Low-Complexity Scalable and Multiview Video Coding

—

Laagcomplexe schaalbare en meervoudig-perspectieve videocompressie

—

Sebastiaan Van Leuven

Promotor: prof. dr. ir. R. Van de Walle, dr. ir. J. De Cock

Proefschrift ingediend tot het behalen van de graad van
Doctor in de Ingenieurswetenschappen: Computerwetenschappen

Vakgroep Elektronica en Informatiesystemen
Voorzitter: prof. dr. ir. J. Van Campenhout
Faculteit Ingenieurswetenschappen en Architectuur
Academiejaar 2012-2013

Laagcomplexe schaalbare en meervoudig-perspectieve videocompressie

Low-Complexity Scalable and Multiview Video Coding

Sebastiaan Van Leuven

Promotoren: prof. dr. ir. R. Van de Walle, dr. ir. J. De Cock
Proefschrift ingediend tot het behalen van de graad van
Doctor in de Ingenieurswetenschappen: Computerwetenschappen

Vakgroep Elektronica en Informatiesystemen
Voorzitter: prof. dr. ir. J. Van Campenhout
Faculteit Ingenieurswetenschappen en Architectuur
Academiejaar 2012 - 2013

ISBN 978-90-8578-612-2
NUR 965
Wettelijk depot: D/2013/10.500/45

Dankwoord

"You know, it's funny what's happening to us.
Our lives have become digital, our friends now virtual,
and everything you could ever want to know is just a click away.
Experiencing the world through endless second hand information isn't enough.
If we want authenticity, we have to initiate it."

–Travis Rice, The Art of Flight (2011)

Het werk dat hier voor u ligt, is niet het werk van één persoon. Vele mensen hebben bijgedragen tot wat het uiteindelijk geworden is. Graag had ik de mensen bedankt die mij op één of andere manier geholpen hebben om dit te verwezenlijken. Vooreerst had ik graag mijn promotor prof. Rik Van de Walle bedankt omdat hij mij de kans heeft gegeven dit doctoraat aan te vangen, omdat hij mij de vrijdheid heeft gegeven verschillende pistes te bewandelen en mij hiervoor te voorzien in voldoende middelen. Daarnaast had ik ook graag mijn co-promotor dr. Jan De Cock bedankt om mij gedurende deze periode bij te staan met raad en daad. Maar ook om mij tijdig te wijzen op mogelijke hindernissen en vooral om de talloze schrijffouten uit mijn papers te halen, en niet in het minst om dit werk verscheidene malen te fine tunen. Zijn hand in dit werk is niet te onderschatten.

Ik had ook graag het Agentschap voor Innovatie door Wetenschap en Technologie (IWT) bedankt voor het ter beschiking stellen van een specialisatiebeurs, die mij in staat stelde om dit werk zonder financiële kopzorgen te kunnen uitvoeren.

I also would like to mention the members of the jury, whom provided me with valuable feedback, which improved the quality of this work: prof. Pedro Cuenca, prof. Patrick De Baets, dr. Jan De Lameilleure, prof. Peter Lambert, prof. Aleksandra Pizurica, and prof. Peter Schelkens.

Daarnaast had ik ook nog graag mijn collega's en voormalig collega's bedankt voor de samenwerking, de mogelijkheden die gecreëerd zijn en de talloze discussies die ervoor gezorgd hebben dat er steeds nieuwe inzichten kwamen. Hierbij had ik ook graag Prof. Monteanu van de Vrije Universiteit Brussel bedankt voor de samenwerking die we de afgelopen jaren gekend hebben.

One might think that the essence of this book lies in the groundbreaking research that has been performed. To me, the essence of this book is not the research or the results, it's the experience that I have gone through. For me, the journey has been the reward.

During this journey, I had the opportunity to do research in universities abroad. Of course these adventures would not be possible without the aid and support of local professors and lab members who toke me in. I would like to thank Prof. Pedro Cuenca, Charo, Javi, José Luis, Jesus, and Rafa from Universidad de Castilla-La Mancha in Albacete. In Boca Raton at Florida Atlantic University I had the great support of prof. Hari Kalva, Velibor, Rashid, Oscar, Reena, Thomas, Keiko, Carolina, and Isabel. I especially would like to thank Bell for taking such good care of me and doing the nice trips during the weekends; your wisdom is endless.

Ik had ook van deze gelegenheid gebruik willen maken om enkele mensen te bedanken die mij de afgelopen jaren in aanraking hebben gebracht met de boeiende wereld van computerwetenschappen en meer bepaald met videocompressie. Ik denk meer bepaald aan mijn voormalige docenten van de Hogeschool Antwerpen (ook gekende als *De Paardenmarkt*) Luc Pieters en Tim Dams om mij de richting te wijzen en prof. Peter Schelkens van de Vrij Universiteit Brussel om mij in een vroeg stadium de beginselen van videocompressie bij te brengen. Dit stelde mij in staat om meer dan 10 jaar lang er iets boeiends van te maken.

Uiteraard zijn er in de afgelopen periode veel mensen met wie ik prachtige tijden beleefd heb. Zo waren er de onvergetelijke momenten, zowel intra- als extramuraal in Antwerpen met Ann, Anke, Loes, Fré, Rob, Hendrik, Kris, Sammy en Glenn. Naast Sammy en Glenn, kon ik ook in Gent rekenen op geweldige mensen waar ik boeiende momenten mee beleefd heb. Anna, Ben, Benjamin, Jef, Jens, Maarten, Pieterjan, Steven, Stijn en Tim bedankt voor de geweldige twee jaar in en rond de Plateau. Maar ook Ma-

rit, Stefanie, Ariane, Nathalie, Bart, Bert en Sander voornamelijk dan rond de Plateau ...

De afgelopen jaren heb ik tijdens mijn doctoraat aan de Zuiderpoort de hulp gekregen van geweldige mensen rond mij. Anna en Glenn, bedankt voor de chocopauzes, het joggen tijdens de middag en bovenal het aanhoren van mijn geniale ideeën. Glenn en Jan bedankt voor de talloze reflectiemomenten, de geweldige standaardisatiemeetings en de lange avonden gevuld met programmeren en papers schrijven op talloze hotelkamers over de hele wereld. Jullie maakten deel uit van de meest intense momenten de afgelopen vier jaar. Een groot deel van het werk in dit boek kon enkel maar gebeuren door jullie hulp. Bedankt voor de geweldige samenwerking.

Uiteraard kan de boog niet altijd gespannen staan en had ik prachtige vrienden om geweldige avonturen mee te beleven. Niet toevallig had Hendrik, naast een paar memorabele Paardenmarkt-verhalen, ook hier regelmatig weer een rol in. Maar ook mijn snowboardvrienden waarmee ik elke winter meerdere malen de bergen opzoek zijn mij ontzettend dierbaar; Bart, Cindy, Dirk, Elke, Koen, Kristof, Sven, Tiny. Graag had ik Els speciaal willen bedanken voor het organiseren van tal van shortski tripjes en Jef om nog steeds een prachtig voorbeeld te zijn en mij continu bij te sturen. En uiteraard Robby, bedankt voor de prachtige pow-pow weekjes.

Dit werk had ik nooit kunnen verwezenlijken zonder de steun van mijn twee dichtste vrienden. Guy en Wim, jullie zijn er altijd voor mij geweest. Samen hebben we prachtige dingen verwezenlijkt en hebben we mooie moment gekend op het water en aan de wal. Maar bovenal, kon ik altijd bij jullie terecht, soms om een avond te praten, en soms ook om meer dan een jaar te komen logeren ...

Tot slot had ik graag mijn familie bedankt en in het bijzonder mijn ouders. Dit doctoraat was enkel mogelijk dankzij de steun van mijn ouders die steeds in mij geloven, die mij steeds mijn eigen richting laten gaan en die steeds klaar staan voor mij. Ik kan enkel maar blij zijn jullie als ouders te hebben.

Gent, juli 2013

Sebastiaan Van Leuven

Table of Contents

Glossary

Symbols

BLK	sub-macroblock type
Did	Dependency layer or spatial layer Identifier
Qid	Quality layer Identifier
Tid	Temporal layer Identifier
BLK_{AVC}	sub-macroblock type of the co-located macroblock in the H.264/AVC bitstream
$MODE_{AVC}$	$MODE$ of the co-located macroblock of the H.264/AVC input bitstream
$MODE$	macroblock mode
$MODE_{BL}$	base layer macroblock mode
$MODE_{EL}$	enhancement layer macroblock mode
$MODE$	macroblock mode
MVd	Motion Vector Difference
QP	quantization parameter
SW	Search Window
SW_{BL}	SW for base layer
SW_{EL}	SW for enhancement layer
μ_{BL}	macroblock type of the co-located macroblock in the base layer
μ_{EL}	macroblock type of a macroblock in the enhancement layer
QP_{BL}	quantization parameter of the base layer
QP_{EL}	quantization parameter of the enhancement layer

A

ATM	AVC based 3D video Test Model
ATM-EHP	ATM-Enhanced High Profile
ATM-HP	ATM-High Profile
AVC	Advanced Video Coding

B

BDPSNR	Bjøntegaard Delta PSNR
BDRate	Bjøntegaard Delta bit rate

C

CABAC	Context Adaptive Binary Arithmetic Coding
CAVLC	Context Adaptive Variable Length Coding
CGS	Coarse Grain quality Scalability
CIF	Common Intermediate Format
CU	Coding Unit

D

dB	decibel

F

FGS	Fine Grain quality Scalability

H

HD	High Definition
HEVC	High Efficiency Video Coding
HM	HEVC Test Model

HTM HEVC based 3D video Test Model

I

ILP Inter-Layer Prediction
ISO International Standardisation Organisation
ITU International Telecommunication Union
ITU-T ITU Telecommunication Standardization Sector

J

JCT-3V Joint Collaborative Team on 3D Video Coding of MPEG and VCEG
JCT-VC Joint Collaborative Team on Video Coding of MPEG and VCEG
JM Joint Model
JSVM Joint Scalable Video Model
JVT Joint Video Team

M

MFC MPEG Frame-Compatible
MGS Medium Gain quality Scalability
MMCO Memory Management Control Operations
MPEG Moving Picture Experts Group
MVC Multiview Video Coding
MVHEVC Multiview Video Coding extension of HEVC

N

NAL unit Network Abstraction Layer unit

P

POC	Picture Order Count
PSNR	Peak Signal-to-Noise Ratio
PU	Prediction Unit

Q

QCIF	Quarter CIF
QoE	Quality of Experience

R

RD	Rate-Distortion
RPS	Reference Parameter Set

S

SAD	Sum of Absolute Differences
SEI	Supplemental Enhancement Information
SVC	Scalable Video Coding

T

TS	Time Saving

V

VCEG	Video Coding Experts Group

English summary

Ubiquitous multimedia consumption has never been as popular as today. Over the last decades multimedia usage has changed dramatically. Digital television has made its appearance into the mass market, Internet television was non-existing in the past, and Video-on-Demand applications are gaining interest. Moreover, people are not only consuming more video than ever before, but the usage is now shifting to a broad range of devices and at different places. Many devices are capable to visualize content, not only at home or at fixed locations, but also in a mobile environment. In the past, the focus was more on the passive consumption of video. Currently, due to easy-to-use video editing applications, more users are capable of producing video themselves. The increasing mobility of the end user, as well as the changing network connectivity properties are posing challenges for current video encoding techniques. These challenges include the reduction of the cost for encoding, transmitting, and adapting the bitstream in the network.

The growing amount of video requires more intelligent coding systems in order to further reduce the bandwidth over existing systems. Moreover, the current trend seems to be that higher resolutions than HD (e.g., Ultra HD or 4K) will make their appearance in the mass market. Furthermore, for specific applications, different encoding architectures are being developed. To allow scalability of bitstreams for different types of mobile devices and varying network characteristics, scalable video coding (SVC) has already been standardized. Since a multitude of encoded video data is still only available in a single-layer structure, like MPEG-2 or H.264/AVC, transcoding bitstreams is an essential approach to cope with scalability. Finally, medium term forecasts expect the introduction of high quality 3D video. Therefore, in order to allow for optimal transmission of such content, 3D video is currently being standardized.

Both the increase of the total bandwidth required for video, and the different domains of future video applications require a high efficiency compression system that allows to efficiently cope with this flexibility. In this work, three application domains (i.e., **SVC encoding**, **transcoding**, and

3D video) are investigated and solutions to reduce the complexity are proposed. This complexity reduction is achieved by reducing the processing power required for the encoding or transcoding process. Moreover, for 3D video, a system is proposed that is easy to design and allows for a short time to market.

