channels. We use the three variants of weighted PSNR proposed by Simone et al. (2008), namely the Weighted PSNR (WPSNR), the weighted PSNR on the Mean Square Error ($\mathrm{WPSNR_{MSE}}$), and the weighted PSNR on PIXel ($\mathrm{WPSNR_{PIX}}$) expressed as follows.

$$\mathrm{WPSNR} = w_{\mathrm{Y}}\mathrm{PSNR_Y} + w_{\mathrm{Cb}}\mathrm{PSNR_{Cb}} + w_{\mathrm{Cr}}\mathrm{PSNR_{Cr}},$$

$$\mathrm{WPSNR_{MSE}} = 10\log_{10}\frac{(2^B-1)^2}{w_{\mathrm{Y}}\mathrm{MSE_Y} + w_{\mathrm{Cb}}\mathrm{MSE_{Cb}} + w_{\mathrm{Cr}}\mathrm{MSE_{Cr}}},$$

$$\mathrm{WPSNR_{PIX}} = 10\log_{10}\frac{(2^B-1)^2}{\frac{1}{MN}\sum_{y=1}^{M}\sum_{x=1}^{N}(\mathrm{P}-\hat{\mathrm{P}})^2},$$

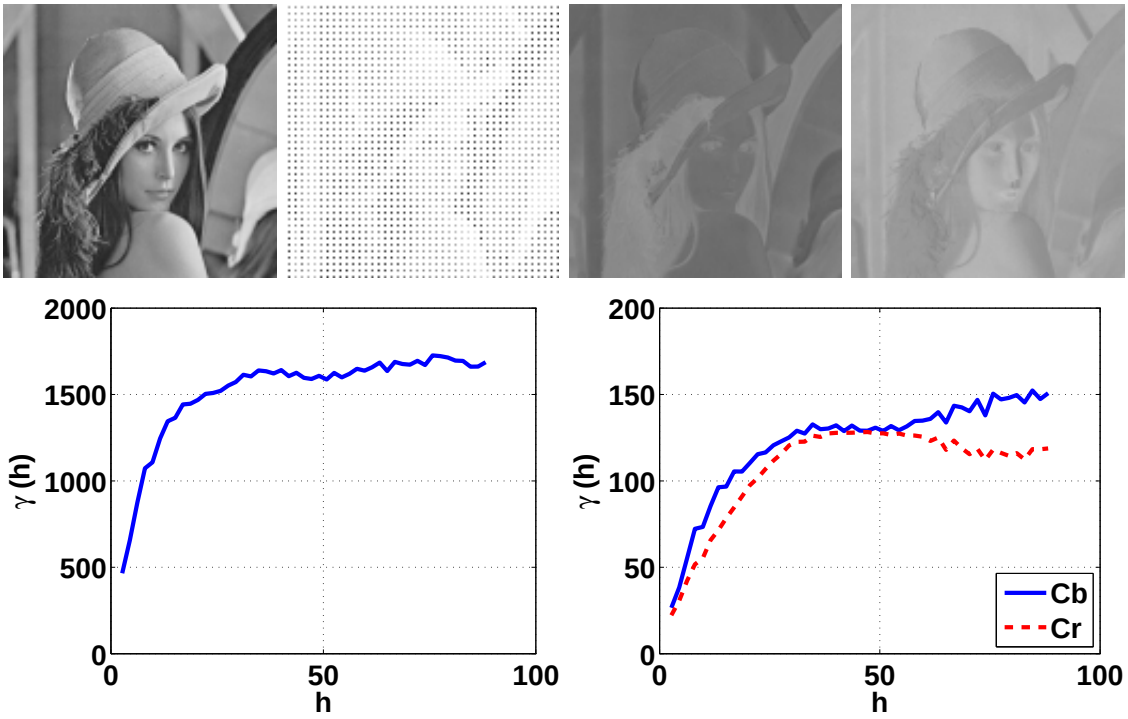


Figure D.2: Images from 128×128 Lena image, from left to right in the top row: Y channel, samples of Y pixels where **k** = [5], Cb and Cr channels; bottom row: semivariograms of the Y (left), Cb and Cr channels (right), also with **k** = [5].

where $\mathrm{P} = w_{\mathrm{Y}}\mathrm{I}_{\mathrm{Y}}(x,y) + w_{\mathrm{Cb}}\mathrm{I}_{\mathrm{Cb}}(x,y) + w_{\mathrm{Cr}}\mathrm{I}_{\mathrm{Cr}}(x,y)$ and analogous to $\hat{\mathrm{P}}(\hat{\mathrm{I}})$ where I and $\hat{\mathrm{I}}$ refer to the two images to compare. The image dimensions are denoted by $[M,N]$ and $B = 8$ represents the bit depth of the images in this work. The weights are set as $w_{\mathrm{Y}} = 0.8$ and $w_{\mathrm{Cb}} = w_{\mathrm{Cr}} = 0.05$.

D.3 Results and discussion

We present and discuss our experimental results from four true-color 768×512 test images from Kodak (2010) as depicted in Figure D.3. They are selected because of the variety and intensity of the colors. We employ the DACE MATLAB toolbox [Nielsen et al. (2009)] on a PC with a 2.99GHz processor and 3.46GB RAM. In the `dacefit` function in the toolbox, we choose `@correxp` and set `theta0` = 2 for exponential correlogram. In all cases we set **k** = [1], $S = 32$ pixels and $d = 3$ pixels following our work in [Panggabean et al. (2010)]. Since lossless coding of the pixel values is not yet considered, we define *compression ratio* $\mathrm{CR} = N^2/K$. Exemplary resulting images are to be seen on screen for best VQ for assessment by the readers.

The chroma images the four test images are interpolated using WK. The quality of the result, as depicted in Figure D.4, is objectively quantified using PSNR and *mean structural similarity index* (MSSIM) [Wang et al. (2004)] metrics with increasing N from 1 to 10. As **k** = 1, hence $\mathrm{CR} = N^2$. Table D.1 compares the VQ of the interpolated chroma images in PSNR and MSSIM from WK at CR = 4 and that from 4:2:0 sub-sampling using

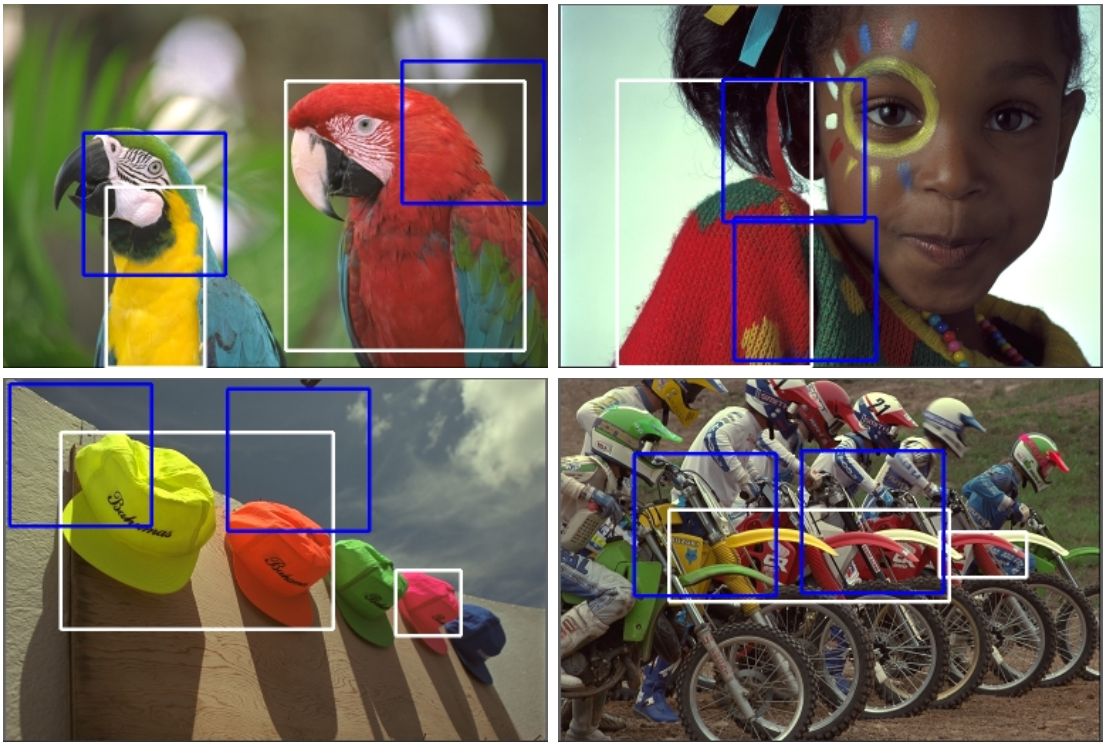


Figure D.3: True-color 768×512 test images from Kodak (2010), clockwise from top-left: BIRDS, FACE, RACE, and HATS. The rectangular bounding boxes are from the sixth column in Figure D.6 and the blue squares refer to Figure D.7.

MATLAB's bicubic interpolation, which processes 25% of image pixels. Evidently in general WK performs slightly better than 4:2:0 chroma sub-sampling using bicubic interpolation.

Table D.1: Comparing WK at CR = 4 and the 4:2:0 chroma sub-sampling with bicubic interpolation technique in the quality of the resulting chroma images in PSNR and MSSIM. The rows from top to bottom denote PSNR Cb, PSNR Cr, MSSIM Cb, and MSSIM Cr, respectively.

BIRDS		FACE		HATS		RACE	
4:2:0	WK	4:2:0	WK	4:2:0	WK	4:2:0	WK
48.69	48.18	50.55	50.87	48.48	49.28	45.49	46.16
48.27	48.11	45.46	45.97	49.71	50.04	46.59	47.22
.9924	.9963	.9926	.9965	.9929	.9964	.9807	.9915
.9907	.9958	.9788	.9915	.9936	.9970	.9847	.9930

To quantify the contribution of the interpolated chroma images to the quality of the output color images using the three weighted PSNR, we use the luminance taken from that of the output image from MATLAB's JPEG compression at quality 90/100. The arbitrary choice of coding technique for luminance is for the sake of simplicity since we focus only on the chrominance. The values of WPSNR, $\text{WPSNR}_{\text{MSE}}$, $\text{WPSNR}_{\text{PIX}}$ metrics for the four test images are plotted in Figure D.5. We can see that the chroma channels

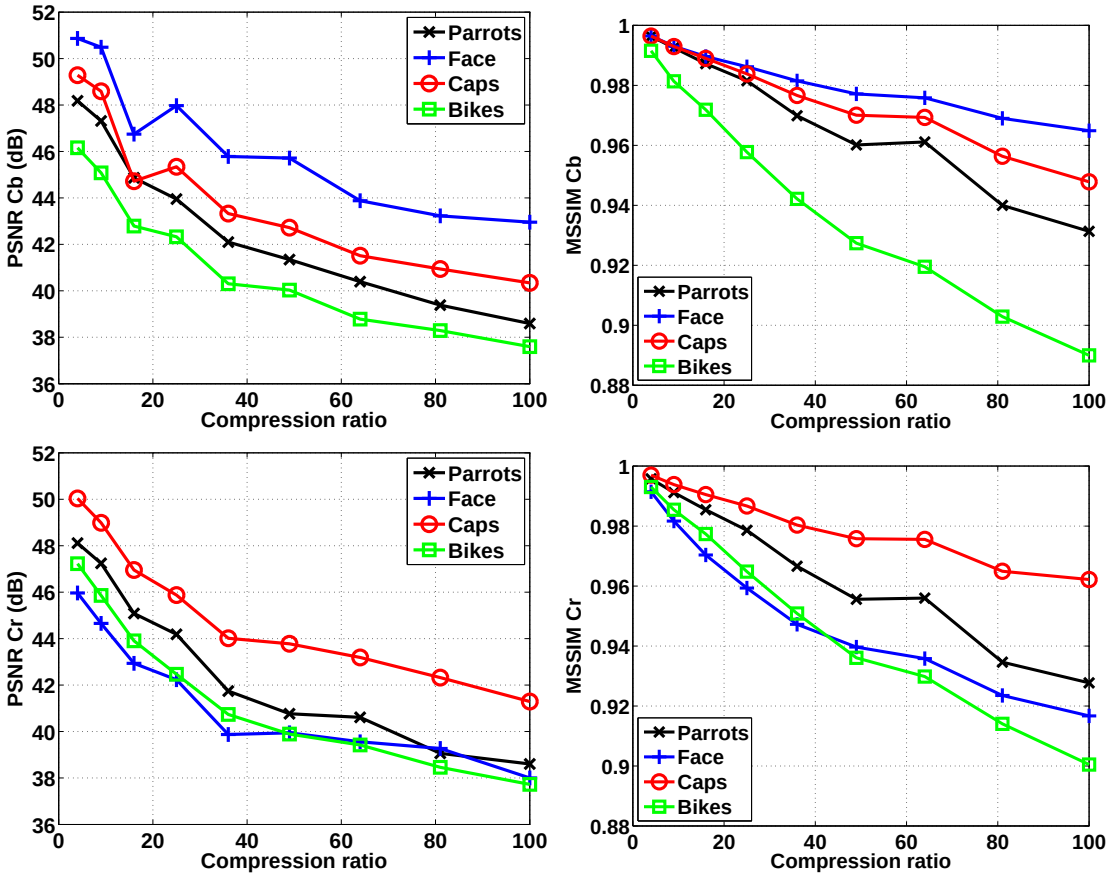


Figure D.4: VQ of Cb (top) and Cr (bottom) images interpolated by WK in PSNR and MSSIM against CR.

cause approximately 2dB reduction in each metric from the lowest to the highest CR. Observing the frequency content of the four test images reveals that RACE and FACE test images contain many edges and textures, while the other two have much less. Hence the three metrics perform consistently.

In terms of processing time, WK performs exponentially as shown in Figure D.5. As WK estimates each pixel in the image, the image resolution determines computational load rather than the number of input pixels used to compute the semivariogram. The asymptotic behavior of the computing time starting from CR = 36 also indicates this.

A sensitive segment does not have to appear in both chroma images. Thus if such segment appears in only the Cb image, then higher CR can be achieved for that segment in the Cr image, and vice versa. For example, in three binary images of BIRDS test image with colored bounding boxes in Figure D.6, one bounding box from the difference image is not present in those of the Cb image. Thus only the segment in the Cr image will be compressed with limited CR, which result is included in Figure D.7. Moreover segments in a chroma image can be ignored given their absence in the difference image after thresholding, as shown for FACE image in Figure D.6.

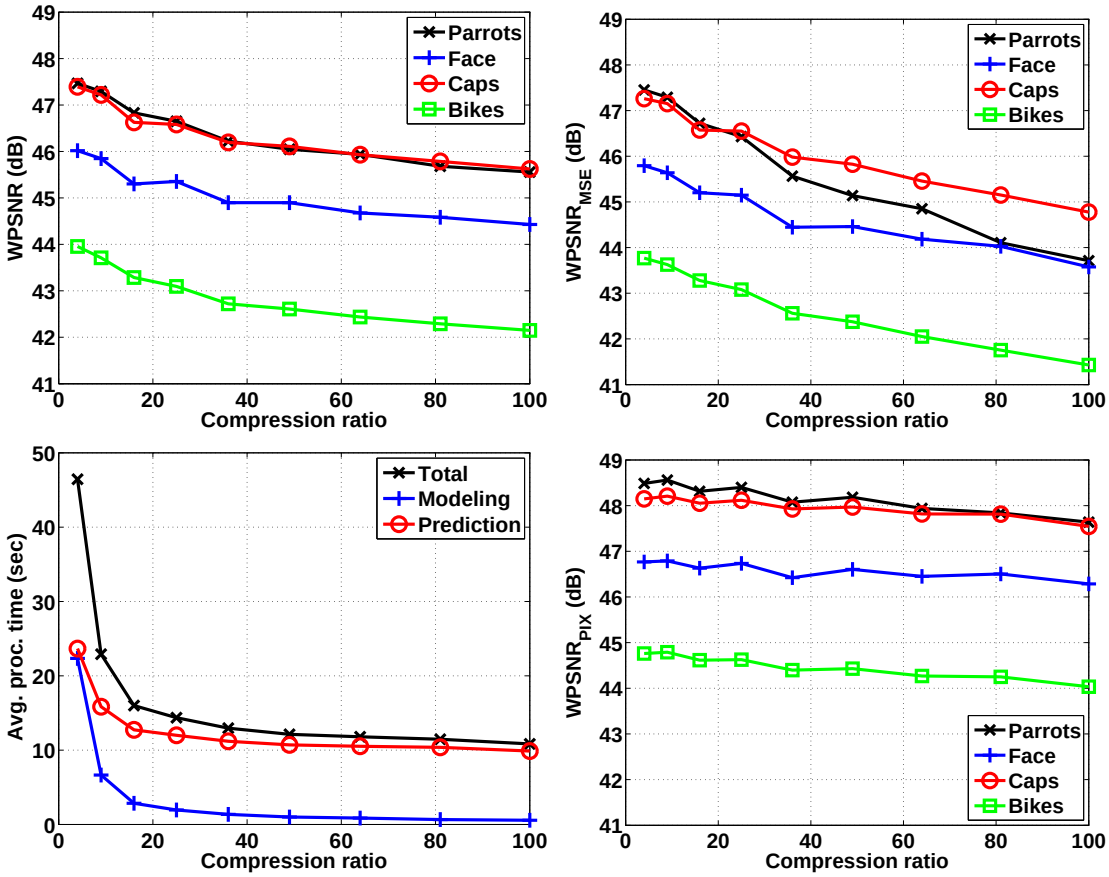


Figure D.5: Clockwise from top-left: WPSNR, WPSNR$_{MSE}$, WPSNR$_{PIX}$, and WK's average processing time against CR.