Currently, the drawback of the existing SVC architectures is the high encoding complexity. The abstract concept of layers has been introduced in scalable video coding to allow for scalability within the boundaries of these layers. However, each layer requires a different encoding step. Furthermore, for enhancement layers additional inter layer prediction has to be evaluated, which increases the encoding complexity. However, information from previously encoded layers is not used to reduce the complexity of the enhancement layers. In order to reduce the complexity of SVC encoding, a fast mode decision model has been developed based on an analysis of previously encoded bitstreams. This analysis shows the probability for selecting a macroblock type in the enhancement layer based on the macroblock type in the base layer. The model reduces the list of modes that have to be evaluated for the enhancement layer mode decision process based on the selected mode of the co-located macroblock in the base layer. Consequently, the list of evaluated modes for the enhancement layer is reduced such that the complexity is limited significantly. The proposed fast mode decision model only requires 25% of the encoding complexity (which is a reduction of the encoding complexity with 75%), while state-of-the-art fast mode decision models still require around 48% of the total encoding complexity. Furthermore, the same compression efficiency is achieved as these state-of-the-art solutions.

Additional to the fast mode decision model, generic techniques have been proposed. The proposed generic techniques can be combined or used as a standalone technique to reduce the complexity. Moreover, they can be combined with existing fast mode decision models to further reduce the complexity and are applicable in a broad range of encoding algorithms. A study on the impact of the coding efficiency for each proposed generic technique, as well as for a combination of the techniques is carried out. Using these techniques, the complexity is reduced by 88%. Furthermore, the proposed generic techniques are combined with existing optimized techniques to show that the generic techniques also provide solutions for existing optimized systems.

Transcoding is performed in the network to adapt a bitstream to varying conditions of bandwidth or end users devices. In order to reduce the cost of the transcoding process, the complexity should be as low as possible, since a high complexity requires more expensive hardware, but also comes with a higher energy cost. A closed-loop transcoder is proposed, which transcodes an input H.264/AVC stream to a scalable bitstream. This proposed closed-loop transcoder limits the complexity of the encoding process of both the base and enhancement layer by reducing the mode decision process based on the input H.264/AVC mode. The base layer complexity reduction is achieved by exploiting both the H.264/AVC input bitstream information, while for the enhancement layer also the encoded base layer information is used. The proposed technique results in a slight reduction in coding efficiency for the base layer compared to a non-optimized cascaded decoder-encoder. On the other hand, the enhancement layer results in a better prediction compared to this non-optimized cascaded decoder-encoder. By using the H.264/AVC input bitstream, the base layer encoding process is aware of the possible decisions which can be taken for the higher quality enhancement layer. Consequently, the overall compression efficiency is not affected significantly. The complexity reduction for the proposed closed-loop transcoding algorithm is reduced by 91.52%.
A hybrid transcoder, which is achieved by combining the proposed closed-loop transcoder with an existing open-loop transcoding architecture, yields additional complexity reduction compared to the closed-loop transcoder. Meanwhile, drift effects are reduced and scalability is increased for the base layer compared to open-loop transcoding. Moreover, the complexity of this system can be scaled by adjusting the number of open- and closed-loop transcoded frames. Consequently, a complexity reduction between 95.73% and 99.10% of the complexity can be achieved.

Finally, 3D video encoding has been optimized so future video coding standards can encode 3D video more efficiently in order to transmit the encoded data over the network. A hybrid architecture is proposed which allows forward compatibility with existing H.264/AVC systems. The center view is encoded using H.264/AVC, while the side views are encoded with the High Efficiency Video Coding (HEVC) standard. Using HEVC reduces the bit rate for the side views by approximately 50%. The depth information required for autostereoscopic 3D is encoded using a multiview HEVC extension. This yields an easy to design hardware encoder with a significant encoding gain and low complexity. In the future, this architecture can be updated by encoding the center view as HEVC. This results in an easy

to adapt hardware design for the encoder, with a significant gain in coding efficiency and reduction in encoding complexity compared to current solutions.
Moreover, additional tools can be used to exploit redundancies such that the compression efficiency for depth and side views are even further increased. However, in such a situation, HEVC should be a commonly used encoding technology, since in such an architecture no compatibility with monoscopic H.264/AVC is supported. A performance overview shows that the hybrid architecture is able to reduce the bandwidth with approximately 50% compared to H.264/AVC. However, a fully HEVC based system further reduces the bandwidth with 26% for a multi-view HEVC system. The bandwidth can be further reduced to 34.84% if also low-level tools are introduced. On the other hand, allowing low-level tools requires a more complicated hardware design due to the dependencies between texture and depth views.

The proposed low complexity scalable and multiview encoding techniques aim at reducing the encoding complexity or reducing the encoding architecture complexity. The encoding complexity is reduced for SVC enhancement layers by proposing both a fast mode decision model and generic techniques which can be applied with existing optimizations. The proposed transcoding technique reduces both the base and enhancement layer encoding complexity. Moreover, on an architectural level a combination with an open-loop transcoder is suggested. For 3D video, the architectures of different coding techniques are evaluated. By reducing the encoding complexity or the architectural (design) complexity, the generation and transmission of encoded video is optimized. Therefore, the forecasted increase in the amount of video data will not necessarily yield an increase of the energy cost, allowing the growth of mobile video consumption.

Nederlandstalige samenvatting –Dutch Summary–

Multimediale toepassingen zijn alomtegenwoordig en nog nauwelijks uit ons dagelijkse leven weg te denken. Gedurende het laatste decennium is het gebruik van multimediale data drastisch gewijzigd. Zo waren we onder meer getuige van de introductie van digitale televisie, het toenemend gebruik van internettelevisie en de stijgende populariteit van video-op-aanvraag. Daarnaast wordt niet enkel in toenemende mate video geconsumeerd, maar wordt video steeds meer gebruikt op verschillende toestellen en op verschillende plaatsen. Veel toestellen kunnen reeds video afspelen in een mobiele context. Waar vroeger de klemtoon lag op de consumptie van video, stellen eenvoudige toepassingen ons meer en meer in staat om zelf te produceren. De mobiliteit van de eindgebruiker en de wijzigende netwerkeigenschappen zorgen voor nieuwe uitdagingen voor videocompressie.

De stijgende hoeveelheid video vereist intelligentere compressiesystemen. De toenemende vraag naar en productie van video zorgen er immers voor dat de compressiesystemen de totale bandbreedtekost voor video moeten laten dalen ten opzichte van de huidige systemen. Bovendien wijzen de huidige trends erop dat in de nabije toekomst resoluties hoger dan HD (bv. Ultra HD of 4K) bij de eindgebruiker hun intrede zullen doen. Daarnaast zullen voor verschillende applicaties andere encodeerarchitecturen ontwikkeld worden. Schaalbare videocodering (SVC) werd reeds gestandaardiseerd om aanpassingen (schaling) van videostromen toe te laten teneinde de wisselende netwerkeigenschappen en diversiteit van toestellen op te vangen. Aangezien deze standaard nog niet wijdverspreid is, is de meerderheid van geëncodeerde data uitsluitend in een één-lagige structuur beschikbaar, zoals MPEG-2 of H.264/AVC. Om deze stromen aan te passen aan het netwerk of aan de toestellen van de eindgebruiker, is het transcoderen van bitstromen noodzakelijk om met schaalbaarheid om te gaan. Verder wordt op middellange termijn ook de introductie van hoogkwalitatieve 3D-video verwacht, waarvoor momenteel 3D-videocompressietechnieken worden ontwikkeld.

Zowel de toename van de totale bandbreedte benodigd voor video, alsook de verschillende toepassingsdomeinen verlangen hoogwaardige compressiesystemen die toelaten om efficiënt met deze flexibiliteit overweg te kunnen. In dit werk worden mogelijkheden aangereikt om de complexiteit te verlagen voor drie systemen. Voor **schaalbare videocodering**, **transcodering** en **3D-video** wordt de beoogde complexiteitsreductie bekomen in termen van de benodigde rekenkracht van het encodeer- of transcodeerproces. Daarenboven wordt voor 3D-video een systeem voorgesteld dat eenvoudig te ontwikkelen is, teneinde op korte termijn een werkend systeem op de markt te kunnen introduceren.

Momenteel is het nadeel van de bestaande SVC-architecturen de hoge complexiteit voor zowel de encoder als de decoder, dewelke op grote schaal moet gebruikt worden. Het concept van lagen werd geïntroduceerd bij SVC om de mogelijkheid te bieden om binnen deze lagen schaalbaarheid te voorzien. Desalniettemin vereist elke laag een eigen encodeerstap. Daarenboven wordt voor de verbeterings- of uitbreidingslaag een voorspelling tussen de lagen (*inter layer prediction*) geëvalueerd, hetgeen de encodeercomplexiteit bijkomend verhoogt. Echter, informatie van reeds geëncodeerde lagen wordt niet gebruikt om de encodeercomplexiteit te reduceren.
Om de complexiteit van de SVC-encodeerstap te reduceren, werd de beslissingsmethode voor de partitioneringsmodes versneld. Deze versnelling is het resultaat van een analyse van vooraf geëncodeerde bitstromen. De uiteindelijke mode van het macroblok uit de basislaag wordt gebruikt om de lijst van waarschijnlijke modes te reduceren, zodanig dat er minder complexiteit nodig is om de meest optimale mode te selecteren. Het voorgestelde versnelde beslissingsmodel vermindert de complexiteit tot 25% (een reductie met 75%), terwijl de state-of-the-art versnelde beslissingsmodellen ongeveer 52% complexiteitsreductie halen. Daarenboven wordt met het voorgestelde algoritme dezelfde compressie-efficiëntie bereikt als deze state-of-the-art modellen.
Naast het voorgestelde laagcomplexe SVC mode-beslissingsproces, werden generieke technieken voorgesteld die de complexiteit van bestaande systemen verder kunnen reduceren. Deze technieken kunnen toegepast worden in een breed scala van encodeeralgoritmen. Daarbij kunnen ze ook onderling worden gecombineerd om de complexiteit verder te reduceren. De impact op de codeerefficiëntie van elke voorgestelde generieke techniek, alsook van de combinatie van de technieken onderling, werd geanalyseerd. Door deze generieke technieken samen te gebruiken, kan een complexiteitsreductie van 88% bereikt worden.

Transcodering wordt in het netwerk uitgevoerd om een bitstroom aan te passen aan de variërende eigenschappen. Om de kost van het transcodeerproces te reduceren, moet de complexiteit zo laag mogelijk gehouden worden. Immers, een hoge complexiteit heeft nood aan duurdere hardware, maar brengt ook een hogere energiekost met zich mee. Een geslotenlustranscoder werd voorgesteld die een H.264/AVC-invoerstroom omzet naar een schaalbare videostroom. Deze voorgestelde geslotenlusencoder beperkt de complexiteit door het encodeerproces van de basislaag te versnellen aan de hand van gegevens uit de ingevoerde H.264/AVC-bitstroom. Daarnaast worden ook de encodeerbeslissingen van de uitbreidingslaag gereduceerd. Deze reductie wordt bekomen door zowel de beslissingen uit de H.264/AVC-invoerbitstroom alsook de reeds geëncodeerde basislaag te hergebruiken. Daarnaast worden voor zowel de basis- als de uitbreidingslaag optimalisaties doorgevoerd voor het versneld zoeken van de ideale bewegingsvector. De voorgestelde techniek kan resulteren in een kleine afname van de codeerefficiëntie voor de basislaag ten opzichte van een niet-geoptimaliseerde referentietranscoder. Bij een dergelijke referentietranscoder worden de beslissingen van de ingevoerde H.264/AVC-bitstroom alsook de beslissingen van de geëncodeerde basislaag niet gebruikt. Door de H.264/AVC-invoerbitstroom wel te gebruiken, kan de basislaag reeds weten welke beslissingen op de hogere kwaliteitsuitbreidingslaag genomen worden. Hierdoor zal de globale compressie-efficiëntie niet zichtbaar worden aangetast. Met de voorgestelde geslotenlustranscodeertechniek wordt een reductie van 91.52% in complexiteit bereikt. Door deze architectuur te combineren met een bestaande openlustranscoder, wordt de complexiteit verder verlaagd. Daarenboven zal de basislaag minder drifteffecten kennen en wordt de schaalbaarheid van de basislaag verhoogd in vergelijking met openlustranscodering. Daarenboven kan de complexiteit van het systeem worden geschaald door het aantal beelden dat met een open- of geslotenlustranscoder wordt verwerkt te wijzigen. Dit leidt tot een mogelijke complexiteitsreductie tussen 95.73% en 99.10%.