D.4 Conclusion

We have presented a study of WK for chroma interpolation that enables network nodes to reduce data with graceful quality degradation. This CbN approach moves data compression from transmitter to transmission to guarantee a maximum delay for video data which is necessary for certain applications in distributed collaborations in tele-immersive environment. We show that WK is a promising candidate for this approach. Very high CR up to 100 in areas with appropriate colors can be achieved without noticeable artifact. We also introduce the concept of color compressibility that denotes the maximum CR for a color without noticeable artifact from WK point of view. We find that areas of certain colors, especially associated with red, in chroma images are very sensitive to artifact. Fortunately such areas can be detected very fast using well-known optimal thresholding. Further studies will follow such as on subjectively quantifying color compressibility and constructing automatic fast object segmentation for chroma interpolation using WK.

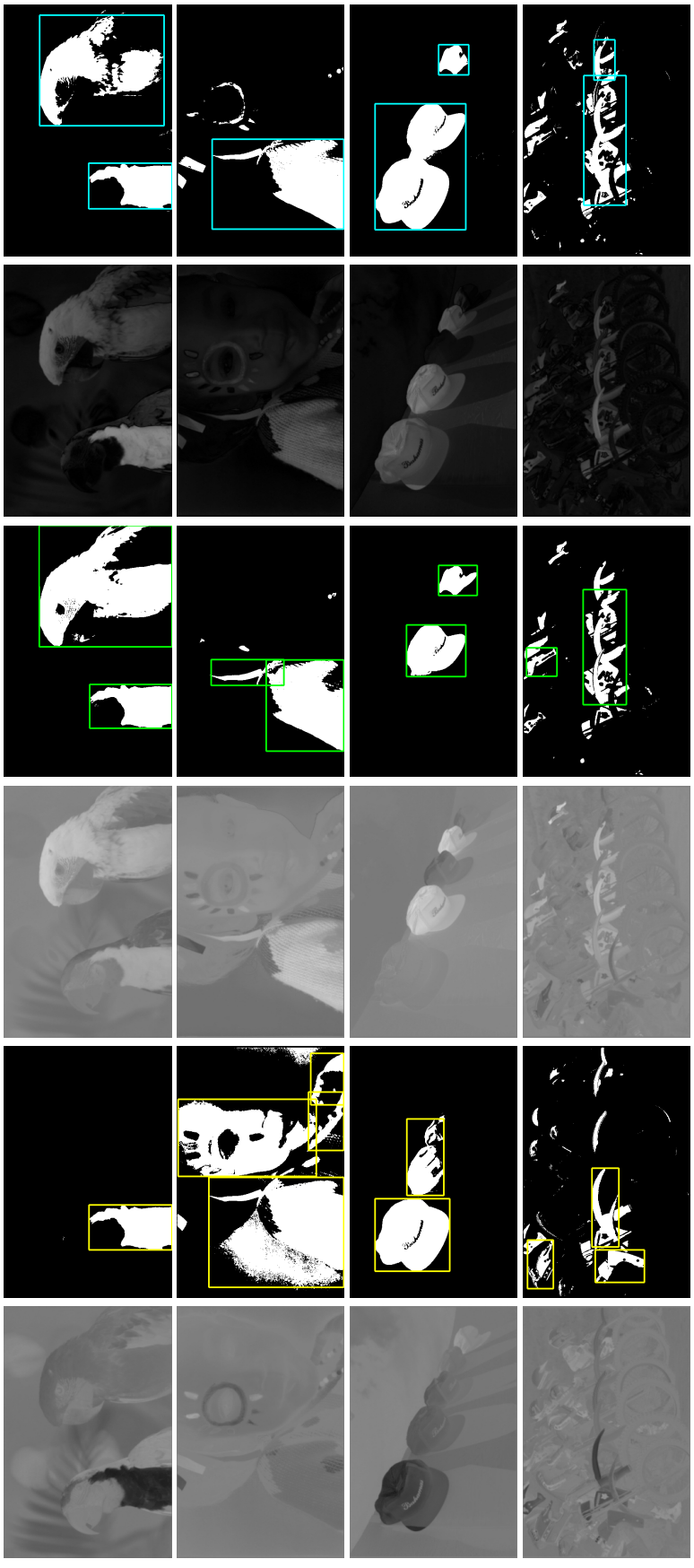


Figure D.6: Segmentation using optimal thresholding for BIRDS, FACE, HATS and RACE test images, respectively from top to bottom. From right to left: the Cb image, the segments from the Cb image, the Cr image, the segments from the Cr image, the image of absolute difference of Cb and Cr images, and the segments from the difference image. Each segment which area is greater than 2% of the image is shown with the bounding box.

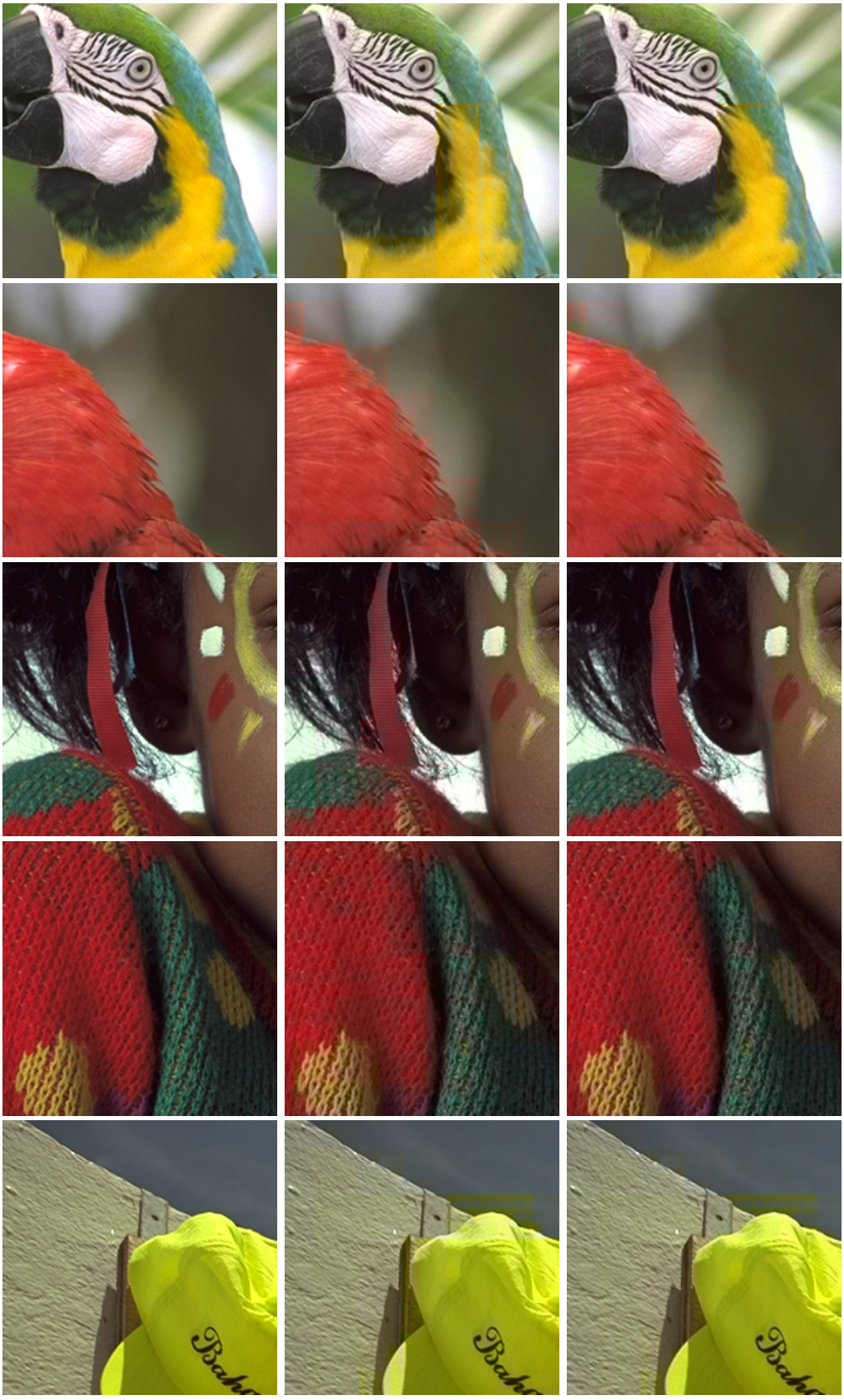


Figure D.7: Comparing the original and output images from WK as denoted by the squares in Figure D.3 for BIRDS, FACE, HATS and RACE test images, respectively from top to bottom. Each square in Figure D.3 intersects with a rectangular bounding box that comes from the corresponding image in the sixth column in Figure D.6. Each set of three images, from left to right, respectively refers to the original, the output color image which both chroma images are compressed by WK at CR = 100, and the same color image where the patched segment in the bounding box is compressed by WK at CR = 25 in the chroma image where it exists (cf. Figure D.6). Notice the different levels of block artifact that appears in areas of different colors.

References

Chafe, C., Gurevich, M., Leslie, G., Tyan, S., 2004. Effect of time delay on ensemble accuracy. In: *Proc. International Symposium on Musical Acoustics*.

Kodak, 2010. Kodak lossless true color image suite. http://r0k.us/graphics/kodak/.

Nielsen, H., Lophaven, S., Søndergaard, J., 2009. DACE, a Matlab kriging toolbox. Technical University of Denmark, www2.imm.dtu.dk/~hbn/dace/.

Panagiotopoulou, A., Anastassopoulos, V., 2007. Super-resolution image reconstruction employing kriging interpolation technique. In: *Proc. IEEE International Workshop on Systems, Signals and Image Processing*. pp. 144–147.

Panggabean, M., Tamer, Ö., Rønningen, L. A., 2010. Parallel image transmission and compression using windowed kriging interpolations. In: *Proc. IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*. pp. 315–320.

Rønningen, L. A., Panggabean, M., Tamer, Ö., 2010. Toward futuristic near-natural collaborations on Distributed Multimedia Plays architecture. In: *Proc. IEEE International Symposium on Signal Proccessing and Information Technology (ISSPIT)*. pp. 102–107.

Ruiz-Alzola, J., Alberola-Lopez, C., Westin, C.-F., 2005. Kriging filters for multidimensional signal processing. *Signal Processing* 85 (2), 413–439.

Samonig, N., 2001. Parallel computing in spatial statistics. Master's thesis, University of Klagenfurt.

Simone, F. D., Ticca, D., Dufaux, F., Ansorge, M., Ebrahimi, T., 2008. A comparative study of color image compression standards using perceptually driven quality metrics. In: *Proc. SPIE Applications of Digital Image Processing Vol. 7073*. pp. 70730Z–70730Z–11.

Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E., 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* 13 (4), 600–612.

Yang, Z., Yu, B., Wu, W., Nahrstedt, K., Diankov, R., Bajscy, R., 2006. A study of collaborative dancing in tele-immersive environments. In: *Proc. IEEE International Symposium on Multimedia*. pp. 177–184.

PAPER **E**

Ultrafast scalable embedded DCT image coding for tele-immersive delay-sensitive collaboration

Mauritz Panggabean, Maciej Wielgosz, Harald Øverby, and Leif Arne Rønningen

This paper, in the original version, has been published in *International Journal of Advanced Computer Science and Applications*, vol. 4, no. 12, 2013, pp. 202–211 (SAI, 2012 Impact Factor: 1.324).

Digital Object Identifier: 10.14569/IJACSA.2013.041230

Abstract

A delay-sensitive, real-time, tele-immersive collaboration for the future requires much lower EED for good synchronization than that for existing teleconference systems. Hence, the maximum EED must be guaranteed, and the visual-quality degradation must be graceful. DMP architecture addresses the envisioned collaboration and the challenges. We propose a DCT-based, embedded, ultrafast, quality scalable image-compression scheme for the collaboration on the DMP architecture. A parallel FPGA implementation is also designed to show the technical feasibility.

E.1 Introduction

Figure E.1 shows a simple example of the envisioned collaboration. A and B engage each other in a real-time delay-sensitive communication. They are both a source and a receiver, whereas C only receives data from them. As EED is not critical for C, C can use video-streaming technologies over the Internet. The capacity in the multihop links between A, B and C varies because other users outside the collaboration also use them. Moreover, the target QoE is so high that it closely approximates reality, i.e. near-natural.

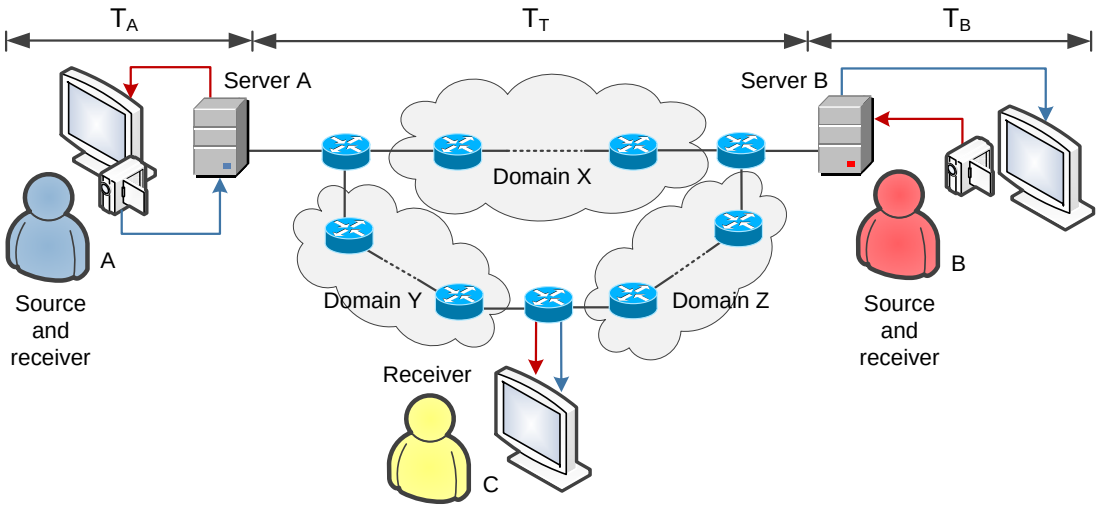


Figure E.1: A simple model of communication.

The collaboration environment is an immersive CS more advanced than the CAVE [DeFanti et al. (2011)]. To achieve the near-natural QoE, each surface in the CS is tiled with autostereoscopic multiview 3D displays with arrays of high-end cameras, microphones, and speakers. At high frame rate, the video traffic from a CS is several orders of magnitude higher than that in typical videoconferencing. It is reduced by *segmenting* only the important objects in the video, such as the faces and bodies of the performers.

A maximum EED for good synchronization between A and B must be guaranteed. Some studies show that the optimal EED for synchronizing rhythmic clapping hands

from different places is 11.5ms [Chafe et al. (2004)]. Longer delays will produce increasingly severe tempo deceleration while shorter ones yield a modest yet surprising acceleration. Since musical instruments such as percussion are rhythmically very similar to clapping hands, percussion musicians who collaborate from remote places require the same EED for synchronization. It also applies to collaborative dancing because dancers perform based on visual cues from each other [Yang et al. (2006)]. Other cases include collaborative singing and remote conducting [Rønningen and Wittner (2011)].

An EED consists of delays due to propagation, transmission, and signal processing. Propagation delay is caused by physical distances, and transmission delay depends on link capacity, queueing delay, and computations at the network nodes. The latter is the electronic bottleneck that limits the achievable capacity of a network [Tucker (2006)]. Less capacity in the network causes congestion and increases queuing delay. Instead of multipath transmission, single path is assumed to simplify routing delay. Exploiting temporal redundancy when encoding video data gives better quality but increases encoding delay. *Intraframe* video encoding is, therefore, preferred as shown by a recent experiment that uses JPEG [Holub et al. (2012)]. Since we pursue *very low latency for encoding and decoding* in the order of μs per frame, *parallel* computation must be used as much as possible.

The DMP architecture has been proposed to facilitate the envisioned collaboration [Rønningen (2011)] with the idea that maximum EED is guaranteed if each network node guarantees that the local delay never exceeds its maximum value. This value and the propagation delay can be estimated prior to packet transmission. Because the routers and switches in DMP have advanced functionalities to guarantee QoS, DMP belongs to the network-centric approach rather than the end-system-based approach [Wu et al. (2000)].

The idea has three important implications. First, a DMP network node must be able to drop parts of the video packets deliberately whenever necessary to guarantee its local delay. The dropping must be fast, and the buffer size must be optimal. Determining the latter is not the goal of this work.

Second, the packet dropping to guarantee graceful VQ degradation must be conducted intelligently. The video contents in the packets must be arranged and transmitted in decreasing order of the importance to VQ. The less important the contents in a packet, the higher the dropping priority. Packets that contain very essential contents, however, must never be dropped. This leads to the property of *quality scalability* in the wanted image-compression scheme.

Third, fast packet dropping means that it occurs in compressed domain. By providing information necessary for this in the bitstream, the cycle of decoding, dropping, and re-encoding at a node is avoided. This and the second implication mean that the bitstream can be truncated at any point to yield the reconstructed image at a lower bitrate. The quality at the final rate after dropping should be the same with that if it is encoded directly at that rate, i.e. *embedded coding* [Shapiro (1993)].

The objective of this work is to design an image-compression scheme that has all the properties aforementioned: ultrafast, embedded, quality scalable, fully parallelized, and supporting the processing of segmented objects with arbitrary shapes. Note that we do not pursue better coding performance than that of non-scalable image coding standards

because it is unfair and irrelevant. The envisioned collaboration allows for sub-optimal VQ as the price for guaranteeing maximum EED as long as the VQ is gracefully degraded. Tradeoff is normal in image/video coding. For instance, H.264/MPEG-4 AVC [ITU-T (2003)] and x264 [x264 (2013)] provide profiles and presets to meet various priorities such as low complexity or high performance.

This paper is structured as follows. Section E.2 details the proposed image compression technique. Experimental results follow in Section E.3 with discussion and analysis. Section E.4 discusses the complexity of the algorithms for implementation on field-programmable gate array (FPGA). Section F.4 concludes the paper with summary and further ideas.

E.2 The proposed image-compression scheme

The DMP approach resembles the concept of layered coding such as in scalable video coding (SVC) [Ohm (2005); Schwarz et al. (2007); Sun et al. (2007)] and JPEG 2000 [Schelkens et al. (2009); Taubman and Marcellin (2001); Taubman (2000)]. SVC achieves temporal, spatial, and quality scalability by removing parts of the video bitstream to adapt it to different end-users' preferences and varying terminal capabilities or network conditions. Proposed to supersede JPEG, JPEG 2000 is an image compression standard and coding system based on wavelet transform. Some of the improvements over JPEG are as follows: superior compression performance, multiple resolution representation; progressive transmission by pixel and resolution accuracy; spatial, quality and channel scalability; support of lossless and lossy compression; embedded coding; facilitated processing of regions of interest; error resilience. Consequently, they make JPEG 2000 more complex and computationally demanding.

The properties aforementioned make the proposed image-compression technique (Figure E.2) somewhat different from the existing ones. For example, the quantization, a key step such as in JPEG image compression (Figure E.3), is the principal cause for the loss of information, while the loss in DMP is due to the deliberate packet dropping at network nodes, i.e. the proposed scheme has no such quantization. Moreover, the techniques optimize bandwidth utilization by aiming for the best VQ at a given bitrate with no guarantee over maximum EED.

E.2.1 Block ranking and transform

The encoder of the proposed scheme consists of three major steps: block ranking, transform, and entropy coding (variable length encoding, VLE, and run-length encoding, RLE). After an input picture is divided into $N \times N$ blocks and the color space is converted to YCbCr, the block ranking automatically classifies each block into one of several ranks according to how important the block contents are to VQ. The importance of a block is indicated by the level of distortion to human perception when the binary representation of the block content becomes less precise, e.g. via quantization or packet dropping. The blocks are independent from each other, and thus can be processed concurrently. In this work $N = 8$ pixels, and users can define their own ranking method.

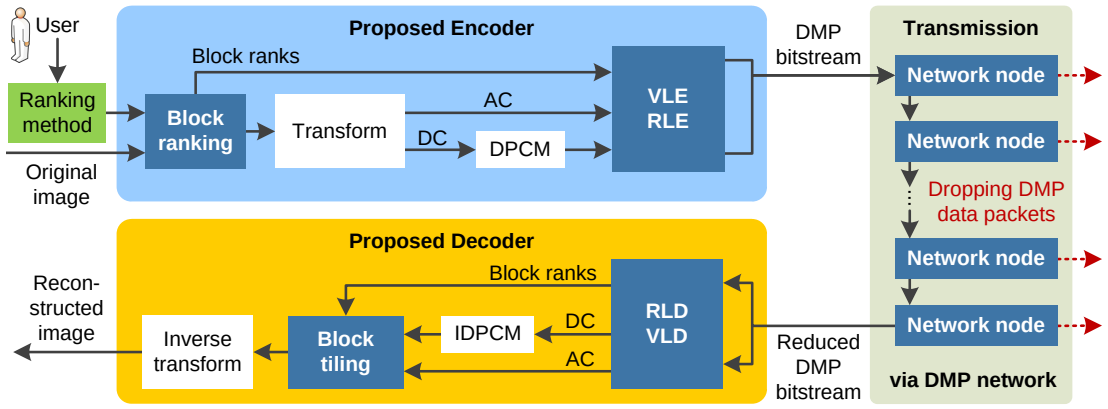


Figure E.2: The proposed image-compression technique.

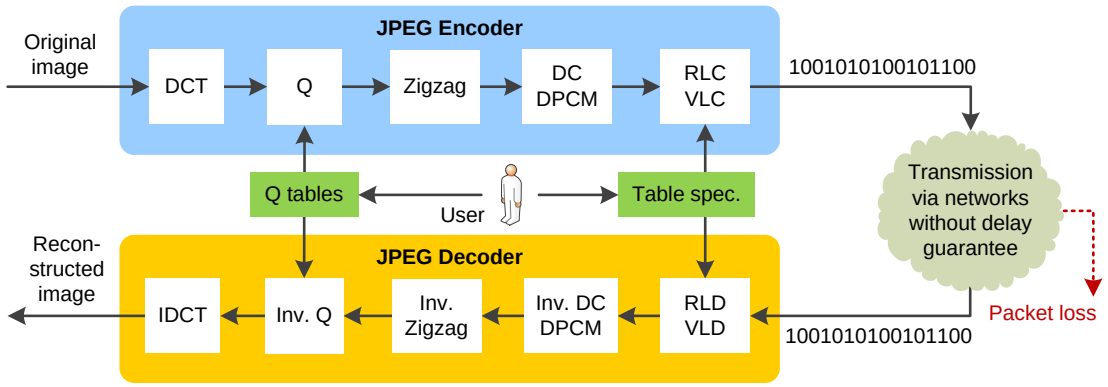


Figure E.3: Block diagram of JPEG image compression.

For the transform, two dimensional DCT (2D-DCT) [Ahmed et al. (1974)] is selected for two main reasons. First, it is widely used because of the excellent energy compaction. Second, many fast hardware (HW) implementations of 2D-DCT have been reported, e.g. in [Tumeo et al. (2007)]. The most recent work closest to ours is that by Van der Vleuten et al. (2000), which incorporates quality scalability to JPEG by encoding the DCT coefficients bit-plane by bit-plane, starting at the most significant one. Although the performance is similar to that of JPEG without quantization or entropy coding, the algorithm, particularly the scan order, must be adapted to each image. Our scheme is agnostic to the input image.

The block ranking can be applied before or after the block transform. The first only has 64 pixel values of the block luma available for analysis and ranking, whereas 64 DCT coefficients are additionally present in the latter. We choose the first option because various statistical properties of pixel values have been used for content classifications in images [Zhang and Tan (2002); Chang et al. (2006)]. Furthermore, luma values are integers, but DCT coefficients use floating points. Therefore, computing pixel values requires less resources and time than if DCT coefficients are added to the computation. Moreover, DCT coefficients in natural images are more complex to use for classification purposes [Sorwar et al. (2001)].

The statistical measures for ranking the blocks must be highly accurate and fast to compute. We use the entropy E, which measures the amount of information and uncertainty contained in data [Shannon (1948)]. For a grayscale image with N unique pixel values, it characterizes the texture therein as

$$E = -\sum_{i=1}^{N} p_i \log_2 p_i$$

where p_i is the probability of the ith pixel value from the histogram counts. The block entropy BE rises when the frequency content of the block increases.

Shown in Figure E.4 for LENA image using 8×8 blocks, the BE values are between 2 and 6 in all images tested (Figure E.5). Since the colors correspond well with human perception, BE is a good indicator of the frequency content in a block. The constant range of BE can be used to define the thresholds for arbitrary number of block ranks for dropping. We use the following four block ranks with the ranges: low ($\lceil BE \rceil \leq T_L$), low-medium ($T_L < \lceil BE \rceil \leq T_{LM}$), medium-high ($T_{LM} < \lceil BE \rceil \leq T_{MH}$), and high ($\lceil BE \rceil > T_{MH}$), where T_L, T_{LM} and T_{MH} are thresholds in positive integers.

The criterion for the high-frequency rank is not very accurate because some of the blocks are grouped into the medium-high rank. It leads to the use of block variance BV to improve the block-ranking accuracy. The variance of a grayscale image with M pixel values is given by

$$V = \frac{1}{M-1} \sum_{i=1}^{M} (x_i - \hat{x})^2$$

where x_i denotes the intensity value of the ith pixel, and $\hat{x}$ is the average of all the pixel values. In the proposed block-ranking algorithm (Algorithm E.1), $\lceil x \rceil$ rounds the scalar x to the nearest integer towards plus infinity, $T_V = 1$, $T_L = 3$, and $T_{LM} = 4$. The resulting BV values for LENA image are shown in Figure E.4. The BE and BV are not only fast to compute in HW (Section E.4), but also correspond well with human perception (Section E.3).

Algorithm E.1 Proposed algorithm for block ranking

```
if ⌈BV/100⌉ ≤ T_V then
    if ⌈BE⌉ ≤ T_L then
        Rank 4: low frequency (blue)
    else if T_L < ⌈BE⌉ ≤ T_LM then
        Rank 3: low-medium frequency (green)
    else if ⌈BE⌉ > T_LM then
        Rank 2: medium-high frequency (yellow)
    end if
else
    Rank 1: high frequency (red)
end if
```

Encoded and included in the bitstream as side information, the produced block ranks must never be lost because it will jeopardize the image reconstruction from the transmitted packets at the receiver. They are also used in structuring the encoded DCT coefficients into the data packets to enable packet dropping in the compressed domain.

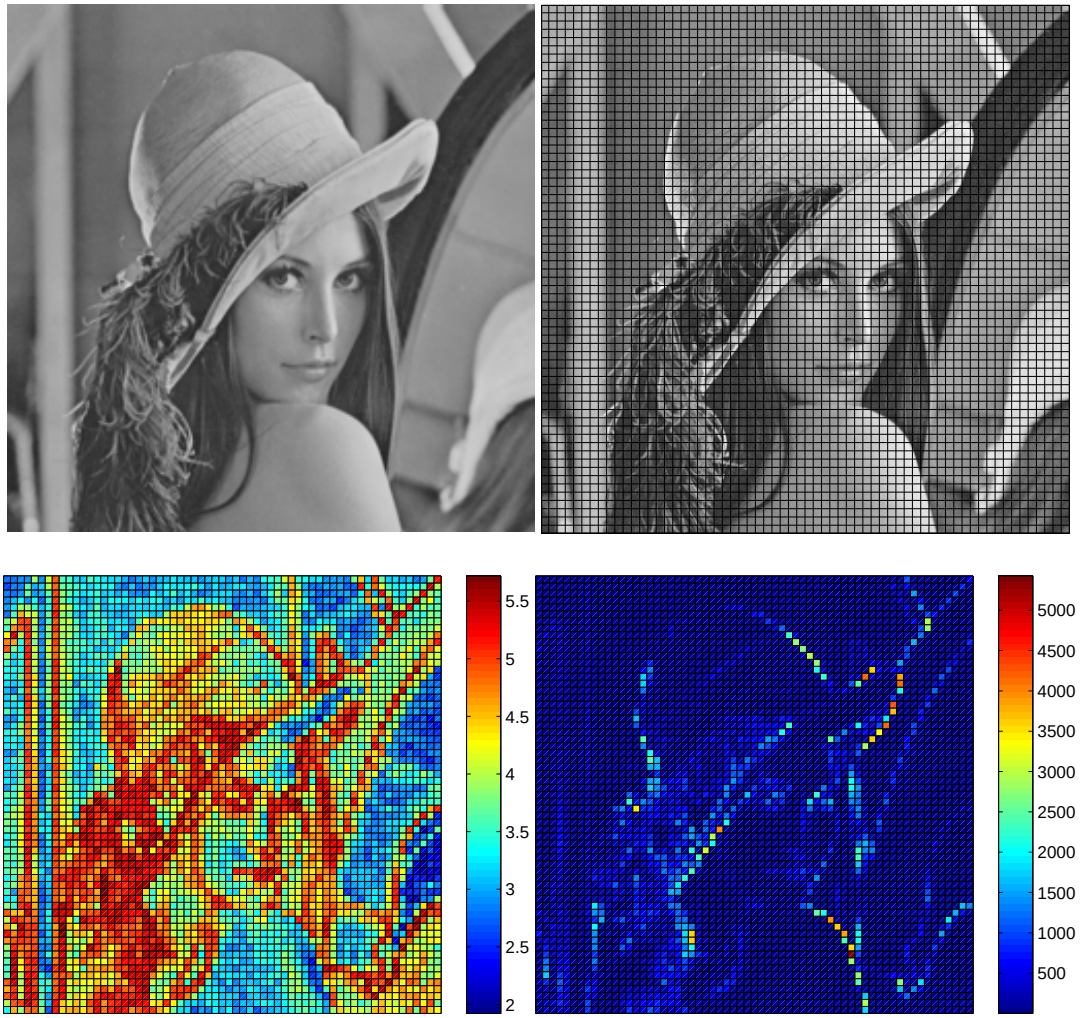


Figure E.4: Clockwise from top left: the original 512×512 LENA image, 8×8-blocks tiled on the image, the variance and entropy of the blocks.

After the block ranking, 2D-DCT is applied to each of the luma block independently, and it produces 64 DCT coefficients per block, which comprise one DC coefficient and 63 AC coefficients. As the DC coefficient contains the average value of the block, it is essential for reconstruction and must not be dropped. The location of an AC coefficient in a block indicates the importance. The only information loss in the proposed encoder, rounding the values to the nearest integers reduces the precision for faster computation with less memory use.

E.2.2 Universal codes for entropy coding

The distribution of the rounded DCT coefficients is key in encoding them losslessy and efficiently. Figure E.6 depicts the empirical probability density functions (PDFs) of the rounded AC coefficients between -10 and 10 from DCT and Walsh-Hadamard transform (WHT) for comparison. They include more than 99% of all the coefficients for WHT in all the test images, but it is between 80% and 99% for the DCT. Note the

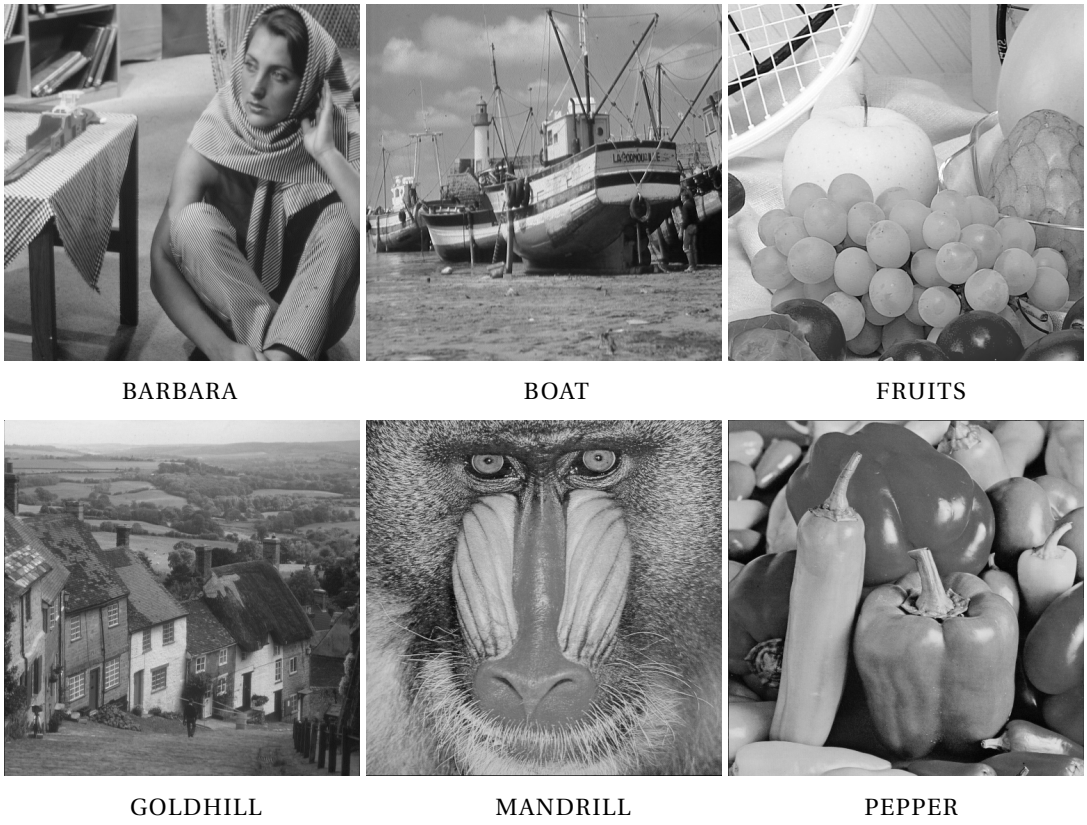


Figure E.5: The test images besides LENA.

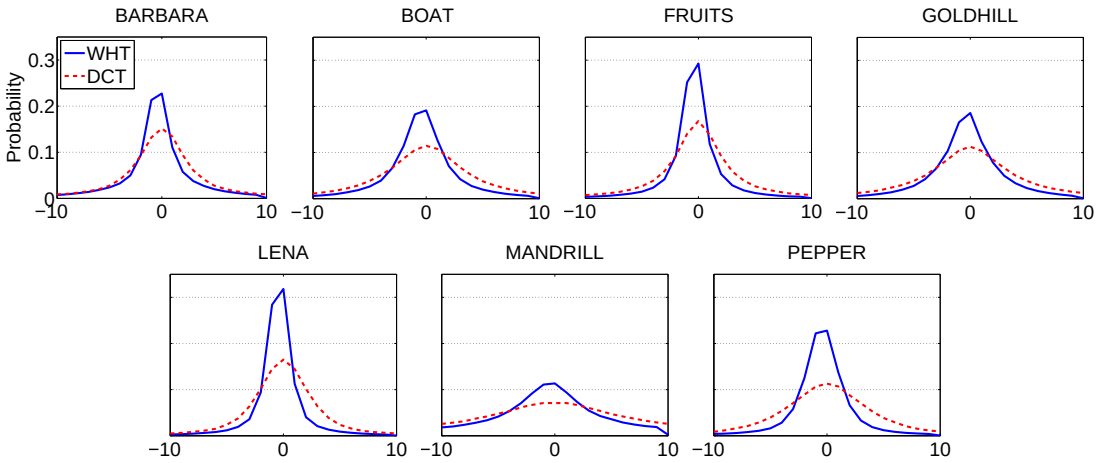


Figure E.6: Empirical PDF of AC coefficients from DCT and WHT for the test images.

symmetry around zeros in the PDFs. The quality of the reconstructed images using DCT in peak-signal-to-noise ratio (PSNR) is always a few dB higher than that for WHT due to the rounding of the DCT coefficients.