Tot slot werd 3D-videocompressie geoptimaliseerd teneinde toekomstige videocodeerstandaarden toe te laten 3D-video efficiënt te encoderen, zodanig dat deze geëncodeerde data eenvoudig over het netwerk verzonden kan worden. Een hybride architectuur werd voorgesteld die toelaat om compatibiliteit met bestaande H.264/AVC-systemen te vrijwaren. Het centrale gezichtspunt werd geëncodeerd met H.264/AVC, terwijl de zijdelingse gezichtspunten werden geëncodeerd met *High Efficiency Video Coding* (HEVC). Dit reduceert de bandbreedte voor de zijdelingse gezichtspunten met ongeveer 50%. De diepte-informatie noodzakelijk voor 3D-video kan worden geëncodeerd door een multiview-encodering gebaseerd op HEVC toe te passen. Dit leidt tot een eenvoudig aanpasbaar ontwerp van een hardware-encoder; met zowel een significante winst in encodeerprestatie alsook een lage complexiteit ten opzichte van hedendaagse H.264/AVC-gebaseerde multiview systemen. In de toekomst kan deze architectuur geupdatet worden door het centrale gezichtspunt als HEVC te encoderen. Dit zal voor extra compressie-efficiëntie zorgen. Daarnaast kunnen bijkomende mechanismen gebruikt worden om meer redundantie uit te buiten, zodat zelfs de diepte en zijdelingse gezichtspunten efficiënter kunnen worden gecomprimeerd. Maar een dergelijk scenario kan enkel verkregen worden wanneer HEVC een wijdverspreide standaard is, aangezien zulke architecturen geen compatibiliteit met monoscopische H.264/AVC meer ondersteunen. Een performantieanalyse toont aan dat de hybride architectuur in staat is om de bandbreedte te reduceren met ongeveer 50% ten opzichte van H.264/AVC simulcast. Een volledig HEVC-gebaseerd multiview architectuur zal de bandbreedte verder laten dalen met 26% ten opzichte van de voorgestelde hybride AVC-HEVC architectuur. Een bijkomende reductie in bandbreedte van 34.84% ten opzichte van de hybride structuur kan bekomen worden wanneer ook optimalisaties voor onderliggende lagen worden geïntroduceerd. Het nadeel van deze laatste optie is dat het hardwareontwerp complexer wordt ten gevolge van de afhankelijkheden tussen textuur en dieptemappen.

De voorgestelde laagcomplexe videocompressietechnieken voor schaalbare en meerdere-gezichtspunten video richten zich ofwel op het verminderen van de encodeercomplexiteit ofwel op het reduceren van de architecturale kost. De encodeercomplexiteit is verminderd voor SVC-uitbreidingslagen door middel van het voorgestelde snelle modebeslissingsmodel en generieke technieken. De voorgestelde transcodeertechniek vermindert de encodeercomplexiteit van zowel de basis- als uitbreidingslaag. Daarenboven wordt ook op architecturaal niveau een combinatie van een transcoder met

een open en gesloten lus voorgesteld. Tot slot werden verscheidene architecturen voor 3D-videocodering geëvalueerd. Door de encodeercomplexiteit of de architecturale complexiteit te reduceren wordt het produceren en verzenden van geëncodeerde video geoptimaliseerd. Daardoor zal de voorziene toename in videodata niet noodzakelijk leiden tot een toename in de totale energiekost en blijft een stijging van het gebruik van mobiele video mogelijk.

List of Publications

Publications in international journals

1. Sebastiaan Van Leuven, Jan De Cock, Rosario Garrido-Cantos, José Luis Martínez, and Rik Van de Walle, *"Generic Techniques to Reduce SVC Enhancement Layer Encoding Complexity"*, in IEEE Transactions on Consumer Electronics, Vol. 57, nr. 2, pp 827-832, May 2011.

2. Sebastiaan Van Leuven, Glenn Van Wallendael, Jan De Cock, Koen De Wolf, Peter Lambert, and Rik Van de Walle, *"An enhanced fast mode decision model for spatial enhancement layers in scalable video coding"*, in Multimedia Tools and Applications, Vol. 58, nr. 1, pp 215-237, May 2012.

3. Rosario Garrido-Cantos, Jan De Cock, José Luis Martínez, Sebastiaan Van Leuven, and Pedro Cuenca, *"Motion-Based Temporal Transcoding from H.264/AVC-to-SVC in Baseline Profile"*, in IEEE Transactions on Consumer Electronics, Vol. 57, nr. 1, pp 239-246, Feb. 2011.

4. Rosario Garrido-Cantos, Jan De Cock, José Luis Martínez, Sebastiaan Van Leuven, and Antonio Garrido, *"Video Transcoding for Mobile Digital Television"*, in Telecommunication Systems, Springer. (online first: 14 Sept. 2011).

5. Glenn Van Wallendael, Sebastiaan Van Leuven, Jan De Cock, Fons Bruls, and Rik Van de Walle, *"3D Video Compression based on High Efficiency Video Coding"*, in IEEE Transactions on Consumer Electronics, Vol. 58, Nr. 1, pp 137-145, Feb. 2012.

6. Rosario Garrido-Cantos, Jan De Cock, José Luis Martínez, Sebastiaan Van Leuven, Pedro Cuenca, Antonio Garrido, and Rik Van de

Walle, *"On the Impact of the GOP Size in a Temporal H.264/AVC-to-SVC Transcoder in Baseline and Main Profile"*, in Multimedia Systems, Springer. Accepted for future publication.

7. Rosario Garrido-Cantos, Jan De Cock, José Luis Martínez, Sebastiaan Van Leuven, Pedro Cuenca, and Antonio Garrido, *"Scalable Video Transcoding for Mobile Communications"*, in Telecommunication Systems, Springer. Accepted for future publication.

8. Rosario Garrido-Cantos, Jan De Cock, José Luis Martínez, Sebastiaan Van Leuven, Pedro Cuenca, Antonio Garrido, and Rik Van de Walle, *"Temporal Video Transcoding from H.264/AVC-to-SVC for Digital TV Broadcasting"*, in Telecommunication Systems, Springer. Accepted for future publication.

9. Rosario Garrido-Cantos, Jan De Cock, José Luis Martínez, Sebastiaan Van Leuven, Pedro Cuenca, and Antonio Garrido, *"Low complexity transcoding algorithm from H.264/AVC-to-SVC using Data Mining"*, EURASIP Journal on Advanced Signal Processing. Accepted for future publication.

10. Sebastiaan Van Leuven, Jan De Cock, Glenn Van Wallendael, Rosario Garrido-Cantos, and Rik Van de Walle, *"A Hybrid H.264/AVC-to-SVC Transcoder with Complexity Scalability"*, submitted to IEEE Transactions on Consumer Electronics.

Book chapters

1. Sebastiaan Van Leuven, Kris Van Schevensteen, Tim Dams, and Peter Schelkens, *"An Implementation of Multiple Region-of-Interest Models in H.264/AVC"*, in Multimedia Systems and Applications, Signal Processing for Image Enhancement and Multimedia Processing, pp. 214-225, 2008.

2. Rosario Garrido-Cantos, Jan De Cock, Sebastiaan Van Leuven, Pedro Cuenca, Antonio Garrido, and Rik Van de Walle, *"Fast Mode Decision Algorithm for H.264/AVC-to-SVC Transcoding with Temporal Scalability"*, in Lecture Notes in Computer Science, Advances in Multimedia Modeling, Vol. 7131, pp 585-596, 2012.

Papers in international conferences

1. Sebastiaan Van Leuven, Kris Van Schevensteen, Tim Dams, and Peter Schelkens, *"An Implementation of Multiple Region-of-Interest Models in H.264/AVC"*, in Proc. of the Second International Conference on Signal-Image Technology and Internet-Based Systems, pp. 502-511, Jun. 2006, Tunesia.

2. Sebastiaan Van Leuven, Glenn Van Wallendael, Peter Lambert, and Rik Van de Walle, *"Generating Full Length Impaired Movies for Quality of Experience Assessments"*, in Proc. of the 5th Annual EUROMEDIA Conference, pp 56 - 62, Apr. 2009, Belgium.

3. Sebastiaan Van Leuven, Koen De Wolf, Peter Lambert, and Rik Van de Walle, *"Probability analysis for macroblock types in spatial enhancement layers for SVC"*, in Proc. of the IASTED International Conference on Signal and Image Processing 2009, pp. 221-227, Aug. 2009, USA.

4. Steven Verstockt, Sebastiaan Van Leuven, Rik Van de Walle, Elmar Dermaut, Steven Torelle, and Wouter Gevaert, *"Actor Recognition for Interactive Querying and Automatic Annotation in Digital Video"*, in Proc. of the 13th IASTED International Conference on Internet and Multimedia Systems and Applications, pp. 149-155, Aug. 2009, USA.

5. Glenn Van Wallendael, Sebastiaan Van Leuven, Rosario Garrido-Cantos, Jan De Cock, José Luis Martínez, Peter Lambert, and Rik Van de Walle, *"Fast H.264/AVC-to-SVC Transcoding in a Mobile Television Environment"*, in Proc. of the 6th International Mobile Multimedia Communications Conference, Portugal, June 2010.

6. Rosario Garrido-Cantos, Jan De Cock, José Luis Martínez, Sebastiaan Van Leuven, Pedro Cuenca, Antonio Garrido, and Rik Van de Walle, *"Video Adaptation for Mobile Digital Television"*, in Proc. of the IEEE Third Joint IFIP Wireless and Mobile Networking Conference (WMNC), Hungary, Oct. 2010.

7. Rosario Garrido-Cantos, Jan De Cock, José Luis Martínez, Sebastiaan Van Leuven, Pedro Cuenca, Antonio Garrido, and Rik Van de Walle, *"On the Impact of the GOP Size in an H.264/AVC-to-SVC*

Transcoder with Temporal Scalability", in Proc. of the 8th international conference on advances in mobile computing and multimedia (MoMM), Nov. 2010, France.

8. Sebastiaan Van leuven, Glenn Van Wallendael, Jan De Cock, Rosario Garrido-Cantos, José Luis Martínez, Pedro Cuenca, and Rik Van de Walle, *"Generic techniques to improve SVC enhancement layer encoding"*, in Proc. of the 2011 IEEE International Conference on Consumer Electronics (ICCE), pp. 135-136, Jan. 2011, USA.

9. Rosario Garrido-Cantos, Jan De Cock, José Luis Martínez, Sebastiaan Van Leuven, Pedro Cuenca, Antonnio Garrido, and Rik Van de Walle, *"An H.264/AVC to SVC TemporalTranscoder in Baseline profile"'*, in Proc. of the 2011 IEEE International Conference on Consumer Electronics (ICCE), pp. 339-340, Jan. 2011, USA.

10. Sebastiaan Van Leuven, Jan De Cock, Glenn Van Wallendael, Rik Van de Walle, Rosario Garrido-Cantos, José Luis Martínez, and Pedro Cuenca, *"A Low-Complexity Closed-Loop H.264/AVC to Quality-Scalable SVC Transcoder"*, in Proc. of the 17th IEEE International Conference on Digital Signal Processing (DSP), July 2011, Greece.

11. Glenn Van Wallendael, Sebastiaan Van Leuven, Jan De Cock, Peter Lambert, Rik Van de Walle, Joeri Barbarien, and Adrian Munteanu, *"Improved Intra Mode Signaling for HEVC"*, in Proc. of the 2011 IEEE International Conference on Multimedia and Expo (ICME), July 2011, Spain.

12. Sebastiaan Van leuven, Jan De Cock, Glenn Van Wallendael, Rik Van de Walle, Rosario Garrido-Cantos, José Luis Martínez, and Pedro Cuenca, *"Combining Open- and Closed-Loop Architectures for H.264/AVC-TO-SVC Transcoding"*, in Proc. of the 2011 IEEE International Conference on Image Processing (ICIP), pp. 1661-1664, Sept. 2011, Belgium.

13. Rosario Garrido-Cantos, Pedro Cuenca, Antonio Garrido, Jan De Cock, Sebastiaan Van Leuven, Rik Van de Walle, and José Luis Martínez, *"Low complexity adaptation for mobile video environments using data mining"*, in Proc. of the IEEE Fourth Joint IFIP Wireless and Mobile Networking Conference (WMNC), Oct. 2011, France.

14. Glenn Van Wallendael, Sebastiaan Van Leuven, Jan De Cock, Fons Bruls, and Rik Van de Walle, *"Multiview and Depth Map Compression based on HEVC"*, in Proc. of the 2012 IEEE International Conference on Consumer Electronics (ICCE), pp. 168-169, Jan. 2012, USA.

15. Rosario Garrido-Cantos, Jan De Cock, Sebastiaan Van Leuven, Pedro Cuenca, Antonio Garrido, and Rik Van de Walle, *"Fast Mode Decision Algorithm for H.264/AVC-to-SVC Transcoding with Temporal Scalability"'*, in Proc. of 18th International Conference on Advances in Multimedia Modeling (MMM), Jan. 2012, Austria.

16. Rosario Garrido-Cantos, Jan De Cock, José Luis Martínez, Sebastiaan Van Leuven, Pedro Cuenca, Antonio Garrido, *"H.264/AVC-to-SVC Temporal Transcoding using Machine Learning"*, in Proc. of the 16th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES), Sept. 2012, Spain.

17. Rosario Garrido-Cantos, Jan De Cock, José Luis Martínez, Sebastiaan Van Leuven, Pedro Cuenca, Antonio Garrido, *"Temporal Video Transcoding for Digital TV Broadcasting"*, in Proc. of the 5th joint IFIP Wireless and Mobile Networking Conference (WMNC), Sept. 2012, Slovakia.

18. Sebastiaan Van Leuven, Hari Kalva, Glenn Van Wallendael, Jan De Cock, and Rik Van de Walle, *"Joint Complexity and Rate Optimization for 3DTV Depth Map Encoding"*, in Proc. of the 2013 IEEE International Conference on Consumer Electronics (ICCE), pp. 191-192 , Jan. 2013, USA.

19. Sebastiaan Van Leuven, Jan De Cock, Glenn Van Wallendael, Rosario Garrido-Cantos, and Rik Van de Walle, *"Complexity Scalable H.264/AVC-to-SVC Transcoding"* in Proc. of the 2013 IEEE International Conference on Consumer Electronics (ICCE), pp. 328-329, Jan. 2013, USA.

20. Glenn Van Wallendael, Nicolas Staelens, Sebastiaan Van Leuven, Jan De Cock, Peter Lambert, Piet Demeester, and Rik Van de Walle. *"Evaluation of Full-Reference Objective Video Quality Metrics on High Efficiency Video Coding."* in Proc. of IFIP/IEEE International Workshop on Quality of Experience Centric Management (QCMan 2013), May 2013, Belgium.

21. Glenn Van Wallendael, Jan De Cock, Sebastiaan Van Leuven, Andreas Boho, Peter Lambert, Bart Preneel, and Rik Van de Walle. *"Format-Compliant Encryption Techniques for High Efficiency Video Coding"*. Accepted for publication in the proceedings the 2013 IEEE International Conference on Image Processing (ICIP), Sept. 2013, Australia.

22. Luong Pham Van, Jan De Cock, Glenn Van Wallendael, Sebastiaan Van Leuven, Rafael Rodriguez-Sánchez, José Luis Martínez, Peter Lambert, and Rik Van de Walle *"Fast Transrating for High Efficiency Video Coding Based on Machine Learning"*. Accepted for publication in the proceedings of the 2013 IEEE International Conference on Image Processing (ICIP), Sept. 2013, Australia.

23. Johan De Praeter, Jan De Cock, Glenn Van Wallendael, Sebastiaan Van Leuven, Peter Lamber, and Rik Van de Walle. *"Efficient Picture-in-Picture Transcoding for High Effciciency Video Coding"*. Submitted to IEEE International Workshop on Multimedia Signal Processing (MMSP), Sept. 2013, Italy.

Contributions to standardization organizations

1. Glenn Van Wallendael, Sebastiaan Van Leuven, Jan De Cock, and Rik Van de Walle, *"Intra Encoding acceleration by simplification of RDOQ"*, JCTVC-E262, m19789, Mar. 2011, Geneva, Switzerland.

2. Glenn Van Wallendael, Sebastiaan Van Leuven, Jan De Cock, and Rik Van de Walle, *"CE6.b.2: Cross-check report of Combined Intra Prediction with Parallel Intra Coding"*, JCTVC-E263, m19790, Mar. 2011, Geneva, Switzerland.

3. Glenn Van Wallendael, Sebastiaan Van Leuven, Jan De Cock, and Rik Van de Walle, *"CE6.e: Cross-check of Intra Smoothing"*, JCTVC-F411, m20838, Jul. 2011, Torino, Italy.

4. Glenn Van Wallendael, Sebastiaan Van Leuven, Jan De Cock, and Rik Van de Walle, *"CE6.c: Cross-check of Differential Coding of Intra Mode (DCIM)"*, JCTVC-F413, m20840, Jul. 2011, Torino, Italy.

5. Fons Bruls, Glenn Van Wallendael, Jan De Cock, Bart Sonneveldt, Sebastiaan Van Leuven, *"Description of 3DV CfP submission from*

Philips & 'Ghent University - IBBT"', ISO/IEC MPEG, m22603, Dec. 2011, Geneva, Switzerland.

6. Sebastiaan Van Leuven, Fons Bruls, Glenn Van Wallendael, Jan De Cock, and Rik Van de Walle, *"Hybrid 3D Video Coding"*, ISO/IEC MPEG, m23669, Feb. 2012, San Jose, USA.

7. Glenn Van Wallendael, Sebastiaan Van Leuven, Jan De Cock, and Rik Van de Walle, *"Cross-check of Adaptive Resolution Coding (ARC)"*, JCTVC-H0185, m23058, Feb. 2012, San Jose, USA.

8. Glenn Van Wallendael, Sebastiaan Van Leuven, Jan De Cock, and Rik Van de Walle, *"CE5.b: Transform skipping choices based on block parameters"*, JCTVC-H0169, m23050, Feb. 2012, San Jose, USA.

9. Sebastiaan Van Leuven, Glenn Van Wallendael, Jan De Cock, Fons Bruls, Ajay Luthra, and Rik Van de Walle, *"Overview of the coding performance of 3D video architectures"*, ISO/IEC MPEG m24968, Apr. 2012, Geneva, Switzerland.

10. Glenn Van Wallendael, Sebastiaan Van Leuven, Jan De Cock, and Rik Van de Walle. *"Cross-check of Intensity Dependent Quantisation"*, JCTVC-I0495, m25007, Apr. 2012, Geneva, Switzerland.

11. Glenn Van Wallendael, Sebastiaan Van Leuven, Jan De Cock, and Rik Van de Walle, *"Description of scalable video coding technology proposal by Ghent University - IBBT"*, JCTVC-K0049, Oct. 2012, Shanghai, China.

12. Sebastiaan Van Leuven, Glenn Van Wallendael, Jan De Cock, and Rik Van de Walle, *"TE 3: Cross-Check of 4.2.2 Intra prediction based on differential picture"*, JCTVC-L0253, m27594, Jan. 2013, Geneva, Switzerland.

13. Sebastiaan Van Leuven, Glenn Van Wallendael, Jan De Cock, and Rik Van de Walle, *"3D-CE6.h: Cross check of Simplification of Simplified Depth Coding (JCT3V-C0143)"*, JCT3V-C0158, m27916, Jan. 2013, Geneva, Switzerland.

14. Sebastiaan Van Leuven, Glenn Van Wallendael, Robin Bailleul, Jan De Cock, and Rik Van de Walle, *"CE6.h: Cross check*

of CABAC Context Reduction for SDC (JCT3V-D0032)", JCT3V-D0037, m28602, Apr. 2013, Incheon, Korea.

15. Sebastiaan Van Leuven, Glenn Van Wallendael, Robin Bailleul, Jan De Cock, and Rik Van de Walle, *"CE6.h related: Cross Check of Updating Mechanism for Coding of Depth Lookup Table (Delta-DLT) (JCT3V-D0054)"*, JCT3V-D0055, m28833, Apr. 2013, Incheon, Korea.

1 Introduction

Digital video compression has gone a long way the last two decades. In 1991, the Moving Picture Experts Group (MPEG)[1] standardized MPEG-1 [1]. From the introduction of MPEG-1 onward, the amount of digital video has been rising ever since. This increase in digital video required improved encoding schemes. In 1995, MPEG-2 [2] was released, which opened the door for applications such as DVD (Digital Versatile Disc), digital video broadcasting and on-line video. Currently, the de-facto standard for video compression is the widely used H.264 — MPEG-4 part 10: Advanced Video Coding (H.264/AVC), which is a collaboration of the Joint Video Team (JVT) of both MPEG and the Video Coding Experts Group (VCEG) within ITU-T SG16/Q.6. H.264/AVC [3] is targeted towards High Definition (HD) content. As more and more content is becoming available in HD, a huge amount of effort is required to compress the data. Moreover, forecasts indicate that by 2017 global IP traffic will reach 1.4 zettabytes per year, from which 73% will be video content. Currently 60% of the total Internet traffic is video, while 528 exabytes per year is transmitted [4].
These numbers show the need and impact of video compression. Important in this context is the energy consumption associated with digital video. Firstly, energy is required to operate a network and transmit data. Secondly, video encoding is a complex process, which requires a significant amount of energy. Due to the increase in video content volume, the energy

[1] Formally known as ISO/IEC JTC1/SC29/WG11.

consumption will only grow, while the cost of energy is rising. In order to limit the energy cost for the production and transmission of video, it is of utmost importance to reduce the encoding complexity and limit the bit rate required to transmit video data. Compared to MPEG-2, H.264/AVC encoding reduces the bit rate by 60% [5, 6]. Different solutions to reduce the H.264/AVC encoding complexity have already been presented. Most of these optimizations focus on reducing the motion vector search space [7–9] and on the mode decision process [10–12]. Reducing the search range for motion vectors has been a research topic for a long time, and has been well investigated.

1.1 Scalable Video Coding

Besides the increase in IP traffic and video content, another trend is noticeable. An increasing amount of video is not consumed on classical television sets but on alternative devices instead. People have never been more mobile, and they also require watching video in different environments. Not only streaming video to home computers, or using tablets as a secondary screen, also video on mobile devices, such as smart phones, is gaining popularity. This broad range of devices that have to be served requires flexible video compression schemes. The requirements for watching mobile video are different compared to the classical television scenario. Not only a wide range of spatio-temporal resolutions should be supported, but also different types of devices with limited resources (e.g., reduced bandwidth, lower resolution, less battery power or less memory availability) and different complexity constraints. Furthermore, not all of those devices are connected to a high bandwidth network, so bandwidth constraints should also be taken into account. Lastly, not only broadcasted video should be delivered on alternative devices, also video originally intended for mobile devices is now playable on TV sets.
Scalable Video Coding (SVC), which is an extension of H.264/AVC and standardized as annex G of the H.264 standard, was introduced to create a single bitstream that is able to support a wide range of devices and network conditions, such that not a bitstream for each type of device (HDTV, tablet, smart phone) has to be created [13]. With SVC, there is no need to encode different streams for multiple types of devices. Meanwhile, the required total bit rate to provide all these devices with an adequate video stream will be lower compared to encoding a video stream for each device independently. Based on H.264/AVC encoding, SVC allows to encode a video such that a bitstream is created that can be scaled depending on the requirements of the device or the available network capacity.

In order to achieve a scalable bitstream in SVC, firstly a so-called base layer is encoded in H.264/AVC, representing a low resolution and low quality version of the input video stream. The goal is to allow as many end users as possible to watch the video by making the base layer as easily decodable as possible. Secondly, on top of this base layer, enhancement layers are encoded. These enhancement layers allow those devices that are able to receive and decode more than the bare minimum to improve video quality, in accordance to their capabilities. Three types of scalability are supported (i.e., spatial, temporal, and quality), leading to three corresponding types of enhancement layers. Each enhancement layer is placed in a new Network Abstraction Layer Unit (NAL unit)[2]. Depending on the available bit rate or the device capabilities, NAL units are either routed to the end user or dropped in the (congested) network. Even when all packets arrive, the end user device can decide not to decode some enhancement layer packets (e.g., in order to reduce energy consumption).

1.2 3D Video

A last trend in the digital video ecosystem is the increasing immersivity. Users want to be more and more involved in the experience. Surround audio systems are becoming more popular, and most modern televisions have HD resolution. While technology for higher than HD resolutions (Ultra HD, 4K, 8K) is still under development, 3D video is making its way to the consumer market. 3D video allows the user to watch a video and perceive this as a 3D scene, which increases the immersivity. Currently, 3D video using glasses is getting more and more available in the consumer market and in mainstream movie theaters. This technology displays two slightly different images, a left and a right view, corresponding to the images perceived by the left and right eye respectively. The glasses act as a filter to separate the left and right view such that the corresponding view is projected on the correct eye. Because the images correspond to what each eye would have perceived of the scene, our brains interpret the displayed sequences as a 3D video sequence.
The stereoscopic 3D technology has some drawbacks. Firstly, most end users do not want to wear glasses while watching television or movies. Secondly, the perceived 3D scene is only regarded from one viewpoint. This means that moving your head in front of the screen, results in perceiving

[2]NAL units represent the data on a network layer. One unit can be routed over the network independently of other NAL units. Each layer is assigned one or more NAL units and only one enhancement layer can be in the NAL unit.