The probability of zeros is always less than 0.20 for DCT and slightly higher for WHT. Producing shorter average length of codewords with higher probability of zeros [Solomon and Motta (2010)], the Huffman code is not efficient for this case. Moreover,

the distribution of the DCT coefficients could not be known beforehand. The Huffman code is favored when more zeros are present in the high-frequency coefficients caused by stronger quantization.

The dropping of the DC coefficients starting from the least important implies that RLE is not suitable for this work because it introduces more dependencies in the resulting bitsream. The run lengths of the coefficients can also be very short because of no quantization. This is a challenge because RLE can increase performance in lossless coding.

Furthermore, coding techniques that can only be decoded when the bitstream is complete, such as the Burrows-Wheeler transform (BWT) [Burrows and Wheeler (1994)], are also not suitable. It is, in fact, impossible because the received stream at the end are truncated due to dropping. Nevertheless, they are useful for encoding the block ranks and the rounded differences after applying differential pulse-code modulation (DPCM) to the DC coefficients.

Excellent texts such as [Solomon and Motta (2010)] comprehensively discuss and compare various coding techniques available. They lead us to the use of universal codes (UCs) for entropy coding for the applicability regardless of the data distribution. We propose using the Fraenkel and Klein C^1 Fibonacci code [Fraenkel and Klein (1996)] (FK_1) based on the comparison of well-known UCs in [Fenwick (2003)]. The recurrence relation $F(i) = F(i-1) + F(i-2)$ with seed values $F(0) = 0$ and $F(1) = 1$ defines the sequence $F(i)$ of the famous Fibonacci numbers. The Zeckendorf's theorem states that any integer can be formed as the sum of Fibonacci numbers [Vajda (1989)]. Thus, for a positive integer number n, if $d_0, d_1, ..., d_k$ represent n, then we have $n = \sum_{i=0}^{k-1} d_i F_{i+2}$ and $d_k = d_{k-1} = 1$ where F_i is the ith Fibonacci number. A Zeckendorf representation $Z(n)$ is coded by writing a binary vector with a 1 wherever that Fibonacci number is included, but F_1 is omitted due to redundancy. For example, since $19 = 13(F_7) + 5(F_5) + 1(F_2)$, it means $19 = (1 \times 13) + (0 \times 8) + (1 \times 5) + (0 \times 3) + (0 \times 2) + (1 \times 1)$ which gives $Z(19) = 101001$.

A very important property of $Z(n)$ is that two adjacent 1's never occur. Therefore, the FK_1 code produces $FK(n)$ by writing $Z(n)$ in the reverse order and appending another 1 as a terminating comma; hence, $FK(19) = 1001011$. Decoding n from $FK(n)$ is straightforward and only involves additions, making it fast for HW implementation. Using the code for signed integers is possible after *bijection*, i.e. mapping the real values in signed integers into symbols in positive values. Table E.1 shows the FK_1 codewords of the symbols n from applying bijection to the real values x.

Table E.1: Some examples of $FK_1(n)$ for symbols from real values after bijection

x	n	$FK(n)$	x	n	$FK(n)$
0	1	11	3	6	10011
1	2	011	-3	7	01011
-1	3	0011	4	8	000011
2	4	1011	-4	9	100011
-2	5	00011	...	...	...

The FK_1 code offers several advantages [Fenwick (2003)]. First, unlike using adaptive parametrized codes, storing tables of codewords in the network nodes for packet dropping is unnecessary; hence, more efficient use of resources. Second, using two 1's as the delimiter between consecutive coded symbols gives more robustness against transmission errors than table-based codes such as the Huffman code. Third, because of the universal codewords, simply reading from lookup tables (LUTs) allows fast encoding and decoding. Fourth, the memory allocated for the LUTs is also very small (Section E.4). Fifth, no prefix code used also means higher efficiency.

The encoding strategy for the DCT coefficients and the block ranks is proposed as follows. First, the block ranks are encoded in the raster fashion using BWT because only four integer symbols are used, which saves around 18% than using 2-bit binary encoding. Using run lengths gives very little gain, merely around 0.01 kilobytes. The BWT is currently the best lossless compression technique, especially for text, with fast implementations available [Solomon and Motta (2010)] such as the *bzip2* [Seward (2012)].

Second, the AC coefficients are processed and encoded separately from the DC. Prior to encoding, DPCM and bijection are applied to the signed coefficients. The resulting symbols for the DC coefficients are then encoded into binary string using the FK_1 code. The bitstream from the block ranks and the DC coefficients must not be dropped.

Third, the 63 AC coefficients from each block are grouped into 63 series according to their position, following the zigzag direction as used in JPEG. The symbols after bijection are encoded in parallel using the FK_1 code. The symmetry of their distribution (Figure E.6) motivates the use of bijection. The coefficients starting from the most top-left block are transmitted first in the raster fashion, and the series are sent according to their series number.

If the probability of zeros in the resulting binary string is very high, i.e. higher than 0.9, the run lengths of ones and zeros can be encoded further, for example, using the Golomb codes [Golomb (1966)]. In all the test images, however, the probability is only around 0.7. The bitstream from the Golomb code must be decoded before dropping. Since the reduced bitstream after dropping must be encoded again using the Golomb code, processing time at the network nodes increases.

E.2.3 Data structure and packet format

The data structure also affects the coding. The proposed data structure depicted in Figure E.7 can be used to arrange the blocks, ranks and coefficients for encoding. Since all the AC coefficients of the same rank and index are grouped together, dropping them as a group when necessary is straightforward. Packet dropping is fast because checking each block rank for dropping is not needed. Moreover, since the bitstream of the group is long, more compression gain can be achieved using Golomb codes on the run lengths of the zeros. The proposed HW design and implementation of the DMP network node also obtain higher throughput with longer input bitstream. This area opens many interesting questions for future work. For this work we use fixed-length packets.

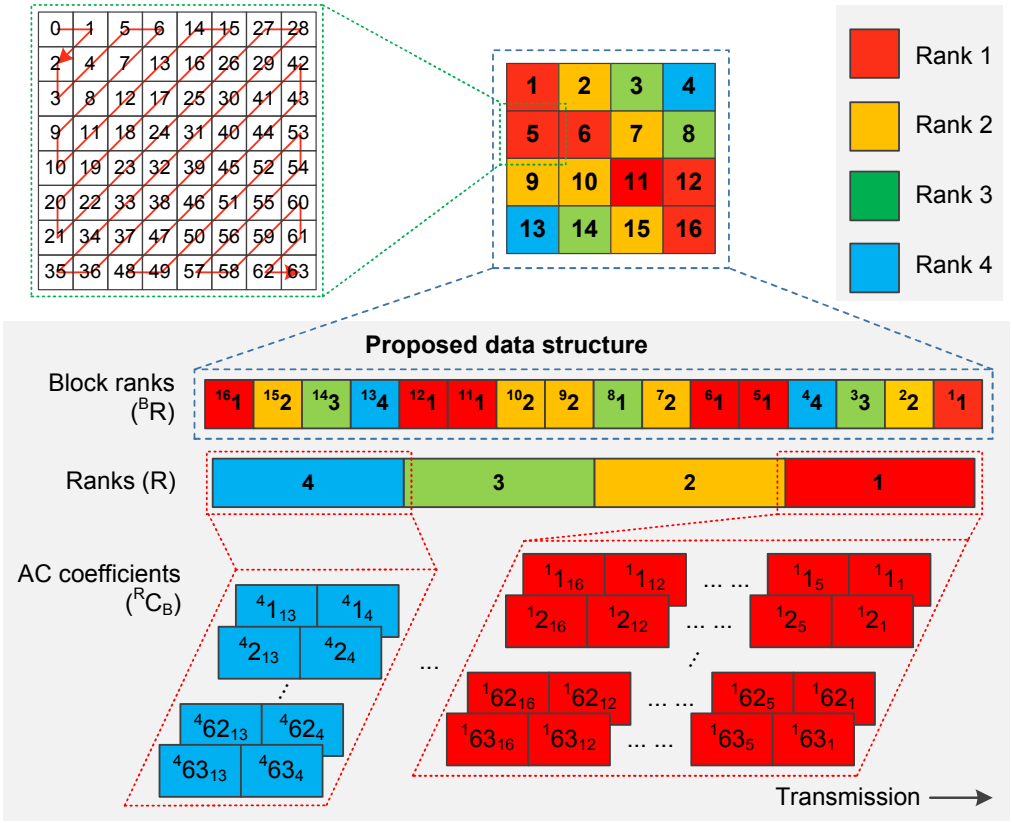


Figure E.7: Proposed data structure for packetization and transmission of the encoded blocks, ranks, and DCT coefficients.

E.3 Results and discussion

Seven standard grayscale test images are used in the experiments. The results are produced using MATLAB, and the *bzip2* codec [Seward (2012)] is used for the BWT-based compression. All images exhibited in this section should be seen with magnification on screen for the best perceptual quality.

The first two images in Figure E.8 depict the block maps of LENA image using only entropy and that using the proposed block-ranking method. Both maps have the same blue and green blocks, but not those in the other two colors. There are more red blocks in the second image, for example on the edges and in the area of the fur. This illustrates the importance of the block variance in ranking the blocks. It produces more red blocks because they are more sensitive to visual distortion.

Figure E.8 also displays four sets of areas according to the four ranks produced by the proposed ranking method. They show the high classification accuracy of the proposed ranking method. Different techniques to classify the contents can be employed, for example using edge and texture detection to detect the edges and the textured areas. This idea, however, is not necessary because the blocks containing them can be successfully categorized as those in red by the proposed method.

The distribution of the four block ranks in the test images as shown in Figure E.9

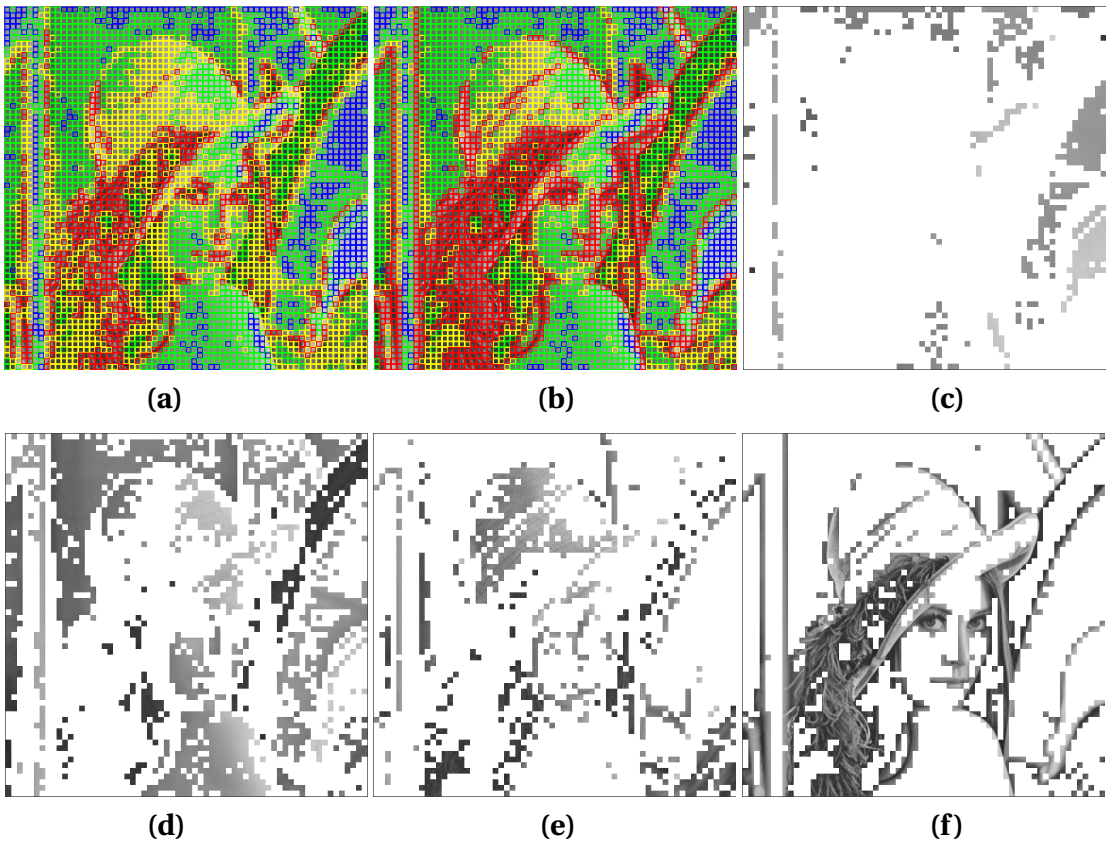


Figure E.8: The four-rank block map of LENA image using only entropy (a) and that using the proposed ranking method (b). The image is decomposed into five ranks as follows (with decreasing dropping priority): low frequency in blue (c), low-medium in green (d), medium-high in yellow (e), and high in red (f). Borders are added for better view.

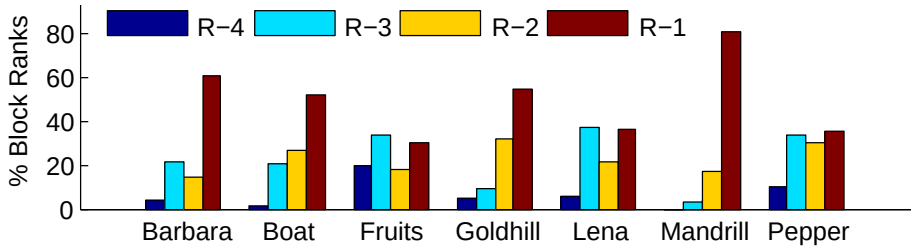


Figure E.9: The distribution of the four ranks in the test images.

indicates that all the rank maps of the images correspond well with human perception. It can be checked by visually comparing the distribution with the original images in Figure E.5. Rank-1 blocks are dominant in BARBARA, BOAT, GOLDHILL, and especially MANDRILL due to many textured areas present therein. In the other images, the portions of the blocks of Rank 1 and Rank 3 are almost the same because of the flat areas (Rank 3) and the edges (Rank 1).

Some examples of the rank map and the reconstructed images are shown in Figure E.10 from PEPPER image. Figure E.10 (b) is reconstructed from the received bitstream

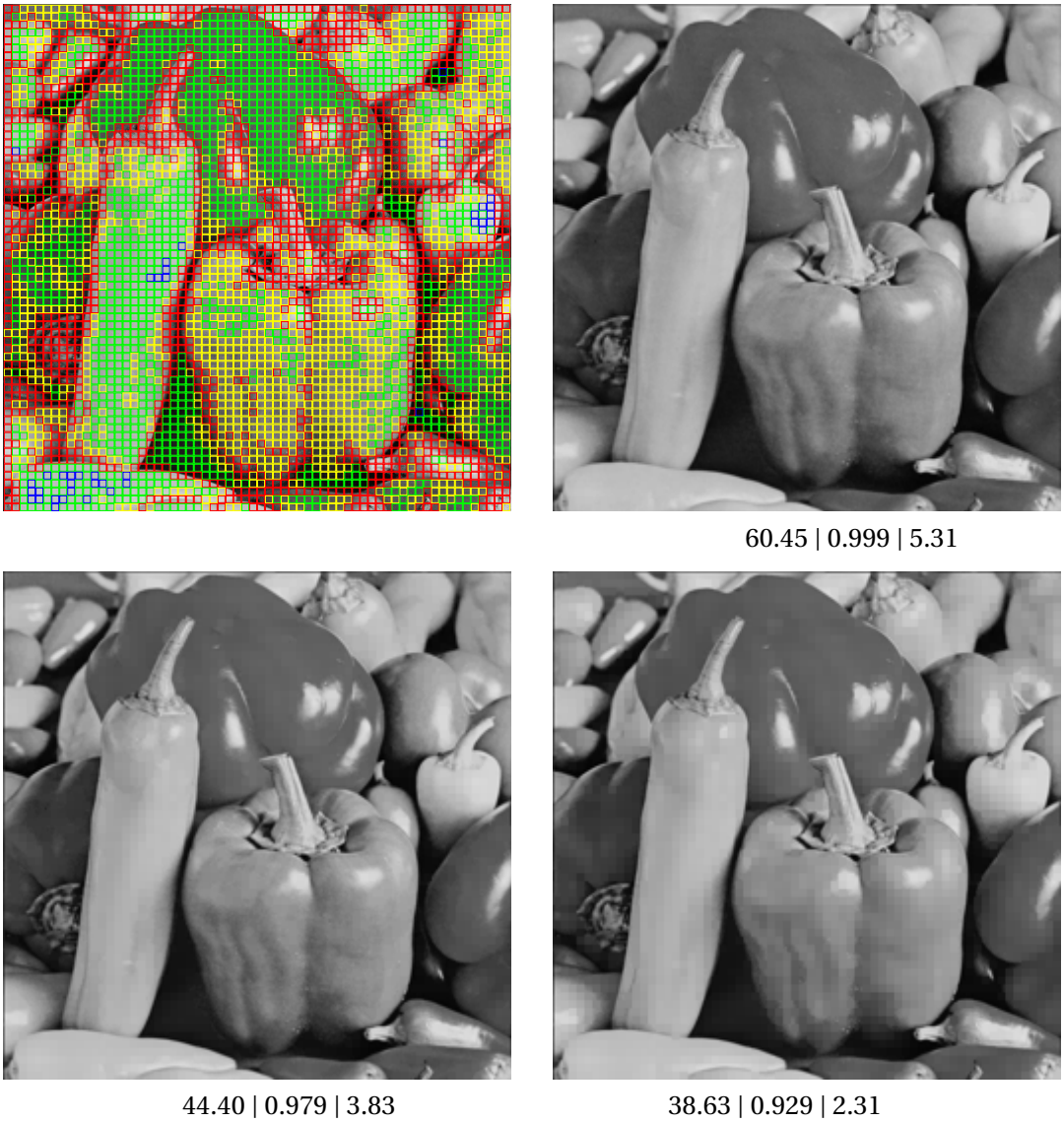


Figure E.10: The examples from PEPPER image: the rank map with entropy and variance (a); the images reconstructed without the DCT coefficients from Rank 4 (b), from Ranks 4 and 3 (c), and from Ranks 4, 3, and 2 (d). The PSNR (dB), MSSIM and bitrate (bpp) are provided underneath.

without the DC coefficients of Rank 4 because they have been dropped completely. When the network capacity is reduced, the nodes start dropping the AC coefficients of Rank 3 beginning from those of the 63th index. When all the AC coefficients of Rank 3 have been dropped, the resulting image quality is shown in Figure E.10 (c). The dropping is continued until the worst quality is achieved by reconstructing only from the DC coefficients of all ranks (Figure E.10 (d)). The results are accompanied with the bitrate in bits per pixel (bpp) as well as the PSNR and the MSSIM [Wang et al. (2004)] as the objective VQ metrics. Furthermore, the RD plots using the two quality measures for the test images are shown in Figure E.11. The plots from JPEG are also provided for comparison.