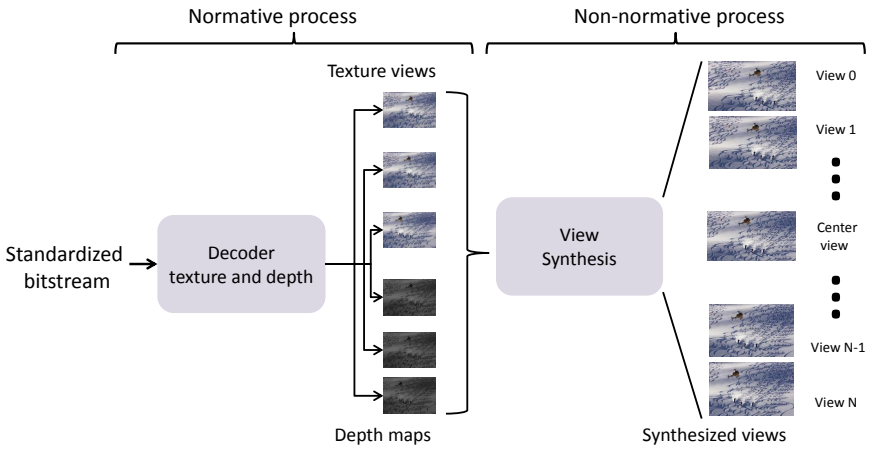


Figure 1.1: Overview of the MPEG standardization work flow indicating the non-normative view synthesis.

the same viewpoint, while in a real 3D environment a different viewpoint is perceived. In order to solve both issues, autostereoscopic displays have been developed. Those displays redirect the light of each view directly to the corresponding eye.

Currently, the Joint Collaborative Team on 3D Video Coding of MPEG and VCEG (JCT-3V) is standardizing a generic approach to encode the HD video data required by the 3D displays. Proprietary standards, such as the S3D format introduced by Philips, are incompatible with different television sets, and might even require different information to be encoded, such as occlusion data. Therefore, on the long term, a standardized solution is preferred. However, only the bitstream will be standardized (and thus be normative). The process of generating all the views will still be proprietary (non-normative) since this will depend on the used display technology and can be one of the differentiators for the perceived quality. Note that within JCT-3V an evaluation of the encoding algorithms has to be performed. This is done by reconstructing the intermediate texture views of the 3D representation using a simple view synthesis process. However, this view synthesis is a non-normative part and is only performed for evaluation purposes. A schematic representation of this distinction between normative and non-normative processes is shown in Figure 1.1.

Transmitting all views of a 3D scene requires a huge amount of data. Therefore, depth maps can be transmitted such that a 3D scene can be generated by using a subset of all the views of a scene. Currently, Multiview Video

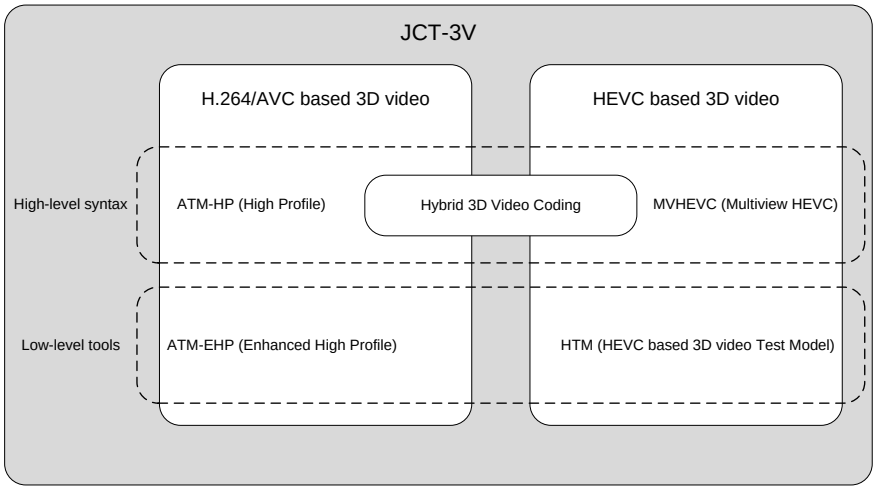


Figure 1.2: Overview of the ongoing JCT-3V standardization work.

Coding (MVC), an extension of the H.264/AVC standard, already allows to encode multiple texture views (up to 256). One texture view (normally the central viewpoint) is encoded using regular H.264/AVC. This viewpoint can be extracted for decoding on regular end user devices that do not require to display the 3D representation. Using MVC, identical syntactical information between views is re-used. Moreover, previously encoded viewpoints can be used as a predictor for the current viewpoint. Since the viewpoints are close to each other, a huge amount of data can be predicted from these previously encoded views. Therefore, the bit rate is reduced significantly, while compatibility with existing systems, software, and hardware is maintained. However, no depth information can be encoded so it is hard to generate an accurate 3D scene representation with viewpoints which are intermediate to the transmitted views. To solve the depth encoding problem, and to evaluate encoding architectures that increase the encoding performance, a next-generation encoding scheme for encoding multiple texture and the corresponding depth views is being defined within the JCT-3V work.

An overview of these activities is shown in Figure 1.2. Two main tracks are followed. Firstly, an encoding system based on H.264/AVC is being investigated. Secondly, an HEVC based system is under development. For each of these approaches, a profile with only high-level syntax and a profile with low-level tool changes are being investigated. The former is basically equivalent to MVC with the inclusion of depth information. However, to reduce the bit rate overhead, the inherent spatio-temporal correlations be-

tween data of the texture views and data of the depth views are exploited in the profiles with low-level changes (ATM-EHP and HTM). In this work, a hybrid architecture is proposed which combines an H.264/AVC base layer with HEVC side views. In this proposed architecture, only high-level syntax changes are required. The proposed architecture is also indicated in Figure 1.2.

1.3 Outline

Given the fact that applications for video compression are more and more differentiating, an increasing number of extensions on standard video compression schemes will be proposed and standardized. This is an easy way to benefit from a widely optimized 2D video encoder, which is adjusted for a well-known functionality. The extensions allow to further increase the importance of a video standard, while the energy consumption will be reduced significantly compared to encoding each independent video stream with a regular 2D video encoder. Therefore, this work focuses on the currently used extensions for video coding, namely, *scalable video coding* and *multiview video coding*.

Given the increasing amount of data, the high energy cost (both to encode the video and to transport the bitstream); the increasing importance of immersivity and the consequently high volumes of data, it is of utmost importance to (re-)generate the bitstreams with an a low energy cost. In this context, SVC encoding reduces the total cost for the bandwidth. Nevertheless the high number of decoders used, the energy consumption at the decoder is mainly dependent on the implementation and used technology. The number of decisions that have to be evaluated at the decoder is limited. Furthermore, post-processing techniques might be used at the decoder side, although, these are non-normative techniques and depend on the manufacturer. On the other hand, SVC encoding is still a computationally complex process. Therefore, a major focus of this research has been the reduction of energy for the encoding process. In this sense, in Chapter 2, an enhanced fast mode decision model for SVC is presented, based on an off-line analysis of SVC encoded bitstreams. Furthermore, generic techniques to optimize the encoding process of SVC bitstreams are presented. These techniques can mostly be combined with the current mainstream optimization techniques (such as fast mode decision models and low complexity motion estimation).

Transcoding from H.264/AVC to SVC is another approach to reduce the overall bandwidth in the network. However, this is also a complex process for which the complexity can be optimized. In Chapter 3, a low-complexity

closed-loop transcoder is presented. This closed-loop system is further improved by designing an architecture which combines the closed-loop transcoder with an existing open-loop transcoder. By doing so, the quality and degree of scalability are improved compared with open-loop transcoding, while compared to closed-loop transcoding, the complexity is significantly reduced.
For 3D video coding, the required bit rate is too high if all required information is transmitted independently. Therefore, architectures to reduce the bit rate are required. Architectures based on HEVC will allow for a low bit rate. However, to maintain compatibility with existing 2D systems, an additional H.264/AVC bitstream has to be encoded. Furthermore, decoders will have both H.264/AVC and HEVC implementations to allow for compatibility with bitstreams from different sources. Therefore, in Chapter 4 a hybrid architecture for encoding three texture views using both H.264/AVC and HEVC is evaluated. This also allows for forward compatibility for existing systems such that the center views can be extracted by conventional H.264/AVC decoders without the requirement to replace those. Meanwhile, bit rate reductions are guaranteed by using HEVC for the texture side views and depth maps.
Finally, in Chapter 5 concluding remarks on the proposed techniques are given.

References

[1] ISO/IEC JTC1/SC29/WG11 (MPEG). ISO/IEC 11172 Information technology Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s. Technical report, ISO/IEC, Nov. 1992.

[2] ITU-T Recommendation H.262 and ISO/IEC 13818-2 (MPEG-2). Generic Coding of Moving Pictures and Associated Audio Information - Part 2: Video. Technical report, MPEG and ITU-T, July 1995.

[3] Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG. Advanced Video Coding for Generic Audiovisual Services, ITU-T Rec. H.264 and ISO/IEC 14496-10 Advanced Video Coding, Edition 5.0 (incl. SVC extension). Technical report, MPEG and ITU-T, Mar. 2010.

[4] Cisco Systems. The zettabyte era. *White Paper*, May 2012.

[5] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan. Rate-constrained coder control and comparison of video coding standards. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7):688–703, July 2003.

[6] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi. Video coding with H.264/AVC: tools, performance, and complexity. *IEEE Circuits and Systems Magazine*, 4(1):7–28, First Quarter 2004.

[7] R. Li, B. Zeng, and M.L. Liou. A new three-step search algorithm for block motion estimation. *IEEE Transactions on Circuits and Systems for Video Technology*, 4(4):438–442, Aug. 1994.

[8] J. Chalidabhongse and C.-C.J. Kuo. Fast motion vector estimation using multiresolution-spatio-temporal correlations. *IEEE Transactions on Circuits and Systems for Video Technology*, 7(3):477–488, June 1997.

[9] J. Y. Tham, S. Ranganath, M. Ranganath, and A.A. Kassim. A novel unrestricted center-biased diamond search algorithm for block motion estimation. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(4):369–377, Aug. 1998.

[10] F. Pan, X. Lin, S. Rahardja, K.P. Lim, Z.G. Li, D. Wu, and S. Wu. Fast mode decision algorithm for intraprediction in H.264/AVC video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 15(7):813–822, July 2005.

[11] D. Wu, F. Pan, K.P. Lim, S. Wu, Z.G. Li, X. Lin, S. Rahardja, and C.C. Ko. Fast intermode decision in H.264/AVC video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 15(7):953 – 958, July 2005.

[12] H. Zeng, C. Cai, and K.-K. Ma. Fast Mode Decision for H.264/AVC Based on Macroblock Motion Activity. *IEEE Transactions on Circuits and Systems for Video Technology*, 19(4):491–499, Apr. 2009.

[13] H. Schwarz, D. Marpe, and T. Wiegand. Overview of the scalable video coding extension of the H.264/AVC standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(9):1103–1120, Sept. 2007.

2

Complexity reduction for scalable video coding

2.1 Introduction to SVC

Scalability is more than ever key to efficiently cope with changing environments. For example, users want to be able to watch television on an HDTV, mobile phone, computer with high bandwidth Internet connection or on a notebook with a low bandwidth wireless connection. Instead of delivering all these streams simultaneously in simulcast, scalable video coding exploits the redundant information between these streams, reducing the bandwidth consumption and thus the operational cost. Scalability has already been introduced in previous standards, the most notable being the Scalable Video Coding (SVC) extension of the H.264/AVC standard. Previous scalability implementations mainly use a multi-loop decoding design, exceptions being MPEG-2 SNR and MPEG-4 FGS scalability. This requires the (low-complexity) decoder to perform multiple decoding steps. Since the complexity of such devices should be limited to reduce the cost and energy consumption, these scalable extensions have never made it into final consumer products. Therefore, single-loop decoding is required, which implies that any layer has to be decodable by using only one motion-compensation loop such that previously encoded layers do not have to be decoded. Consequently, the architectural design of the system is more complicated.

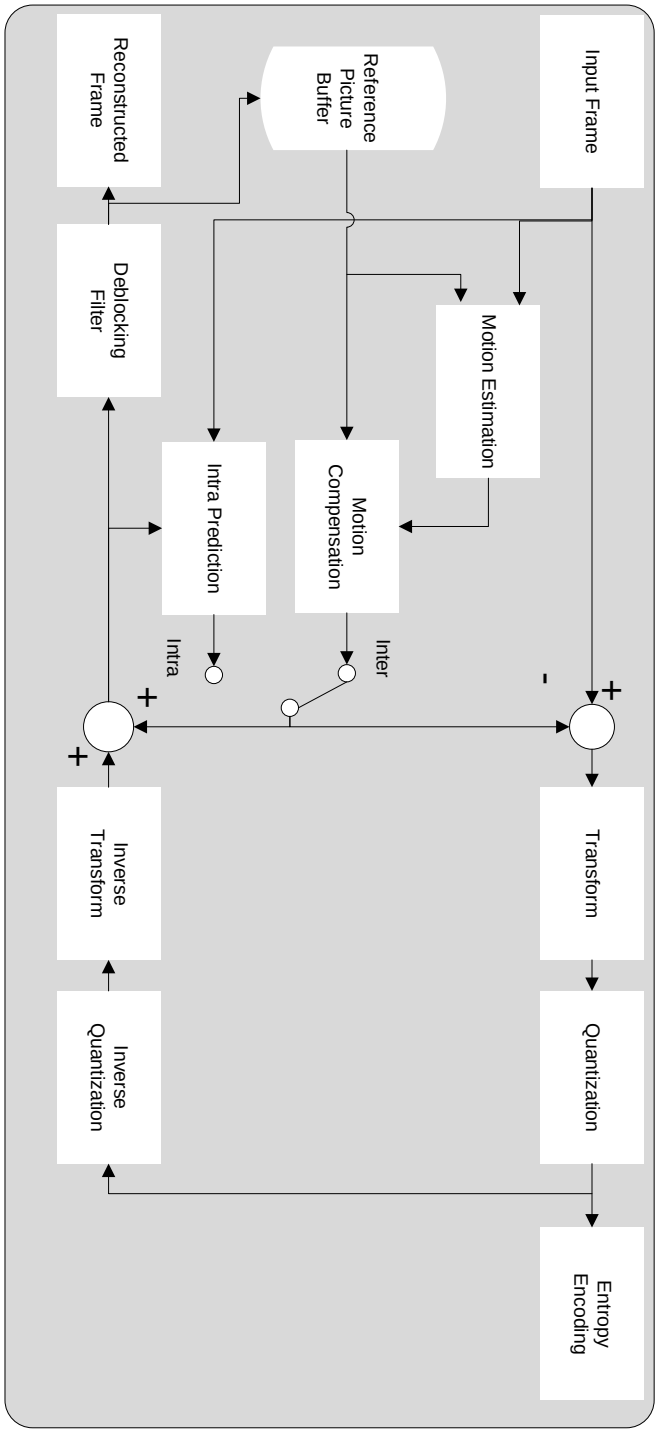


Figure 2.1: Schematical overview of an H.264/AVC encoder.

Figure 2.1 shows a schematical representation of an H.264/AVC encoder [1]. A comparable representation can be seen in Figure 2.2, which represents an SVC encoder [2] for spatial scalability, which is conceptually the most straightforward SVC extension to H.264/AVC. The base layer encoder of Figure 2.2 (bottom part) is identical to the H.264/AVC encoder. However, (upsampled) data from the base layer is used as additional input for the enhancement layer. For the enhancement layer, all building blocks are identical (except for intra prediction), with the difference that upsampled base layer information can additionally be used as prediction. The intra prediction step in the enhancement layer can also use reconstructed base layer samples as a predictor.

For each layer, one motion compensation step is performed. However, the resulting motion compensated image is not used for prediction of the higher layers. This allows to perform only one motion compensation step at the decoder, as can be seen in Figure 2.3. The upsampled residual data is used as a prediction and should not be decoded. The fact that no decoded inter predicted information is required, eliminates the need for motion compensation for each spatial layer.

To reduce the redundancies between layers in SVC, three Inter-Layer Prediction (ILP) techniques are provided such that previously encoded layers can be used as a predictor. Firstly, the mode and motion syntax information from the base layer can be inherited by using the *base_mode_flag* and *motion_prediction_flag*, respectively. If the *base_mode_flag* is true, the enhancement layer partitioning is the same for the base layer[1]. When additionally the *motion_prediction_flag* is true, the macroblock list indices and motion vectors from the base layer are derived. The upscaled motion vectors might not be accurate enough, due to the higher degree of detail in the image. Therefore, a motion vector refinement can be added in the higher spatial layers. In doing so, quarter pixel accuracy is still guaranteed for the high resolution image, yielding a high coding performance. Secondly, intra predicted blocks of the base layer, can be used at the enhancement layer for prediction by signaling the intra base layer mode (I_BL). Thirdly, when the *residual_prediction_flag* switched on true for inter-predicted macroblocks, inter-layer residual prediction is used, which encodes only the difference between the residual signal of the current macroblock and the up-sampled residual signal of the base layer macroblock. The base layer residual data is up-sampled using bilinear interpolation. This operation requires dequantization and inverse transformation but no motion compensation. So the decoding can be done in a computationally efficient way.

[1] If both layers have a different resolution, the partitioning might still be the same. In such cases will the use of a flag reduce the bit rate.

Signaling the *base_mode_flag* as true for intra-predicted macroblocks results in inter-layer intra prediction. This will give the macroblock the designated macroblock type I_BL. The coded base layer is decoded and upsampled to the enhancement layer resolution and a deblocking filter is applied. If required, additional data can be transmitted in the enhancement layer to improve the quality. After decoding the whole picture, again a deblocking filter is applied. Single-loop decoding should also be taken into account for intra-predicted macroblocks in the enhancement layer. Consequently, for inter-layer prediction intra-predicted enhancement layer macroblocks can only use intra-predicted macroblocks from the base layer prediction. Therefore, constrained intra prediction should be applied in the base layer so that intra-predicted macroblocks are independent from inter-predicted macroblock regions. The constrained intra prediction allows for one I_BL macroblock in the enhancement layer to decode different macroblocks from the base layer. However, each of these macroblocks will either have to be predicted from other intra-predicted macroblocks, or when they only have inter-predicted macroblocks as neighbor, those macroblocks will be predicted similar to macroblocks at the border of the image[2] [3]. Note that such texture inter-layer prediction is performed only for intra-predicted macroblocks and that inter-predicted macroblocks can only use the inter-layer residual and motion prediction. This is because only one motion compensation loop is allowed during decoding.

In SVC, three types of scalability features are supported:

Temporal Scalability allows for adapting the frame rate by gradually reducing the number of frames of a sequence. Using hierarchical predictive coding, each frame is assigned a temporal layer. These layers are ordered such that all odd-numbered frames are contained in the highest temporal layer. Predictive coding is only allowed using frames of the same or lower temporal layers. Consequently, the highest temporal layer can be removed without incurring any artifact, such as drift, in the resulting bitstream. The removed frames are evenly distributed in time, resulting in a smooth temporally downsampled video sequence. This mechanism can be applied for both predictive coded frames (P pictures) or bi-predictive coded frames (B pictures). Since this feature is also supported in single layer H.264/AVC, and no additional complexity is required for hierarchical predictive coding. Therefore, the H.264/AVC encoder block diagram (Figure 2.1) is capable of achieving temporal scalability by intelligently manage the reference pictures. The details of temporal scalability are not further elaborated.

[2]Inter-predicted macroblocks neighboring intra-predicted macroblocks can not be decoded in the base layer to comply with the single-loop decoding concept. Pixels of such macroblocks are set to value 128 for intra prediction.

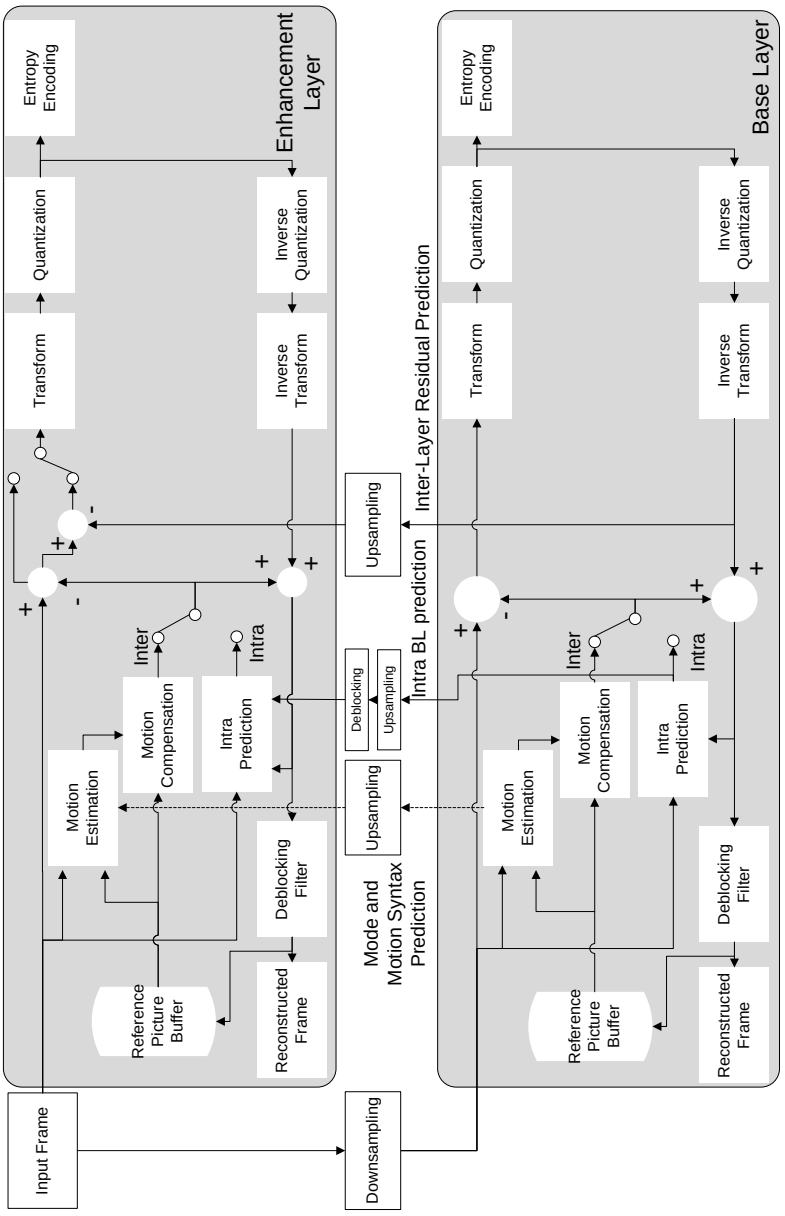


Figure 2.2: Schematical overview of an SVC encoder with spatial scalability.

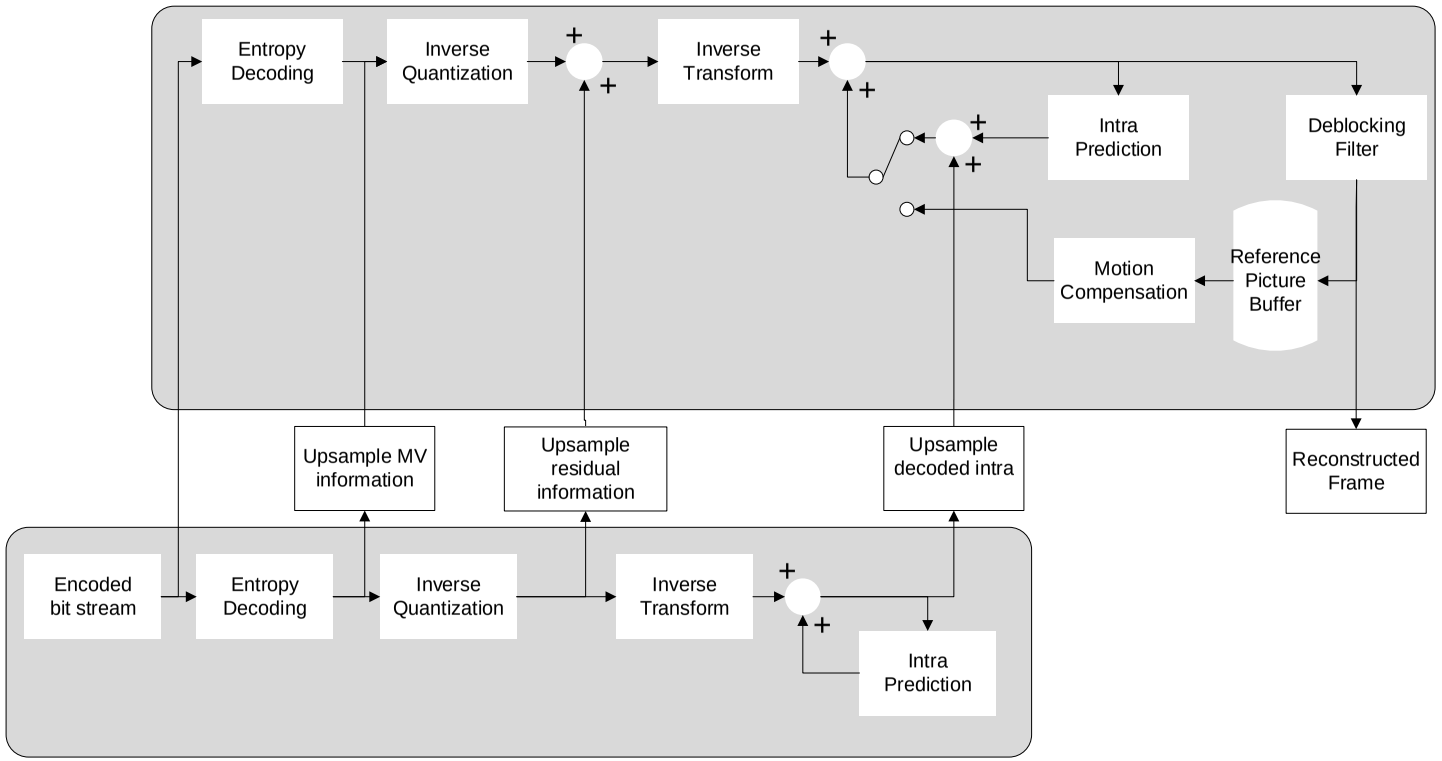


Figure 2.3: Schematical overview of an SVC decoder, where only the highest layer will perform motion compensation.