The performance of the proposed scheme can be improved by using deblocking

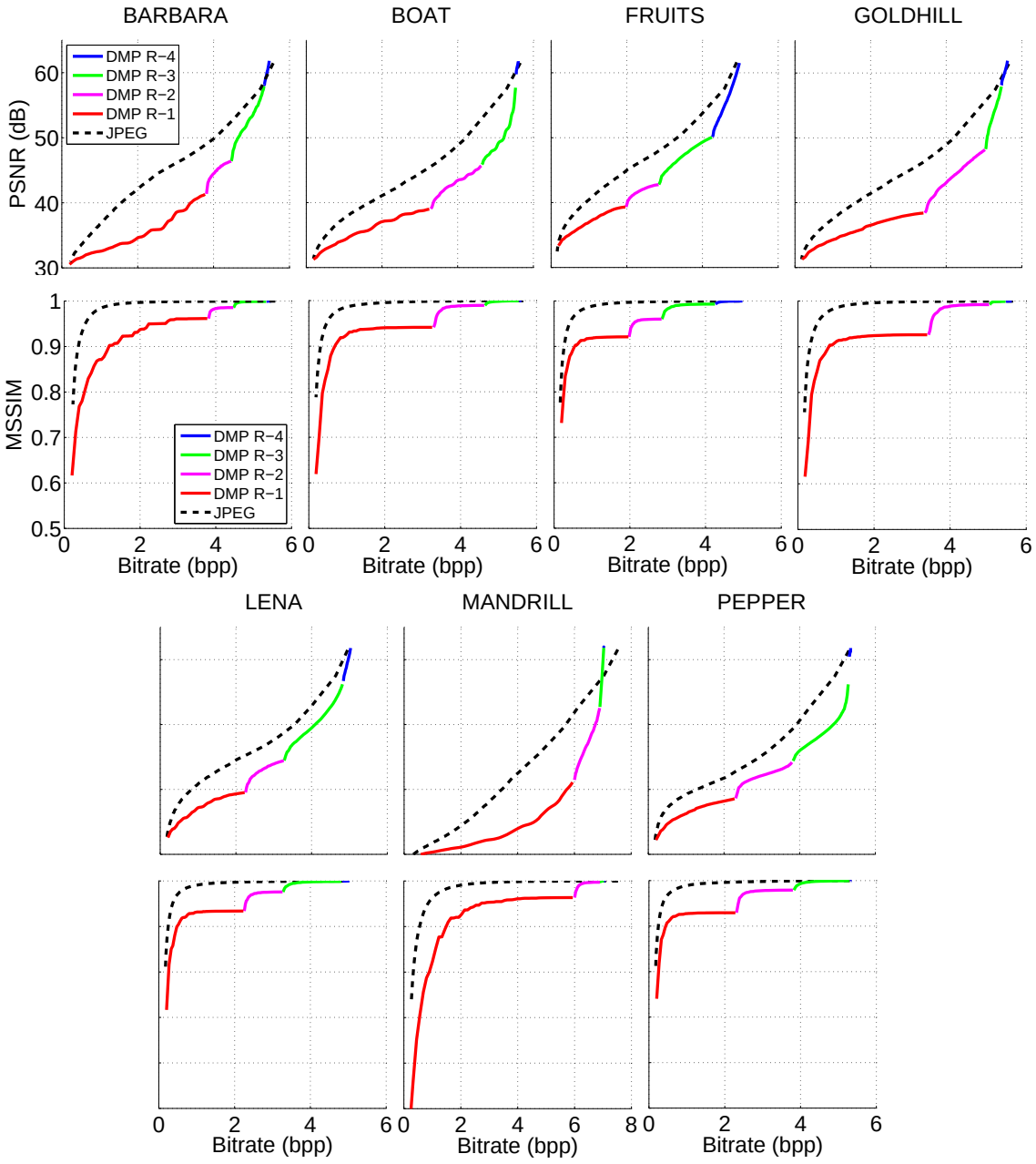


Figure E.11: RD plots of PSNR (top) and MSSIM (bottom) against bitrate.

filter as post-processing step at the receiver. An important role for estimating the bitrate, this step is conducted at the source in many image/video coding techniques. Other possible improvements include intra-prediction, which is not considered here because it creates dependencies between the adjacent blocks and increase the complexity. Note again that we do not aim at better coding performance than that of JPEG or even JPEG 2000.

Figure E.10 shows that the reconstructed images suffer from blocking artifacts that occur only at the blocks which AC coefficients are dropped. The artifact, however, is different from the typical blocking artifact in JPEG because the latter occupies much

larger areas consisting of many blocks. The distortion is called *pixelation artifact* due to the resemblance to it. The encoded rank maps in the side information of the bitstream plays another important function; they inform the receiver of the exact locations of the pixelated blocks. Thus those blocks can be directly restored without searching their locations as in typical deblocking algorithms.

For the worst distortion because all the AC coefficients are discarded, a fast depixelization algorithm is proposed in Algorithm E.2 which refers to Figure E.12. Figure E.13 shows some examples of the depixelization for FRUIT and PEPPER images with PSNR and MSSIM values. The algorithm successfully restores the flattened blocks to be more appealing to human perception.

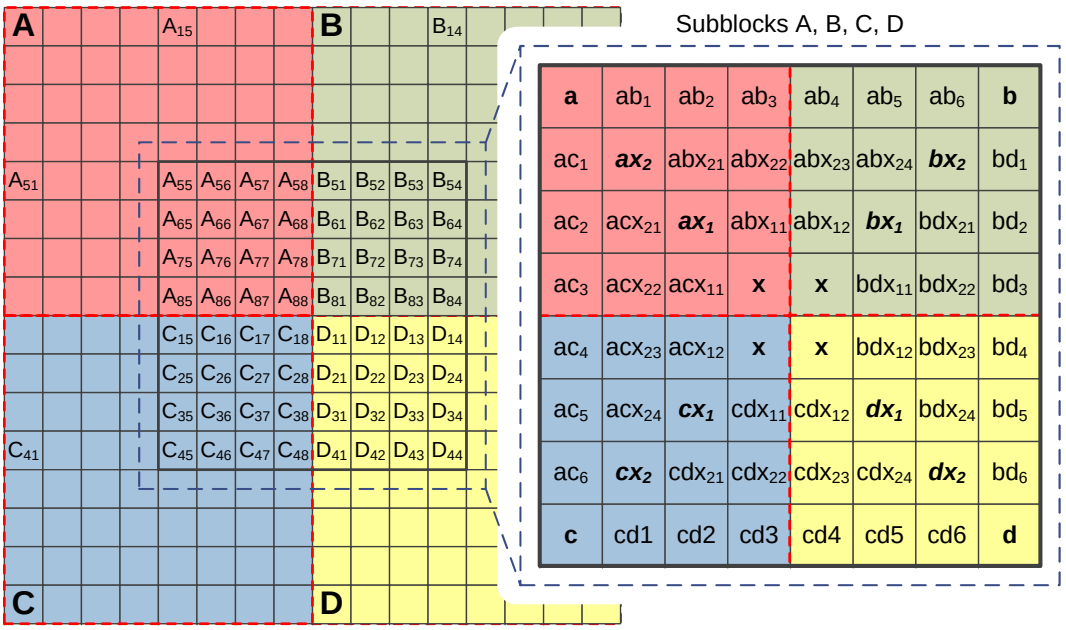


Figure E.12: The proposed depixelization in Algorithm E.2.

The blocks of Rank 2, 3 and 4 can be reconstructed only from the DC coefficients without pixelated blocks because they can be repaired fast using Algorithm E.2 (Figure E.13). In fact, Rank-1 blocks with strong textures and no edges can be made free from pixelation. Nevertheless, the artifacts are still visible in Rank-1 blocks with edges even after depixelization. Repairing the pixelated edges can benefit from more advanced techniques such as in [Kopf and Lischinski (2011)].

Figure E.14 (left) shows a collaborating person as a segmented object from a video frame of an HD test video sequence from [TGFX (2012)]. It is expected to be produced by the cameras on a surface of a CS. Applying the proposed block ranking algorithm produces the blocks in Figure E.14 (right). The blocks of the background can be assigned an additional rank, for example, Rank 5. They contribute an insignificant increase in bits (much less than 1 Kbits) and their DCT coefficients are all discarded. This illustrates that the proposed scheme can encode regions-of-interest with arbitrary shapes.

Algorithm E.2 A depixelization algorithm for the worst distortion

1: *Reference elements of the subblocks A, B, C and D*
2: $X \leftarrow (A_{88} + B_{81} + C_{18} + D_{11})/4$
3: $a \leftarrow A_{55}$, $b \leftarrow B_{54}$, $c \leftarrow C_{45}$, $d \leftarrow D_{44}$
4: **for** Block $M = \{A, B, C, D\}$ **do**
5: $\{m, mx_1, mx_2, X\} \leftarrow \mathrm{LI}(m, X, 2)$
6: **end for**
7:
8: *Non-reference elements of the subblocks A, B, C and D*
9: **if** Rank of block $A \geq T_R$ and $C_A \leq T_C$ **then**
10: **if** Rank of block $B \geq T_R$ and $C_B \leq T_C$ **then**
11: $\{a, ab_1, ..., ab_6, b\} \leftarrow \mathrm{LI}(a, b, 6)$
12: $\{ax_2, abx_{21}, ..., abx_{24}, bx_2\} \leftarrow \mathrm{LI}(ax_2, bx_2, 4)$
13: $\{ax_1, abx_{11}, abx_{12}, bx_1\} \leftarrow \mathrm{LI}(ax_1, bx_1, 2)$
14: **else if** Rank of block $B \geq T_R$ and $C_B > T_C$ **then**
15: $\{a, ab_1, ab_2, ab_3, B_{51}\} \leftarrow \mathrm{LI}(a, B_{51}, 3)$
16: $\{ax_2, abx_{21}, abx_{22}, B_{61}\} \leftarrow \mathrm{LI}(ax_2, B_{61}, 2)$
17: $\{ax_1, abx_{11}, B_{71}\} \leftarrow \mathrm{LI}(ax_1, B_{71}, 1)$
18: **end if**
19: **end if**
20: Run Steps 10-18 to compute the elements with suffix $ac-$
21: Compute the elements of subblocks B, C and D with the logic as in Steps 9-20
22:
23: *Terminal elements in boundary blocks such as block A*
24: **if** Rank of block $A \geq T_R$ and $C_A \leq T_C$ **then**
25: *Non-corner elements* A_{ij} (i=1:4, j=5:8; i=5:8, j=1:4)
26: **for** $i = 1$ to 4 **do**
27: $\{A_{i5}, ..., A_{i8}\} \leftarrow \{a, ab_1, ab_2, ab_3\}$
28: $\{A_{5i}, ..., A_{8i}\} \leftarrow \{a, ac_1, ac_2, ac_3\}$
29: **end for**
30: *Corner elements* A_{ij} (i=1:4, j=1:4)
31: **for** $i = 1$ to 4 **do**
32: $\{A_{11}, A_{22}, ..., A_{55}\} \leftarrow \mathrm{LI}(A_{11}, A_{55}, 3)$
33: Compute the other non-corner elements with the logic as in Steps 15-17
34: **end for**
35: **end if**

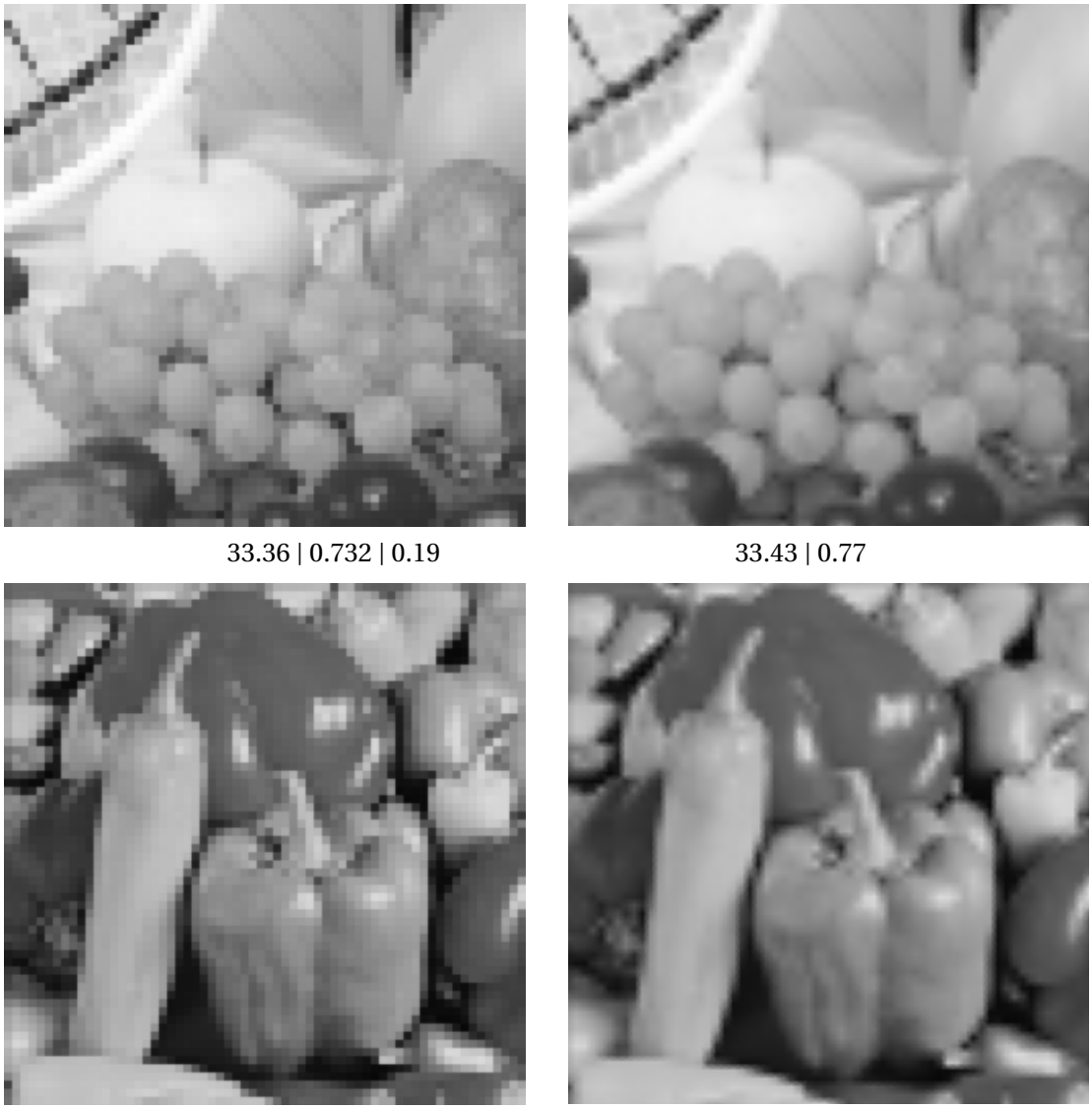


Figure E.13: Some examples of the worst distortion (left) and the improved quality after depixelization (right) for FRUIT (top) and PEPPER (bottom) images. The numbers denote PSNR and MSSIM, respectively.