Quality Scalability adds layers to improve the quality of previously encoded layers. Two types[3] of quality scalability are incorporated in SVC. Coarse Grain quality Scalability (CGS) typically adds a significant higher quality layer on top of the current layer using a spatial enhancement layer with the same resolution. Therefore the encoding block diagram is the same as regular spatial scalability (Figure 2.2). MGS, on the other hand, allows for a finer quality improvement as shown in Figure 2.4. Therefore, MGS is able to accurately scale the bitstream to a corresponding bit rate. MGS allows to devide residual coefficients in different sub-bands. These sub-bands can be transmitted individually and serve as an additional quality on top of previously tranmitted sub-bands. This is a more general approach of the MPEG-2 data partitioning. CGS on the other hand, is implemented as a special case of spatial scalability where the same spatial resolution is used between layers, but a different quantization between both layers is applied. The main difference between CGS and MGS is high-level syntax. However, MGS also allows to divide transform coefficients between slices, allowing for transmitting a slice with a limited number of coefficients and for sending more coefficients in additional slices to improve the quality gradually. Therefore, quality scalability does not demand a high additional complexity compared to single layer encoding. MGS quality layers reuse the selected motion vectors and macroblock modes from the base layer. Moreover, MGS is designed for small quality differences, while such differences are hard to notice. CGS on the other hand requires to evaluate a high number of motion vectors and partitioning sizes. Consequently, CGS can also benefit from optimizations for spatial scalability. Therefore, the complexity reduction will be focused on spatial scalability.

Spatial Scalability allows to generate a bitstream where different resolutions can be extracted [4]. The encoder block diagram is shown in Figure 2.2. Each layer corresponds to a resolution, which can be equal (for CGS) or higher than a previous resolution. No fixed ratio for the resolutions of different layers is defined. Therefore, upscaling a base layer to a higher arbitrary resolution might result in a phase shift. To compensate this phase shift, poly-phase filters are used for upsampling. Depending on the phase shift, different filter coefficients are used. For the residual signal and the chroma components of I_BL a poly-phase two tap filter is applied. The

[3] A third type of quality scalability, Fine Grain quality Scalability (FGS), has been investigated during the standardization of SVC. This allowed to truncate a bitstream at any given point to achieve an extreme fine granularity. However, the syntax overhead to achieve this form of scalability was too high to result in significant gains. Medium Grain quality Scalability (MGS) using the *scan_idx_start* and *scan_idx_end* syntax elements is achieves a similar functionality.

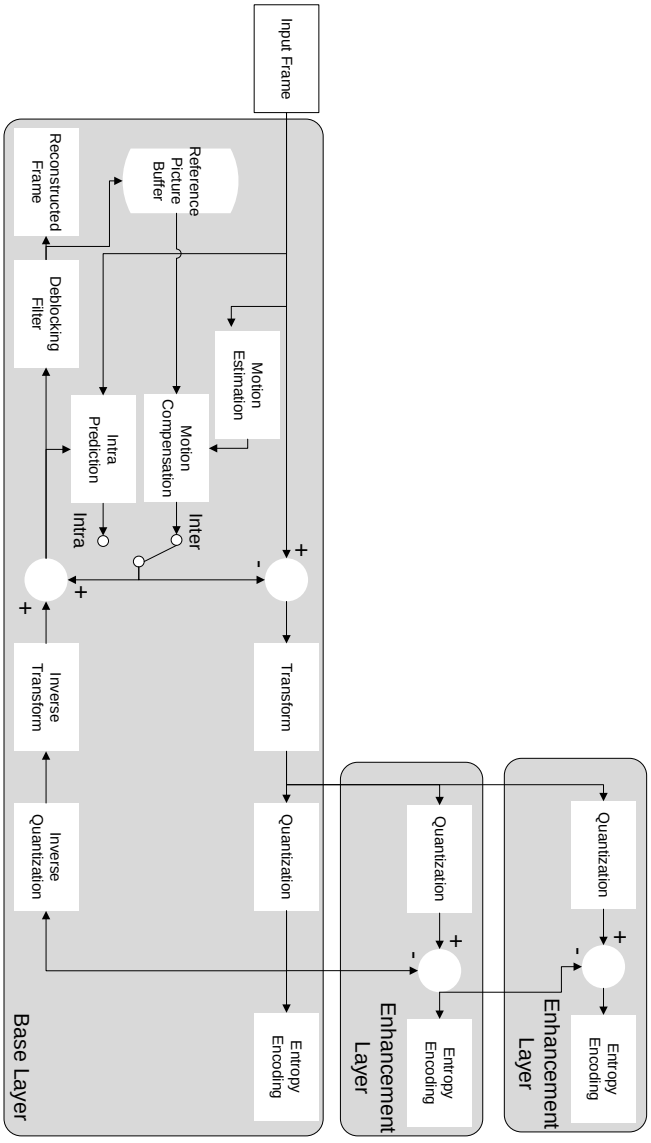


Figure 2.4: Schematical overview of an SVC encoder using Medium Grain quality Scalability.

luma component for I_BL uses a poly-phase four tap filter[4]. The combination of both filters ensures a low complexity, but allows good upsampling results for the luma signal, which has a higher potential to generate visually annoying artifacts.
For encoding a spatial enhancement layer, the same tools and techniques as in H.264/AVC (such as temporal prediction) are also available. In order not to end up with a simulcast system, in which each spatial resolution is encoded separately, additional tools for encoding the enhancement layers are introduced. To exploit similarities between layers, a spatial enhancement layer can be predicted based on a dependency layer, i.e., a previous spatial enhancement layer or the base layer. In these spatial enhancement layers, motion vector information, macroblock type, residual information and samples from previously encoded layers can be derived from already available information using ILP.

To identify to which layer each frame is associated, a layer identifier triplet *(D,T,Q)* is transmitted for every frame. In this triplet, D represents the Dependency layer or spatial layer Identifier (Did), T is the Temporal layer Identifier (Tid) and Q is the Quality layer Identifier (Qid). If multiple layers are stacked, the decoding process is required to have all layers that are referenced by intermediate layers. In the following, the layer which is referenced by the enhancement layer is referred as the base layer, and not necessarily the layer with the lowest Did.
Applications for SVC have not yet met the mass market. Next to a small increase in the decoder complexity, the main reason is the significant increase of the encoding complexity over single-layer H.264/AVC video, due to the layered nature of SVC. Nevertheless, both temporal scalability and quality scalability only slightly increase the encoding complexity, compared to spatial scalability. However, temporal scalability is supported in H.264/AVC without requiring special modifications. Furthermore, quality scalability only allows to slightly modify the bit rate. Meanwhile, the driving force of SVC is spatial scalability, which allows to provide different types of devices with one single bitstream.
For spatial scalability, the encoding complexity is increased significantly. Since it is unknown a priori whether ILP will be beneficial for the macroblock, not only a classic H.264/AVC encoding step is required, but also ILP has to be performed. Without ILP, only an intra-layer inter prediction

[4] [5] Section G.8.6.2.3 Resampling process for Intra_base prediction- Table G-9 contains the 16-phase four tap filter for I_BL upsampling. Section G.8.6.3 describes the resampling process for residual samples.

is applied, which corresponds to a regular temporal prediction using frames from the same dependency layer (*Did*). However, both ILP and intra-layer inter prediction have to be evaluated to decide which method is yielding the lowest Rate-Distortion (RD) cost. Typically, intra-layer inter prediction and ILP require approximately the same complexity. So a rule of thumb is that adding one enhancement layer with the same resolution triples the complexity compared to single layer H.264/AVC encoding at the best quality. It is possible not to perform the ILP, resulting in a simulcast scenario, although a bit rate gain of approximately 10% is observed in [6] due to the use of ILP. Consequently, ILP should not be eliminated during encoding.
In order to reduce complexity, the most complex operations of an encoder are investigated. Optimized algorithms for motion estimation are generally known and widely implemented for a long time [7–11] and are relatively independent of the encoding algorithm. On the other hand, the *mode decision process* invokes the motion estimation process multiple times, while it is also dependent of the encoding algorithm (partitioning, reference list management, etc.). Hence, we will focus on the mode decision complexity for SVC. As can be seen in Table 2.1, the complexity to encode a spatial enhancement layer with a dyadic upscaled resolution requires around 90% of the total encoding complexity. Reducing the base layer complexity will yield some complexity reduction, although this will be limited. Moreover, H.264/AVC encoder optimizations have already been widely studied and can be applied to the base layer. Therefore, reducing the spatial enhancement layer complexity, will significantly reduce the encoding complexity.
The mode decision process ensures that a macroblock is encoded using the RD optimal macroblock type. These types are specified in the H.264/AVC standard [5] and depend on the frame type. A macroblock type defines the way a macroblock is predicted, by defining the partitioning, reference list and the motion vector for each partition. The residual data after motion compensation is signaled for each of these partitions separately. The most important aspects related to macroblock partitioning are highlighted next. The properties of these macroblock modes will be used in the following sections for creating models to reduce the complexity of the macroblock evaluation process.

- Intra-predicted macroblocks are either 4×4 or 16×16 partitioned[5], and use neighboring pixel values from the same frame.

- Inter-predicted macroblocks (P pictures) use one reference list, and the motion vector is always relative to the picture in this list. This

[5] 8×8 partitions are also allowed in High Profile.

Macroblock Mode	Base Layer Complexity (%)	Enhancement Layer Complexity (%)
BL_Skip	-	0.03
Skip	0.01	0.02
Direct	0.01	0.02
16×16	1.13	9.52
16×8	1.18	9.91
8×16	1.32	11.05
8×8	7.01	58.60
Intra_N×N	0.03	0.12
Intra_16×16	0.01	0.03
Total encoding complexity	10.70	89.30

Table 2.1: Overview of the complexity for each macroblock mode relative to the total complexity of the encoder, for dyadic scalability using six sequences (*Harbour*, *Ice*, *Rushhour*, *Soccer*, *Station*, and *Tractor*). The complexity is calculated based on the encoding time for each macroblock mode in both the base and enhancement layer.

motion vector itself is not signaled, but the Motion Vector Difference (MVd) with a predicted motion vector, based on the surrounding motion vectors is signaled. The predictions are limited to *16×16*, *16×8*, *8×16*, and *8×8* partitions. Additionally a P_SKIP macroblock is defined, which is used when the predicted motion vector is used and no residual data is present for an unpartitioned (*16×16*) macroblock. This leads to six macroblock types. Macroblocks with *8×8* partitioning are partitioned into sub-macroblock modes. The four partitions of an *8×8* macroblock can each be sub-divided into *8×8*, *4×8*, *8×4*, or *4×4* sub-macroblock types.

- Bidirectionally inter-predicted macroblocks (B pictures) can use up to two motion vectors for each partition. The same (sub-)macroblock partitions as inter-predicted macroblocks can be used, although, for each partition up to two motion vectors are possible. In case only one motion vector is required, the macroblock type defines which reference list is used for motion compensation. When two motion vectors

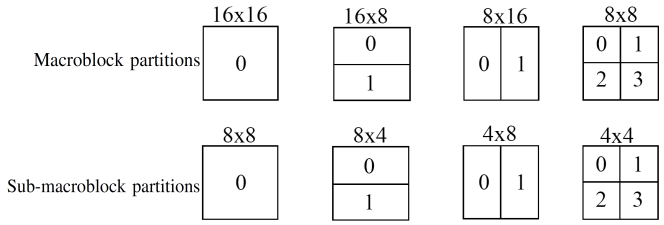


Figure 2.5: Overview of the macroblock partitioning modes.

are used, the motion compensation is done using a (weighted) average of the reference pictures. $B\_Direct\_16\times16$ has been introduced in addition to a B_SKIP. With the $B\_Direct\_16\times16$ macroblock type, no motion vector difference information is signaled, but residual information can be transmitted. Because of the number of partitions, the possibility to use one or two motion vectors, and the list index has, a total of 23 macroblock types have been defined. Furthermore, 13 sub-macroblock types can be used.