E.4 Algorithm complexity and FPGA design

We have proposed an FPGA-based platform for the design and implementation of a DMP network node [Wielgosz et al. (2013)] (Figure E.15). It provides a detailed introduction to the platform architecture and the simulation-implementation environment for the design. Our compact implementation on a Xilinx Virtex-6 ML605 board consumes very small amount of the available resources. Moreover, the elementary operations in our implementation take (much) less than 5 μs as desired to meet the low-latency requirement. The AXI bus and the EDK environment are used to implement both the

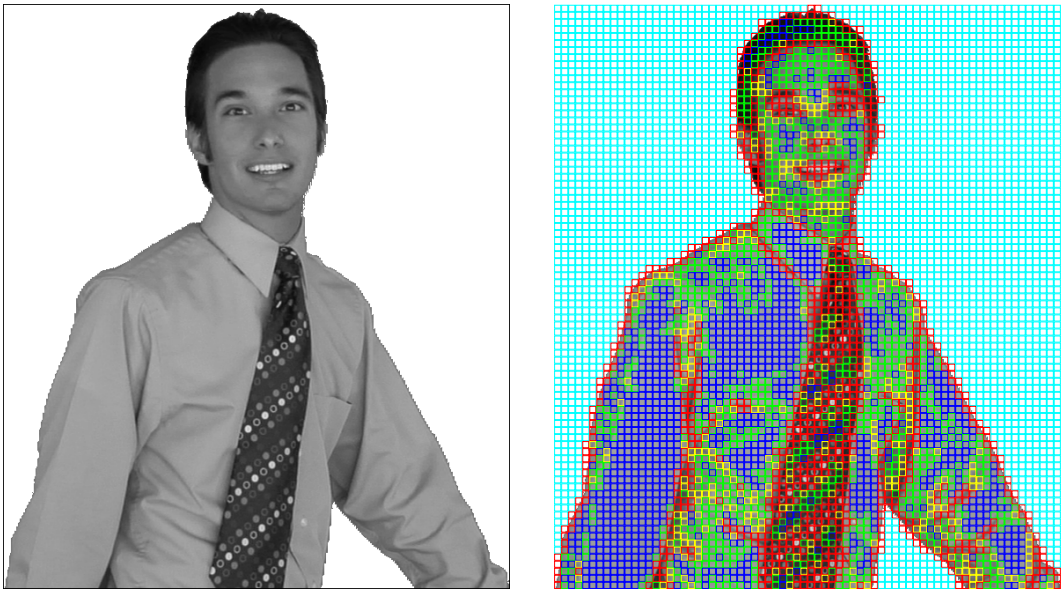


Figure E.14: An image with a segmented object as part of a video frame from a CS's surface (left). The blocks after applying the proposed block ranking algorithm (right). Image border is added for better view by readers.

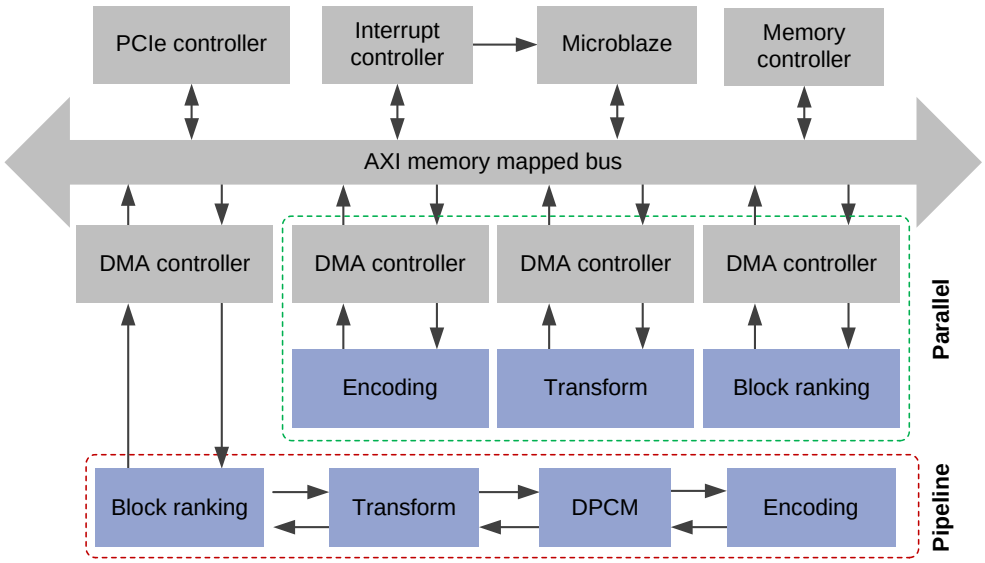


Figure E.15: The FPGA-based architecture of a DMP transmitter with the pipeline and parallel approaches.

transmitter and receiver in DMP. Although the architecture of the access node has different number of compression-scheme components than that of the network node, their core components and adopted processing approach are the same. In addition to controlling the data flow within the FPGA system, Microblaze is also used to establish and maintain the communication with the external DMP servers located on a host (PC machine).

The design's modularity and scalability ease the integration of the external modules into the platform, which can follow parallel, pipeline or hybrid approaches. The first two approaches are depicted in Figure E.15. By assuming equal access in the memory-mapped AXI bus, the parallel approach offers flexibility because it permits software elimination in certain steps of the processing chain if necessary, e.g. the encoding. This is possible because the Microblaze governs all the execution steps of the chain, and they are independently connected to a single AXI bus. On the other hand, the pipeline implementation is more efficient provided that all the modules are used in the processing chain and the pipeline latency is not critical. Adopting both approaches in a hybrid fashion is also an alternative depending on the application.

E.4.1 Calculation of entropy

The complexity for HW design of the major parts of the proposed system is presented as follows. Figure E.16 (a) shows the entropy module, and the dashed line covers the parallel structure.

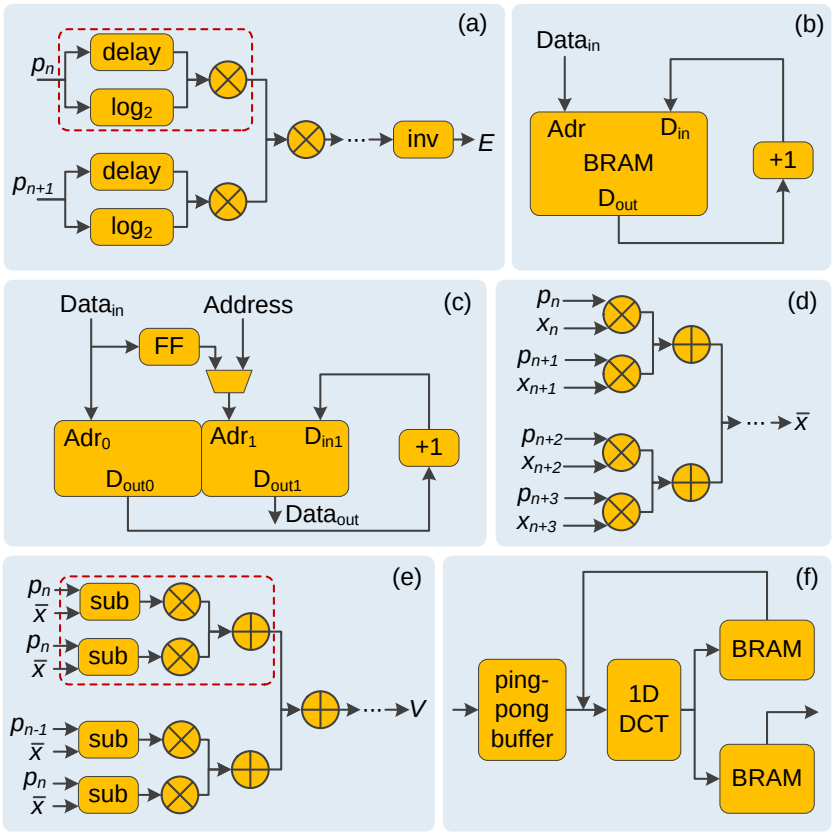


Figure E.16: The modules for calculating entropy (a), histogram (b,c), mean(d), variance (e), and 2D-DCT (f).

The logarithm operation can be implemented as a registered look-up table (LUT) for 8-bit input data at one clock (CLK) [Alachiotis and Stamatakis (2010)]. Thus, the overall latency is 7 CLK, i.e. 3 CLK for each multiplier, and given n parallel structures, it becomes

$1+3\log(n)$ clocks. Consisting of a block RAM (BRAM) memory and an incremental logic, a histogram module with 64 input integer values provides the probability values p_i.

The complexity for HW design of the major parts of the proposed system is presented as follows. Figure E.16 (a) shows the entropy module, and the dashed line covers the parallel structure. The logarithm operation can be implemented as a registered LUT for 8-bit input data at one clock (CLK) [Alachiotis and Stamatakis (2010)]. Thus, the overall latency is 7 CLK, i.e. 3 CLK for each multiplier, and given n parallel structures, it becomes $1+3\log(n)$ clocks. Consisting of a block RAM (BRAM) memory and an incremental logic, a histogram module with 64 input integer values provides the probability values p_i.

Figure E.16 (b) and (c) show the simplified and real diagrams, respectively. The input data for the evaluation of the histogram address the BRAM and the BRAM's D_{out} stores the count of $Data_{\text{in}}$'s prior to the occurrences at the BRAM address bus. The counter is incremented by one and written back to the BRAM at the same address. The BRAM limits the calculation speed as the output data is one CLK delayed with respect to the address bus (a synchronous memory data read); hence, evaluating a single input pixel involves two CLK. Strong parallelization of the computations is possible [Jamro et al. (2007)], and the histogram computation needs 34 LUTs and 23 FFs.

E.4.2 Calculation of mean and variance

Computing mean values of n inputs of $p_i x_i$ (Figure E.16 (d)) takes 3+log(n) CLK, and the variance-calculation module consists of the mean-calculation unit and a set of subtractors, multipliers and adders. They are strongly parallel modules which process the data every clock cycle. The parallelization determines the computation time, and generally it is 4+log(n) CLK plus the latency from the mean-calculation module.

E.4.3 Calculation of 2D-DCT, IDCT and DPCM

By employing a two-pass 1D-DCT transform [Tumeo et al. (2007)], computing a complete 8×8 2D-DCT takes 80 clock cycles and can work at 107 MHz. Adopting a ping-pong fashion, it stores the results of the 1D-DCT by means of an intermediate buffer (Figure E.16 (f)). It is a trade-off between resource consumption and speed which complies well with the idea of an AXI-based Microblaze-controlled architecture. Nevertheless, other implementation approaches for 2D-DCT can be considered, such as replacing the time-consuming multiplications with LUT accesses [Kutka (2002)]. As for the DPCM, its sequential execution flow favors software implementation in Microblaze, and the processing power will not be absorbed because DC coefficients are fewer than AC coefficients.

E.4.4 Encoding and Decoding

Encoding Fibonacci code is simple, but straightforward implementation in iterative procedures needs substantial clock cycles. Therefore, it is better implemented as LUT and executed in one clock, which is feasible because Fibonacci coder for 8-bit numbers consumes merely 8×12 bits, and 3072 bits can fit into a single BRAM memory of 18

Kbits. Thus, it occupies only 2 BRAMs for both encoding and decoding. Moreover, the *bzip2* algorithm can be implemented in software in Microblaze.

E.4.5 Packet Dropping

The dropping module in Figure E.17 is the core component of the QS scheme in a DMP node. The module is integrated to the platform in Figure E.15 also via a direct memory access (DMA) controller. Our strategy is to extensively use AXIS (AXI Streaming) bus which provides system flexibility. All the modules connected to the network node are AXIS-compatible.

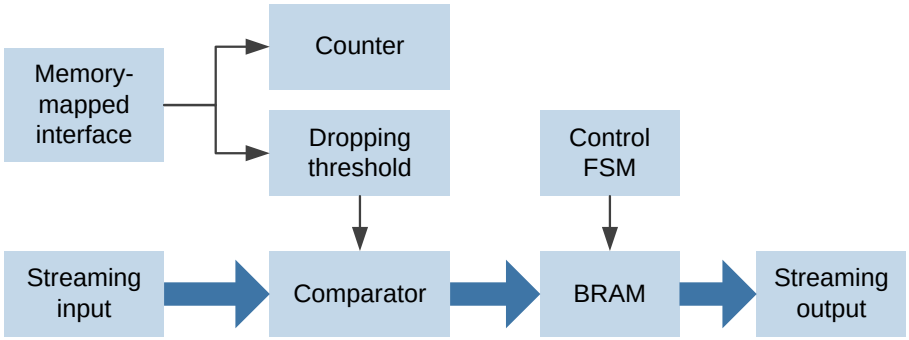


Figure E.17: The proposed structure for DMP dropping module.

The dropping module works as follows. The network packets carrying image data are sent over the PCIe to the external memory (DDR3 in ML605) and stored on a long queue. The Microblaze monitors the status of the queue, programs DMA controller to read the data from the external memory, and writes them to the dropping module. Based on the data received from the other nodes in the network, a current threshold value for dropping is computed and written to the internal register of the dropping module. The data fed into the dropping module from the external memory by the DMA are either dropped or passed through to the internal memory (BRAM) after compared with the threshold. Once the DMA write-operation is finished, the Microblaze is interrupted and informed that the dropping statistics can be read from the internal register of the dropping module. The Microblaze then programs the DMA controller based on the statistics, and the data stored in the internal memory are transferred to the external memory. The dropping process is finished when the data is read from the internal memory of the dropping module and written to the external RAM. All HW modules in both access and network nodes are interconnected with AXIS bus and controlled by the Microblaze.

E.4.6 Depixelization as Post-Processing

Depixelization is essentially a finite-impulse response (FIR) filter operation conducted on block borders. Xilinx delivers a FIR compiler tool which can be used to compile the FIR architectures to generate the depixelization filter [Xilinx (2012)]. It includes

12-bit coefficients and 60-tap input data which can work at 150 MHz and consume 3382 LUT-FF (flip-flop) pairs.

E.4.7 Overall performance

The system performance strictly depends on the chosen architecture (parallel, pipeline or hybrid) and the synchronization between the modules. The following is the gross estimate of the computational time for processing a video frame as a 1920×1080 color image. By assuming 256-bit AXI bus width, 64-pixel block, and 100 MHz FGPA clock cycle as the main constraints, the internal processing speed per single thread becomes 50 Mblocks/s. As the luma and 25% subsampled chroma images equal 48,600 blocks, the essential processing time becomes 1 ms per frame. Since the object-based processing reduces the number of blocks processed per frame, the processing time per frame is less than 1 ms.

The consumed resources are detailed in Table E.2, and the additional pre- and post-processing steps are excluded. LUTs consume the most resources which are roughly 10% of all the XCVLX240T resources, the FPGA used in ML605 [Virtex series (2012)]. It is possible to balance the usage of LUTs with DSP implementation to equalize resource consumption. Consequently, 15 to 20 parallel processing streams can be implemented as in Figure E.15 which reduce the processing time to approximately 50 μs per frame. Moreover, several FPGA boards can be used as a one-stop system [One Stop Systems (2012)] to achieve chip-level parallelism.

Table E.2: Total consumption of resources

Module	#LUT	#FF	#BRAM
Entropy	$m \cdot \log(m) \cdot 115 + n \cdot 19$	$110 + n \cdot 64$	0
Histogram	34	23	1
Variance	$n \cdot 115 + [n + n \cdot \log(n)] \cdot 8$	$n \cdot 110$	0
DCT	123	110	2
Fibonacci	0	0	2

m denotes the number of parallel inputs for entropy module

E.5 Conclusion and future work

We have presented an ultrafast, DCT-based, embedded image-compression scheme which is quality scalable and can process objects with arbitrary shapes. It is designed for network architecture such as DMP that guarantees maximum EED. The encoder mainly consists of block ranking, 2D-DCT and entropy coding. As the quantization as in existing image coding schemes is not present, the main loss of information in this approach is due to the intelligent dropping of data packets by network nodes during transmission to guarantee local delay. Since simplicity and parallelization are favored for minimizing

processing time, block entropy and variance are used for block ranking, which work satisfactorily by yielding four ranks of 8×8-blocks with increasing importance. Universal codes are employed to encode the resulting block ranks and DCT coefficients. The VQ in PSNR and MSSIM of several common test images due to dropping is given against the bitrates, which are also compared to the results from JPEG. JPEG performs better as expected because it can exploit global redundancies in an image and the bitstream, but lacks the scalability. Fundamental differences between the two schemes make such a performance comparison essentially irrelevant. Excessive dropping results in pixelation artifacts that are faithfully contained in the blocks which the receiver can immediately locate from the side information available in the packet headers of the remaining bitstream. A depixelization algorithm, a post-processing step at the receiver, is proposed for the worst distortion. We show how the scheme can be applied to objects of arbitrary non-rectangular areas in images after segmentation. Every video frame, channel, segmented object, and block are processed independently, allowing fully parallel HW implementation. Finally, as indicated by the estimated complexity and resource consumption of the proposed scheme for FPGA implementation, a video frame as a 1920×1080 color image can be processed, encoded and decoded in less than 1ms, sufficient to meet the maximum EED at 11.5ms. Ideas for further performance improvement include incorporating fast intra-prediction and advanced depixelization.

References

Ahmed, N., Natarajan, T., Rao, R., 1974. Discrete cosine transform. *IEEE Transactions on Computers* 23 (1), 90–93.

Alachiotis, N., Stamatakis, A., 2010. Efficient floating-point logarithm unit for FPGAs. In: *Proc. IEEE International Symposium on Parallel & Distributed Processing, Workshops and PhD Forum (IPDPSW)*. pp. 1–8.

Burrows, M., Wheeler, D., 1994. *A block sorting lossless data compression algorithm*. Tech. Rep. 124, Digital Equipment Corporation.

Chafe, C., Gurevich, M., Leslie, G., Tyan, S., 2004. Effect of time delay on ensemble accuracy. In: *Proc. International Symposium on Musical Acoustics*.

Chang, C.-I., Du, Y., Wang, J., Guo, S.-M., Thouin, P., 2006. Survey and comparative analysis of entropy and relative entropy thresholding techniques. *IEE Proceedings - Vision, Image and Signal Processing* 153 (6), 837–850.

DeFanti, T., Acevedo, D., Ainsworth, R., Brown, M., Cutchin, S., Dawe, G., Doerr, K., Johnson, A., Knox, C., Kooima, R., Kuester, F., Leigh, J., Long, L., Otto, P., Petrovic, V., Ponto, K., Prudhomme, A., Rao, R., Renambot, L., Sandin, D., Schulze, J., Smarr, L., Srinivasan, M., Weber, P., Wickham, G., 2011. The future of the CAVE. *Central European Journal of Engineering* 1 (1), 16–37.

Fenwick, P., 2003. *Lossless Compression Handbook*. Academic Press, Ch. *Universal Codes*, pp. 55–78.

Fraenkel, A., Klein, S., 1996. Robust universal complete codes for transmission and compression. *Discrete Applied Mathematics* 64 (1), 31–55.

Golomb, S., 1966. Run-length encodings. *IEEE Transactions on Information Theory* 12 (3), 399–400.

Holub, P., Matela, J., Pulec, M., Srom, M., 2012. Ultragrid: low-latency high-quality video transmissions on commodity hardware. In: *Proc. ACM International Conference on Multimedia*. pp. 1457–1460.

ITU-T, May 2003. Advanced Video Coding for Generic Audio-Visual Services. ITU-T Rec. H.264 and ISO/IEC 14496-10 (AVC), ITU-T and ISO/IEC JTC 1.

Jamro, E., Wielgosz, M., Wiatr, K., 2007. FPGA implementaton of strongly parallel histogram equalization. In: *Proc. IEEE Design and Diagnostics of Electronic Circuits and Systems (DDECS)*. pp. 1–6.

Kopf, J., Lischinski, D., 2011. Depixelizing pixel art. *ACM Transactions on Graphics* 30 (4), 99:1–99:8.

Kutka, R., 2002. Fast computation of DCT by statistic adapted look-up tables. In: *Proc. IEEE International Conference on Multimedia and Expo (ICME)*. pp. 781–784.

Ohm, J.-R., 2005. Advances in scalable video coding. *Proceedings of the IEEE* 93 (1), 42–56.

One Stop Systems, 2012. http://www.onestopsystems.com/.

Rønningen, L. A., 2011. *The Distributed Multimedia Plays Architecture (version 3.20)*. Tech. rep., ITEM, NTNU.

Rønningen, L. A., Wittner, O., 2011. *Experiments on remote conducting between Trondheim and Lisbon*, ITEM, NTNU.

Schelkens, P., Skodras, A., Ebrahimi, T., 2009. *The JPEG 2000 Suite*. Wiley, Series: Wiley-IS&T Series in Imaging Science and Technology.

Schwarz, H., Marpe, D., Wiegand, T., 2007. Overview of the scalable video coding extension of the H.264/AVC standard. *IEEE Transactions on Circuits and Systems on Video Technology* 17 (9), 1103–1120.

Seward, J., 2012. bzip2 codec. http://www.bzip.org/.

Shannon, C., 1948. A mathematical theory of communication. *Bell System Technical Journal* 27, 379–423.

Shapiro, J., 1993. Embedded image coding using zerotrees of wavelet coefficients. *IEEE Transactions on Signal Processing* 41 (12), 3445–3462.

Solomon, D., Motta, G., 2010. *Handbook of Data Compression*. Springer.

Sorwar, G., Abraham, A., Dooley, L., 2001. Texture classification based on DCT and soft computing. In: *Proc. IEEE International Conference on Fuzzy Systems*. pp. 545–548.

Sun, H., Vetro, A., Xin, J., 2007. An overview of scalable video streaming. *Wireless Communications and Mobile Computing* 7 (2), 159–172.

Taubman, D., 2000. High performance scalable image compression with EBCOT. *IEEE Transactions on Image Processing* 9 (7), 1158–1170.

Taubman, D. S., Marcellin, M. W., 2001. *JPEG 2000: Image Compression Fundamentals, Standards and Practice*. Kluwer Academic Publishers.

TGFX, 2012. http://www.timelinegfx.com/.

Tucker, R., 2006. The role of optics and electronics in high-capacity routers. *Journal of Lightwave Technology* 24 (12), 4655–4673.

Tumeo, A., Monchiero, M., Palermo, G., Ferrandi, F., Sciuto, D., 2007. A pipelined fast 2D-DCT accelerator for FPGA-based SoCs. In: *Proc. IEEE Computer Society Annual Symposium on VLSI*. pp. 331–336.

Vajda, S., 1989. *Fibonacci and Lucas Numbers, and the Golden Section Theory and Applications*. Ellis Horwood.

Van der Vleuten, R., Kleihorst, R., Hentschel, C., 2000. Low-complexity scalable DCT image compression. In: *Proc. IEEE International Conference on Image Processing (Volume 3)*. pp. 837–840.

Virtex series, 2012. http://www.xilinx.com/publications/matrix/Virtex_Series.pdf.

Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E., 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* 13 (4), 600–612.

Wielgosz, M., Panggabean, M., Wang, J., Rønningen, L. A., 2013. An FPGA-based platform for a network architecture with delay guarantee. *Journal of Circuits, Systems and Computers* 22 (6).

Wu, D., Hou, Y., Zhang, Y.-Q., 2000. Transporting real-time video over the internet: challenges and approaches. *Proceedings of the IEEE* 88 (12), 1855–1875.

x264, 2013. http://www.videolan.org/developers/x264.html.

Xilinx, 2012. FIR compiler. http://www.xilinx.com/support/documentation/ip_documentation/fir_compiler_ds534.pdf.

Yang, Z., Yu, B., Wu, W., Nahrstedt, K., Diankov, R., Bajscy, R., 2006. A study of collaborative dancing in tele-immersive environments. In: *Proc. IEEE International Symposium on Multimedia*. pp. 177–184.

Zhang, J., Tan, T., 2002. Brief review of invariant texture analysis methods. *Pattern Recognition* 35 (3), 735–747.

PAPER **F**

Resampling HD images with the effects of blur and edges for future musical collaboration

Mauritz Panggabean and Leif Arne Rønningen

This paper, in the original version, has been published in the Proceedings of 23rd *Norsk informatikkonferanse* (NIK) 2010, organized by *NIK-stiftelsen* in Gjøvik, Norway on November 22-24, 2010.

Abstract

Image down/upsampling can give significant bitrate reduction without noticeable quality reduction. This is attractive to our vision of real-time multiparty collaboration from distributed places with video data at HD resolution that must guarantee maximum EED of 11.5ms to enable musical synchronization. Based on the DMP architecture, this paper compares the performances of bicubic and Lanczos techniques as well as one recently proposed for image upsampling in terms of computing time and objective image quality in PSNR and SSIM. The effects of image blur and edges to resampling are also examined and discussed. The results show that the classic Lanczos-2 and bicubic techniques perform the best and thus are suitable for the vision due to their potential for efficient parallel processing in HW. We also show that composite images with different downsampling factors can achieve 4dB increase in PSNR.

F.1 Introduction

The fields of electronics and communication technology have been growing rapidly and opening new and more creative ways of real-time multiparty collaborations limited only by time and space. Our vision is to realize such collaborations in the future with near-natural quality of experience through networked CSs that enable seamless integration of virtual (taped) and live scenes from distributed sites on the continents despite different technical specifications. Figure F.1 depicts a simple exemplary scenario where an audience in Oslo (A) are enjoying a concert featuring two opera singers in a specially designed room, namely a collaboration space. The quality of experience is expected to be *near-natural* which means that they hardly realize that two singers are performing live from two different cities, say Trondheim (B) and Tromsø (C), each in their own CS. Both simultaneous performances are so harmonious with life-like multimedia quality that the audience feel they are enjoying a live opera concert and they are in the very same room with the two singers. Furthermore each opera singer singing live also experiences performing together with the other two displayed in his or her own CS, as if they are on the same stage.

Table B.1 lists the main technical requirements on important aspects for the envisioned collaborations. Guaranteeing the maximum EED to be ≤ 10-20ms to enable good synchronization in musical collaboration is the most challenging. Chafe et al. (2004) reported experimental results on the effect of time delay on ensemble accuracy by placing pairs of musicians apart in isolated rooms and asking them to clap a rhythm together. They reported that longer delays produce increasingly severe tempo deceleration while shorter ones yield a modest yet surprising acceleration. The study found that the observed optimal delay for synchronization is 11.5ms that equates with a physical radius of 2,400 km (assuming signals traveling at approximately 70% the speed of light and no routing delays). Realizing such collaborations with very high quality and complexity is possible only if all requirements can be fulfilled within the maximum EED. We observe that current standards are still unable to realize this vision and it leads to our proposal of the three-layer DMP architecture [Rønningen (2011)], as illustrated in Figure F.1. Along

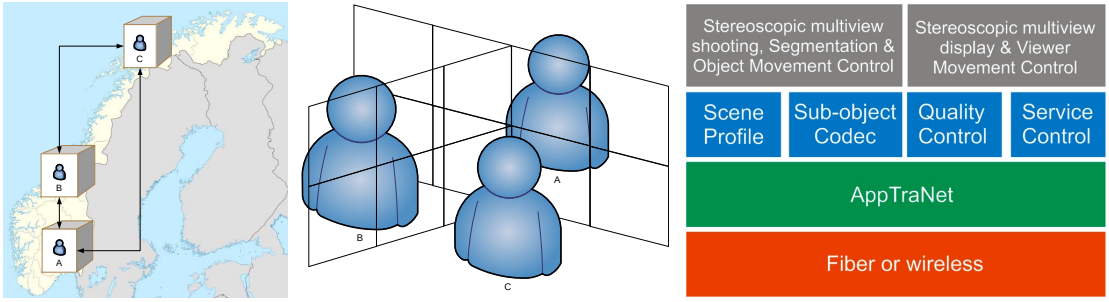


Figure F.1: A simple example of the envisioned collaboration (left) and the corresponding combined CS (middle). All surfaces of the CS consist of arrays of multiview 3D display, dynamic cameras, speakers and microphones. The resulting multimedia data is handled by the proposed three-layer DMP architecture shown from a user's perspective (right).

with the proposed architecture, we also introduce the concept of Quality Shaping built upon the concept of Traffic Shaping [Rønningen (1982)] that degrades the video quality in graceful manner when traffic overloads or system malfunctions occur.

The traffic generated from near-natural scenes at high-definition (HD) resolution from arrays of cameras is estimated to be extremely high which may be up to $10^3 - 10^4$ higher than that of today's videoconferencing systems. Near-natural quality requires an extreme resolution with data rates in Gbps, even to represent a human face alone. Applying block-based video coding to the video data will incur more processing delays particularly for encoding that utilizes interframe dependencies. Therefore DMP system favors full independence between frames and between objects withing a frame in the video data to enable fully parallel video processing in HW. Novel techniques for that purpose are under current investigation in our research. However, as additional phase prior to that, it is attractive to downsample HD images at the transmitter and upsample them at the receiver to considerable reduce the bitrate without losing much quality. The resampling techniques must show promising potential for parallel HW implementation to achieve very fast resampling of HD images.

This paper presents our study of the latter resampling idea based on the two objectives as follows. First, we compare four resampling techniques considered promising from literature in four comparison criteria: computational time, objective image quality assessment using PSNR and SSIM index [Wang et al. (2004)], and the quantified resulting blur. SSIM is used due to its close approximation to subjective image quality assessment and its widespread use. The four techniques are Lanczos technique with two and three lobes, bicubic technique [Keys (1981)] and that proposed by Shan et al. (2008). In the rest of this paper, the first three and the latter are called the classic techniques and the new technique, respectively. Second, as blur and edges are often present in major regions in natural images, we examine their effects to the best performing techniques from the experiments on the first objective. Our goal is to utilize them in applying the resampling techniques to attain further bit saving for transmission without noticeable video-quality reduction.

The organization of the paper is as follows. Section F.2 gives a brief presentation on the examined resampling techniques and elaborates the setup of the experiments. Sec-

tion F.3 presents the experimental results with evaluations that lead to our conclusions in Section F.4 as our main contributions.

F.2 Image resampling techniques and experimental setup

An in-depth formal explanation on image interpolation and resampling can be found for example in [Thevenaz et al. (2000)] where they defined interpolation as a model based recovery of continuous data from discrete data within a known range of abscissa. Among many interpolation and resampling techniques for images, the Lanczos-windowed sinc functions offer the best compromise in terms of reduction of aliasing, sharpness, and minimal ringing [Turkowski and Gabriel (1990)]. Lanczos kernels with two and three lobes are widely used, as shown in Figure F.2. They are called Lanczos-2 and Lanczos-3 afterward in the text, respectively.

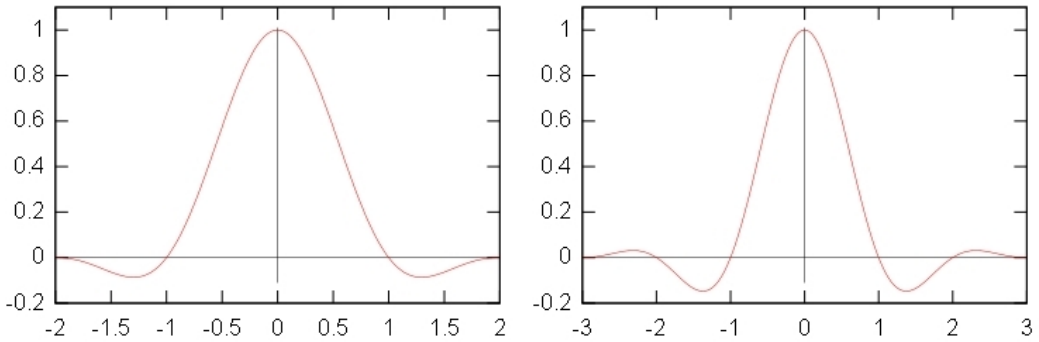


Figure F.2: Kernels of Lanczos-2 (left) and Lanczos-3 (right) techniques.

It is also a common fact that bicubic resampling is widely used for most images as it represents a good trade-off between accuracy and speed. These justify our selection of these classic techniques in our study. In addition to these, most recent years have seen proposals of novel techniques for upsampling natural images, for example in [Shan et al. (2008); Fattal (2007)]. Although the resulting VQ in the latter is promising, but it is unacceptably slow for HD images. The first claims better VQ than that of the latter and also presents possibilities for parallel HW implementation, yet without clear indication of the required computing time. Thus it is included in our study only for image upsampling.

To achieve the two objectives aforementioned, two experiments namely Experiment A and B are conducted in Matlab on a PC with 3GHz processor and 3.46GB RAM, respectively. As one exception, the image upsampling by the new technique employs the application provided online by Shan et al. (2008). The details of the experiments are as follows. Experiment A compares the four resampling techniques in terms of computing time and objective image quality assessment using PSNR and SSIM. We assume that the test images are captured by a digital camera at 1920×1080 resolution and then they are downsampled with a number of downsampling factors (DFs). Afterward the images resulting from the downsampling are upsampled with the same DF by using the four

techniques. The same classic technique is used for both downsampling and upsampling. For simplicity, in the rest of the text, the process of downsampling continued with upsampling the result with a *DF* is called down/upsampling. As the resolution of the downsampled images must be integers, the selected *DF*s in Experiment A are 1.2, 1.5, 2.0, 2.5, 3.0, and 4.0, as shown in Table F.1 with the resulting downsampled resolutions. *DF*s can denote approximate bit savings from the original test images. Figure F.3 illustrates the comparison of the downsampled resolutions of images for transmission to show clearly the magnitude of the possible data reduction. The blur caused by resampling with increasing *DF* will also be quantified by means of an objective blur metric [Crete et al. (2007); Do (2009)].

Table F.1: *DF*s and the resulting resolutions relative to 1920×1080.

DF	Resulting resolutions	*DF*	Resulting resolutions
1.2	1600×900	4.0	480×270
1.5	1280×720	5.0	384×216
2.0	960×540	6.0	320×180
2.5	768×432	7.5	256×144
3.0	640×360	8.0	240×135

Figure F.3: Original resolution (left) and, next to the right, those downsampled with *DF* equals 2.0, 4.0 and 8.0, respectively, to graphically illustrate the magnitude of the data reduction achieved by resampling.

Experiment B examines the effects of image blur and edges to resampling. The level of blur is quantified as in Experiment A. The objective image quality will be presented as a function of *DF*s and the blur metric. The used *DF*s include those in Experiment A with extension to 4.0, 5.0, 6.0, 7.5 and 8.0, as also listed in Table F.1 with the corresponding downsampled resolutions. Different *DF*s are also applied in down/upsampling an image with clear (foreground) and blurred (background) regions. The quality of the resulting images will be quantified as a function of increasing *DF*s. Exemplary resulting images from both experiments will be provided for subjective assessment by the readers.

F.3 Experimental results and evaluations

This section presents more details on the two experiments with the results and evaluations as the main contributions from this work.

F.3.1 Experiment A: comparison of resampling techniques

We used HD images of human faces from [Center for Biological and Computational Learning, MIT (2005)] in this experiment since CSs support live collaboration between humans and human faces are the most important object to process. All test images show human faces either from frontal or non-frontal sides, as shown in Figure F.4. This suits the design of the CS in which the arrays of cameras can capture the face of the persons within from both sides.

Figure F.4: Typical test images with frontal (left) and non-frontal (middle and right) sides.

Our experimental results indicate that each resampling technique show relatively constant performance in all comparison criteria not only in all test images despite different positions of the faces but also in natural images as depicted in Figure F.3. Therefore is it sufficient to present results from a test image as a general representation to compare the performances of the four resampling techniques. Figure F.5 presents the experimental results for the test image on the left in Figure F.4. Sample images of the area around the right eye of the man in the test image are shown in Figure F.6 to illustrate the effects of resampling for subjective assessment by readers. This area is selected because it contains regions with and without many edges and high frequency contents. All images for assessment are to be seen on screen for best viewing quality. All the test images have plain background that fits our assumption that human faces and other important objects in a CS can be efficiently segmented prior to necessary downsampling.

The quality of the images denoted by PSNR and SSIM in Figure F.5 confirm that Lanczos-2 and bicubic techniques deliver very similar performances in all comparison criteria. Although Lanczos-3 technique yields better image quality and causes less blur, they come at the expense of more processing time, particularly for upsampling process. Surprisingly, the worst performance in image quality and computing time goes to the new technique. Despite the high-end computing power for this experiment, the technique fails to work for $DF < 2.0$. For $DF \geq 2.0$, the technique operates within several minutes and this fact justifies the absence of the technique in the diagram on computing time. When $DF = 3.0$ the resulting image quality of this technique behaves very strangely for which good explanation is still required. Nevertheless it is clear from

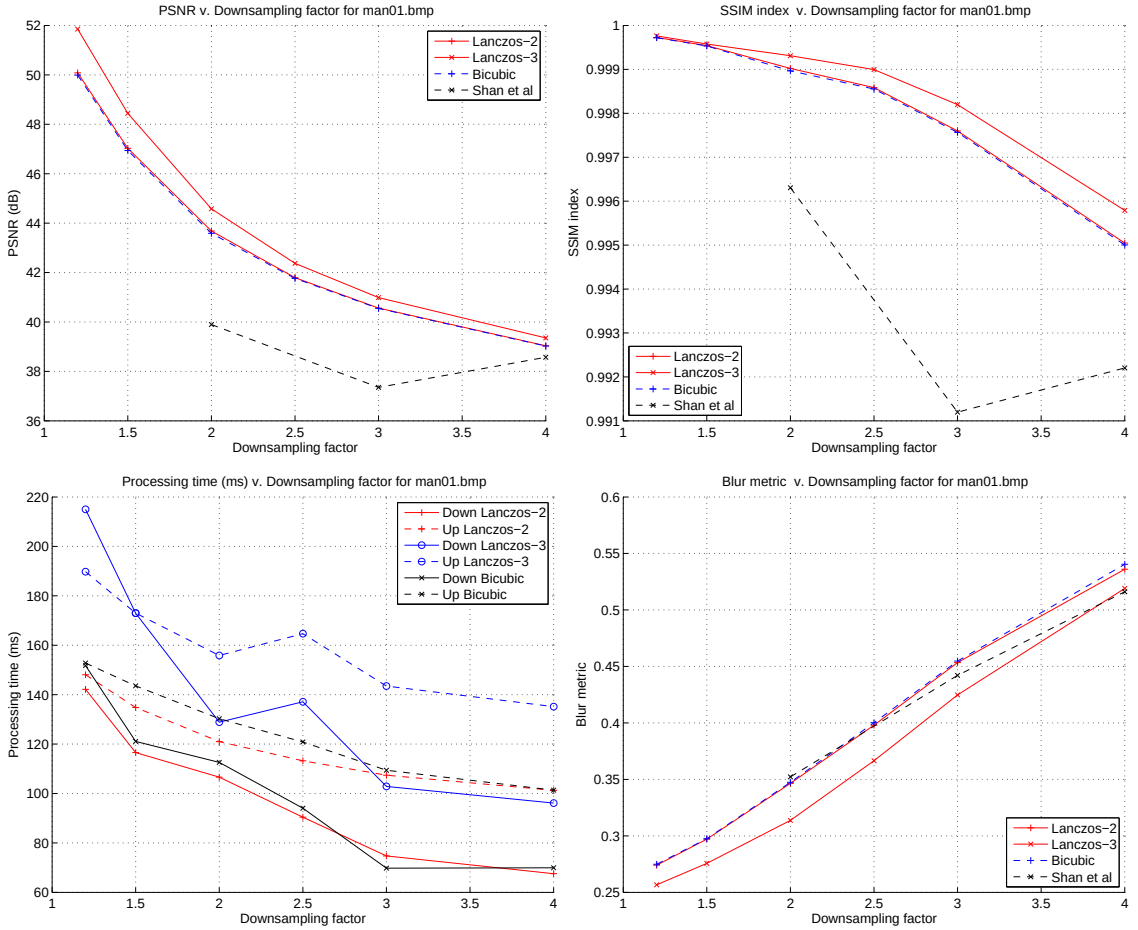


Figure F.5: Image quality in PSNR, SSIM, processing times and blur metrics for the test image on the left in Figure F.4.

the results that Lanczos-2 and bicubic techniques perform the best and are worth exploring further in our research.

There might be a question: what if the image is already captured by the camera at the downsampled resolution? If the image is then upsampled by the resampling techniques, what is the effect to the four comparison criteria? These are interesting questions because if the camera can do that, no more downsampling is necessary after image acquisition which thus saves more time. We attempted to examine this by comparing the images sequentially captured by a high-end camera with a remote control in different resolutions with the same aspect ratio. Despite the use of remote control while capturing the images, there are always minor shifts of pixels in those images which cause considerable reduction in image quality when comparing an original image at a low resolution and that downsampled to that resolution from a higher one. However those images look almost the same from subjective point of view. This study is saved for further work as it requires more advanced device that can capture a scene and save it into images with different resolution with the same aspect ratio in one go. It is aligned to our research since *dynamic* cameras in the CS imply that their parameters, including image resolution and aspect ratio, can be fully controlled on the fly by the DMP system.

Figure F.6: Sample images from the test image on the left in Figure F.4. The top, middle and bottom row refers to DF equals 2.0, 3.0, and 4.0, respectively. The left to right columns refer to bicubic, Lanczos-2, Lanczos-3 and the new techniques, respectively. Images are to be seen on screen for best quality.

F.3.2 Experiment B: the effects of blur and edges to resampling

Artifacts caused by image upsampling have been categorized as ringing, aliasing, blocking, and blurring [Thevenaz et al. (2000)]. The latter is more apparent in sample images shown in Figure F.6 as the DFs are higher. However, if we take the reverse direction, an interesting question raises: what will be the effect to the result after down/upsampling if the original image is more blurred? Experiment B attempts to provide a quantitative answer to the question.

We induce more blur into the test images by applying Gaussian low-pass filter with standard deviation 10 and increasing size. Then we down/upsample the blurred test images using Lanczos-2 technique with all ten DFs in Table F.1. As in Experiment A, the resulting image quality in PSNR and SSIM from the test images are typically similar, as shown in Figure F.7 for the test image on the left in Figure F.4. The legend in Figure F.7 denotes the level of blur objectively quantified in the blurred test images where higher number indicates more blur. Figure F.8 depicts sample images of the right eye area of the man in the test image as in Figure F.6 cropped from the blurred test images.

Comparing the curves in Figure F.7 and sample images in Figure F.8 show some interesting findings. First, the more blurry the original image, the higher the DF for down/upsampling the image with unobjectionable quality of the resulting image. This is clear as images with more blur have lighter tails in SSIM than those of clearer images. The initial values of PSNR and SSIM for $DF = 1.2$ are also higher for test images with more blur. Second, ringing artifact will be induced when the DF is already too high and this occurs more likely to clearer images. This artifact is mainly caused by overshoot and

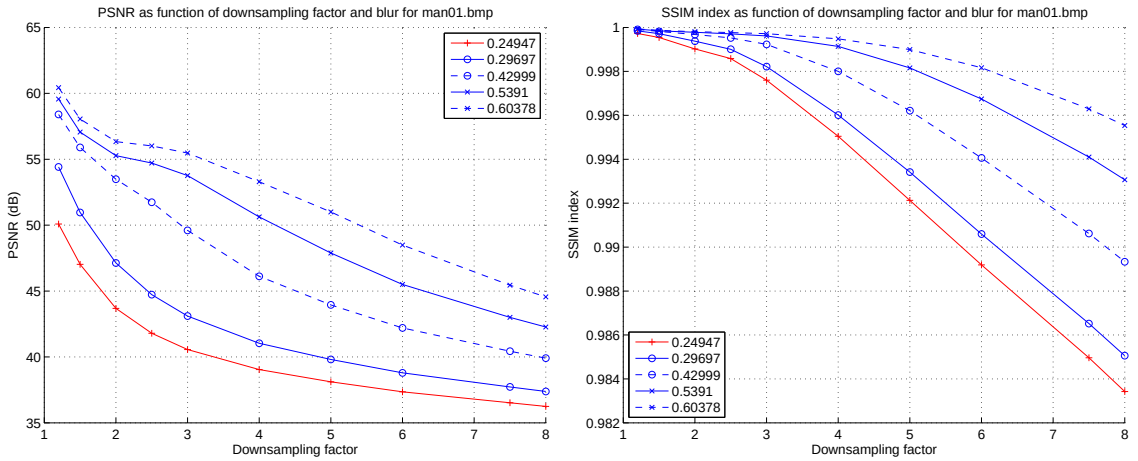


Figure F.7: Image quality in PSNR (left) and SSIM (right) for the test image on the left in Figure F.4 using Lanczos-2 technique.

Figure F.8: Sample images from the test image on the left in Figure F.4. The top, middle and bottom row refers to blur index 0.24947, 0.42999, and 0.60378, respectively. The left column presents original test images with initial blur, while the rest columns to the right are the results with DF equals 2.0, 4.0, and 8.0, respectively. Images are to be seen on screen for best quality.

oscillations by the lobes of the Lanczos-2 filter. It is likely that the presence of ringing artifact can be exploited to indicate the limit of acceptable DFs. Third, the PSNR values for two most blurred images at $DF > 2.0$ are very interesting for further analysis.

It is common that an image has regions of different levels of blur, for example the background is much more blurry than the crisp and clear main object in the foreground. The above results lead us to another interesting question: how will the result look like when such different regions in an image are down/upsampled with different DFs as a function of their level of blur? We attempt to answer it by applying the idea to natural

HD images of selected scenes captured from [Ritchie (2009)].

Assume a test image with a region of interest (ROI) defined in a bounding box. First, the test image and the ROI are down/upsampled with DFs DF_I and DF_{ROI}, respectively, where $DF_{ROI} \leq DF_I$. The image resulting from down/upsampling the test image with DF_I is denoted as *overall* image while the *ROI* image refers to the resulting ROI which quality is compared with the luminance values of the ROI in the test image. Finally, the resulting ROI is patched to the same location in the *overall* image and the eventual resulting image is called a *composite* image.

Figure F.9 shows an original HD image and three resulting images with three configurations of DFs using Lanczos-2 technique as detailed in the figure caption. In the original test image, the background is very blurry without strong edges and the face in the foreground is very clear and rich in details. The image qualities of different regions in PSNR and SSIM from Figure F.9 are presented in Table F.2. Another test image which background region is also blurry but has many strong edges is examined with the same configurations and the results are shown in Figure F.10 and Table F.3.

(a) **(b)**

(c) **(d)**

Figure F.9: Original image (a), overall image with DF = 4.0 (b), composite image which ROI and overall images are down/upsampled with DF equals 2.0 and 4.0, respectively (c), and that with DF equals 2.0 and 8.0, respectively (d). The ROI is 26% of the image. Images are to be seen on screen for best quality.

Images (c) in Tables F.2 and F.3 clearly have the best quality relative to the others of the same scene. The improvement in the quality of the composite images with respect to that of the corresponding overall images is contributed by that of the ROI. Considerable 2-4db and 1.5-2dB increase in PSNR can be achieved by composite images (c) and (d) in Figures F.9 and F.10, respectively. Although composite image (d) in Figure F.9 and

Table F.2: Image qualities of sample images in Figure F.9 (PSNR in dB).

	ROI			Overall			Composite	
Image	DF_{ROI}	PSNR	SSIM	DF_I	PSNR	SSIM	PSNR	SSIM
(a)	4	41.37	0.975	4	45.39	0.998	45.35	0.998
(b)	2	44.80	0.996	4	45.39	0.998	47.27	0.999
(c)	2	44.80	0.996	8	41.63	0.991	45.56	0.998

(a) **(b)**

(c) **(d)**

Figure F.10: Original image (a), overall image with $DF = 4.0$ (b), composite image which ROI and overall images are down/upsampled with DF equals 2.0 and 4.0, respectively (c), and that with DF equals 2.0 and 8.0, respectively (d). The ROI is 26% of the image. Images are to be seen on screen for best quality.

Table F.3: Image quality of sample images in Figure F.10 (PSNR in dB).

	ROI			Overall			Composite	
Image	DF_{ROI}	PSNR	SSIM	DF_I	PSNR	SSIM	PSNR	SSIM
(a)	4	37.35	0.955	4	40.06	0.996	40.04	0.996
(b)	2	41.90	0.994	4	40.06	0.996	41.68	0.998
(c)	2	41.90	0.994	8	35.67	0.973	37.47	0.986

Table F.2 has acceptable quality shown by very high PSNR and SSIM values, the quality of composite image (d) in Figure F.9 and Table F.2 is the worst for that scene. This shows the effect of edges in an image to how far it can be down/upsampled.

Figure F.11: Images extracted from Figure F.9: original (left), $DF = 4.0$ with blur (middle), and $DF = 8.0$ with ringing artifact and more blur (right).

As obvious in composite image (d) in Figure F.9, unacceptable DF causes not only more blurring artifact but also ringing artifact, particularly in the areas where the edges are strong, as depicted in Figure F.11 and also by the images in the fourth column of Figure F.8. Nevertheless, the quality of composite image (c) in Figure F.9 is still acceptable, as shown by the values of PSNR and SSIM. These results lead to a safe recommendation that for an image with clear region as ROI and blurry region, the first region can be down/upsampled with $DF = 2.0$ and the latter with $DF = 4.0$, although strong edges are present in the blurry region. The level of blur must certainly be quantified beforehand. Composite images are expected to perform better in quality and bitrate than overall images. If the density of the edges in the blurry region can be quantified such as in [Phung and Bouzerdoum (2007)], more bits can be saved by applying $DF > 4$ to the blurry area. Better metrics to quantify edge density is open for future work.

F.4 Conclusion

We have presented our examination on four resampling techniques for down/ upsampling HD images. Our experimental results show that the performances of each technique are relatively constant not only in the tested images showing human faces but also in other natural images. Lanczos-2 and bicubic techniques comparatively perform the best in terms of computing time, objective image quality and the level of introduced blur. Computing time is very important in our comparison since this study is driven by our vision of future musical collaboration with maximum EED of 11.5ms to guarantee good synchronization between collaborating musicians. Lanczos-2 and bicubic techniques are based on convolution that is promising for parallel implementation in HW, for example using systolic approach in FPGA [Kung (1982); Nuno-Maganda and Arias-Estrada (2005)], making them good candidates for next pursuit in our research.

The results from our second experiment reveals that blur and edges are important factors when down/upsampling HD images. Higher DFs can be applied to an image or a region in an image that is more blurry than others with unobjectionable quality of the resulting image. Ringing artifact in the result can be used to indicate that the applied DF is already too high. If an image has major clear and blurry regions, usually

as foreground and background, respectively, the latter can be down/upsampled with higher DF than that applied in down/upsampling the first. Combining the results from the two down/upsampling processes yields a composite image that we can expect to give a better image quality with less bitrate than that from applying a DF to down/upsample the entire image. Our experimental results reveal that the improvement of image quality in PSNR can reach up to 4dB in composite images. This increase is attributed to the clear region down/upsampled with higher DF.

As edges limit DFs in down/upsampling due to unwanted ringing artifact, our safe recommendation is to down/upsample the clear and blurry regions with $DF = 2.0$ and $DF = 4.0$, respectively, assuming strong edges present in the blurry region. This should be preceded by applying available metrics to quantify the level of blur and the density of the edges in the blurry region. If the region is very blurry with few weak edges detected, it can be down/upsampled with higher DFs, even up to 8.0 without making the resulting quality reduction very apparent. However if the down/upsampling is applied to the whole HD image without ROI, using $DF = 2.0$ is a safe recommendation.

References

Center for Biological and Computational Learning, MIT, 2005. Face recognition database. http://cbcl.mit.edu/software-datasets/heisele/facerecognition-database.html.

Chafe, C., Gurevich, M., Leslie, G., Tyan, S., 2004. Effect of time delay on ensemble accuracy. In: *Proc. International Symposium on Musical Acoustics*.

Crete, F., Dolmiere, T., Ladret, P., Nicolas, M., 2007. The blur effect: perception and estimation with a new no-reference perceptual blur metric. In: *Proc. SPIE Human Vision and Electronic Imaging XII*. p. 64920I.

Do, Q., 2009. Matlab implementation of image blur metric. http://www.mathworks.com/matlabcentral/fileexchange/24676-image-blur-metric.

Fattal, R., 2007. Image upsampling via imposed edges statistics. *ACM Transactions on Graphics* 26 (3), No. 95.

Keys, R., 1981. Cubic convolution interpolation for digital image processing. *IEEE Transactions on Signal Processing, Acoustics, Speech, and Signal Processing* 29 (6), 1153–1160.

Kung, H., 1982. Why systolic architectures. *Computer Magazine* 15 (1), 37–45.

Nuno-Maganda, M., Arias-Estrada, M., 2005. Real-time FPGA-based architecture for bicubic interpolation: an application for digital image scaling. In: *Proc. International Conference on Reconfigurable Computing and FPGAs*.

Phung, S., Bouzerdoum, A., 2007. Detecting people in images: An edge density approach. In: *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*. pp. I–1229–I–1232.

Ritchie, G., 2009. Sherlock Holmes. Trailer in 1080p resolution. http://www.digital-digest.com/movies/Sherlock_Holmes_1080p_Theatrical_Trailer.html.

Rønningen, L. A., 1982. Input traffic shaping. In: *Proc. International Teletraffic Congress*.

Rønningen, L. A., 2011. *The Distributed Multimedia Plays Architecture (version 3.20)*. Tech. rep., ITEM, NTNU.

Shan, Q., Li, Z., Jia, J., Tang, C.-K., 2008. Fast image/video upsampling. *ACM Transactions on Graphics* 27 (5), 153:1–153:7.

Thevenaz, P., Blu, T., Unser, M., 2000. *Handbook of Medical Imaging*. Academic Press, Ch. *Image interpolation and resampling*, pp. 393–420.

Turkowski, K., Gabriel, S., 1990. *Graphics Gems I*. Academic Press, Ch. *Filters for common resampling tasks*, pp. 147–165.

Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E., 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* 13 (4), 600–612.