For each inter-predicted (sub-)macroblock partition one or two motion vectors can be provided, which indicates the prediction with quarter pixel accuracy. Thus, for any macroblock one to sixteen motion vectors can be provided. Since for each (sub-)macroblock partition a different motion vector can be determined, a search operation is performed for each of these (sub-) macroblock partitions, resulting in a complex operation. Reducing the motion vector search complexity is one of the means to reduce the complexity. More specificically, in Section 2.5 and Chapter 3 the motion vector search is optimized to achieve a lower complexity.
Not all partition sizes are equally important for the mode decision, i.e. some are selected more frequently than others. In the next section, the fast mode decision model is based on these observations. To identify the most optimal macroblock type, the mode decision process is invoked. The mode decision process searches for the most optimal encoding mode of a macroblock by optimizing the RD trade-off. The RD optimization is achieved by minimizing the Lagrangian cost function, given by Equation 2.1 for each macroblock type.

$$J = D + \lambda R \tag{2.1}$$

Here, D represents the distortion between the original and reconstructed signal based on the Sum of Absolute Differences (SAD). R represents the bit rate necessary to encode the macroblock, including the bits to encode

the image data as well as the macroblock type and motion vector information. The Lagrangian multiplier λ is a function of the quantization parameter (QP). The optimal value for λ has been experimentally determined over a large set of different content types in [12]. Each encoder is free to use a different value for λ, since this is non-normative part of the encoder. However, reference software implementations have always used the value proposed in [12].
The mode decision process evaluates for each (sub-)partition size the most optimal motion vector, while all reference lists are evaluated. The mode and reference list yielding the lowest RD is selected and the corresponding macroblock type is used to signal the syntactical information (such as macroblock type and *MVd*) and residual data. So all possible partitions and reference lists are evaluated, resulting in a complex operation. However, these operations can be reduced using knowledge about the scene or already encoded macroblocks. Four main strategies can be followed.

- **Pre-processing techniques** can be used to analyze the content of the picture and determine with a high probability which macroblock types are more likely to be selected by the mode evaluation step [13, 14]. However, such techniques require an adjusted encoder architecture. Additional functionality to analyze the pictures and pre-evaluate partitioning sizes based on the picture statistics has to be available to the encoder core, which additionally requires energy. The encoder mode decision step should be aware of the outcome of the pre-processing step and adjust the evaluation process. Such an approach increases the chip area significantly, and can barely re-use building blocks of the encoder design. Therefore such techniques are not preferred for hardware design.

- **Previously encoded (intra-layer) macroblocks** can assist the mode decision process and limit the list of modes that have to be evaluated. These techniques are mainly developed for H.264/AVC, without taking into account additional available information of SVC. Mostly, such techniques make use of statistical information from co-located macroblocks in previously encoded frames, or from surrounding macroblocks in the same frame [15–17].

- **Selective inter-layer prediction** reduces the spatial enhancement layer encoding complexity by only applying ILP for a limited number of macroblock modes [18].

- **Base layer macroblocks** can also be used to reduce the number of modes to be evaluated. Since SVC results in a renewed encoder ar-

chitecture, new possibilities arise to reduce the mode decision complexity. Firstly, the base layer mode decision information can be used to assist the enhancement layer mode decision process, which will be investigated in the remainder of this chapter. Secondly, motion vector information from the base layer can be used to reduce the motion estimation complexity. This is not further elaborated on in this chapter. However, in Chapter 3 this idea is applied for transcoding.

The first two strategies are also applicable to H.264/AVC and therefore can be used to reduce the base layer complexity. However, as noted before the highest complexity is required for the spatial enhancement layer. Furthermore, as will be pointed out in Section 2.4.9, using an optimized base layer might behave unexpectedly. The third and fourth strategies reduce the SVC enhancement layer encoding complexity. In the third strategy, no encoded base layer information is used. In the fourth strategy the already encoded information is exploited to reduce the macroblock mode evaluations in the enhancement layer. Depending on the applied algorithm, both selective inter-layer prediction and using base layer macroblock information can be combined.
In the following section (Section 2.2) an overview of the related work on encoding the spatial enhancement layer with a reduced complexity is given. Thereafter, in Section 2.3 an analysis of the base layer and the co-located enhancement layer macroblock types is presented. Based on this analysis, a fast mode decision model is derived in Section 2.4. The proposed fast mode decision model is evaluated to identify the complexity reduction. Furthermore, the analysis results in generic techniques that can be used to reduce the spatial enhancement layer encoding complexity (Section 2.5). To indicate the impact on the complexity and coding efficiency, these generic techniques are evaluated as a standalone improvement and as an improvement on existing fast mode decision models. Finally, Section 2.6 has concluding remarks on the complexity reduction for scalable video coding.

2.2 Related work

Extensive research has been done in the field of rate-distortion and complexity optimization for H.264/AVC. Both intra- and inter-prediction have been optimized. In [19] the intra-prediction complexity is reduced based on local edges, while [20] reduces the modes by taking into account the frequency characteristics of the transformed 4×4 partitions. The complexity for inter-predicted macroblocks is reduced in [10], which evaluates a stop criterion to reduce the number of evaluated modes. Furthermore, the

complexity is reduced by limiting the motion estimation process, and use a less optimal motion vector. Based on spatio-temporal characteristics, the motion estimation process is reduced in [11] for a limited number of partitions. These and many more algorithms have been optimized for single-layer H.264/AVC. Furthermore, hardware implementations of most algorithms require pre-processing which is not part to the encoding scheme, therefore requiring more silicon space and thus a higher production and design cost. Nevertheless, these techniques are valuable to reduce the base layer encoding complexity.

Additional base layer information such as motion vectors and macroblock partitioning is available in SVC, which allows to easily reduce the enhancement layer encoding complexity. This does not require a significant change in hardware design and results in a low design complexity. Furthermore, reducing the enhancement layer complexity based on the base layer still allows low complexity techniques for the base layer to be applied. Consequently, an encoder can be designed with these optimizations for the base layer, while new techniques reduce the enhancement layer encoding complexity.

Based on the macroblock type of the co-located base layer macroblock, the set of macroblock types that has to be evaluated in the enhancement layer can be reduced. This idea has previously been proposed for Coarse Grain Scalability (CGS) by [21–23]. Increasing the spatial enhancement layer resolution will increase the complexity significantly. However, it is not stated how these methods perform in a spatial scalability scenario, which is used in the following sections.

For spatial scalability, a method based on the neighboring macroblocks has been proposed in [24], reporting an average time saving of 44.81%. However, no encoded base layer information, such as macroblock types, has been used in the spatial enhancement layer mode decision process. So, the main benefit of SVC encoding, the availability of the base layer information, has not been exploited. Therefore, even higher time savings are feasible, as shown by [25] and [26]. These methods present a classification mechanism for the most probable modes, based on the neighboring base layer modes, resulting in time savings around 65%, with a reported bit rate increase of only 0.17%. A similar approach is used in Section 2.4, yielding 75% complexity reduction.

In [27], time savings are achieved by prioritizing the macroblock modes based on the base layer macroblock type. An early termination strategy is applied based on the state (i.e., all-zero block) of the current macroblock and neighborhood macroblocks. This technique works for CGS as well as for spatial scalability and results in low bit rate increase and low quality

degradation. However, the low reported average time savings, of 20.23% for CGS and 27.47% for spatial scalability, as well as the fact that only dyadic spatial scalability is supported, makes this technique less suited as a stand-alone technique. The small bit rate increase of around 1.6% makes this a good technique to combine with existing fast mode decision models. Because of the low complexity saving presented in [27], this technique will not be further evaluated.

A different early termination strategy based on the base layer information is proposed in [28] and [29]. Based on the neighboring macroblocks of the base and enhancement layer, the mode decision process is optimized. Unfortunately, the presented results only show time savings around 30%. A very effective and simple scheme is presented in [18]. Here, selective inter-layer prediction is proposed. First, all modes are evaluated without inter-layer residual prediction. Thereafter, for the best mode, also inter-layer residual prediction is evaluated. This way, roughly half of the calculations have to be performed, yielding a reported 40% time saving. All modes still have to be evaluated, making this technique ideal for extending existing fast mode decision models. In Section 2.5 the encoding complexity of the proposed generic techniques has been further reduced due to a combination with this selective inter-layer prediction.

The most advanced fast mode decision models for spatial scalability, in terms of RD performance and complexity reduction, can be found in [30] and [31]. These reduce the number of enhancement layer mode evaluations based on the macroblock mode of the base layer. The reduction is achieved by eliminating modes that are not likely to be selected based on an off-line analysis of encoded video streams. After implementing the proposed models, it is noticed that the results for [30], referred to as Li's model, outperform the results for [31]. This idea is similar to the proposed model in Section 2.4, which is also based on an off-line analysis. In the results section, it will be seen that the model proposed in Section 2.4 outperforms [30]. Moreover the proposed model shows to be less affected by the quantization of both layers and yields more stable complexity reductions.

Recently, [32] proposed a method where an all-zero block detection is used for an early termination of motion estimation and mode decision. In [33] a statistical model is presented, based on fixed conditional probabilities without identifying the appropriate quantization levels. Both models compare their results with Li's model and show slightly improved performances. Therefore, to compare the state of the art with the proposed algorithm in Section 2.4, Li's model will be used as a reference. Not only is Li's algorithm compared with the proposed technique, but also their presented measurements are extended. Firstly, it is unclear whether Li et. al. used the

sequences for the modeling also for evaluating their model. Secondly, additional experiments with Li's model are performed because only one single spatial scalability scenario is tested and the same quantization for base and enhancement layer is used.
In prior work [34], a basic analysis on the probabilities for the macroblock types in spatial enhancement layers is given. This analysis takes into account the occurrence of the macroblock types in the enhancement layer for different quantizations of both layers. However, the analysis does not consider the macroblock type of the co-located macroblock in the base layer. Furthermore, it does not employ different resolutions for base and enhancement layer. Therefore, an elaborate analysis of the enhancement layer macroblock type probability based on the base layer macroblock types is required to create a fast mode decision model.

2.3 Enhancement layer macroblock type analysis

In this section a profound analysis of the macroblock type of a macroblock in the enhancement layer (μ_{EL}) versus the macroblock type of the co-located macroblock in the base layer (μ_{BL}) is presented. The quantization parameter of the base layer (QP_{BL}), quantization parameter of the enhancement layer (QP_{EL}), resolutions of base and enhancement layer and μ_{BL} are taken into account. The μ_{EL} highly depends on the visual content in the macroblock, and consequently it is highly correlated with the μ_{BL}. However, it can be observed that also a different quantization of both layers will influence the most optimal μ_{EL}. First, the methodology of the analysis is explained, thereafter the analysis is given.

2.3.1 Methodology

The analysis has been performed using five test sequences (i.e., *Bus*, *Foreman*, *Mobile*, *Crew*, *City*). These five sequences represent different kinds of motion and texture, such that the conclusions of the analysis can be extended to a wide range of sequences. All encoded sequences contain two spatial layers, a base layer and one spatial enhancement layer. The first three sequences have a Quarter CIF (QCIF) resolution for the base layer, while the enhancement layer has a Common Intermediate Format (CIF) resolution. The last two sequences have a CIF resolution at the base layer, and a 4CIF resolution for the enhancement layer.
To analyze the impact of the quantizers, for each sequence, a number of streams were generated with varying QP_{BL} and QP_{EL}. For both QP_{BL} and QP_{EL} holds: QP_{BL}, $QP_{EL} \in \{12, 15, 18, 21, 24, 27, 30, 33\}$. For all

sequences, each combination QP_{BL} and QP_{EL} is encoded, noted as the ordered pair *(*QP_{BL}*,* QP_{EL}*)*, which leads to 64 combinations for one sequence. In practical situations $QP_{BL} < QP_{EL}$ frequently occurs when the increased resolution has a lower quality. There are rarely any practical applications for sequences where $QP_{BL} \ll QP_{EL}$ (i.e., the quality of the enhancement layer is significantly reduced compared to the base layer), although these streams are included in the analysis for completeness of this study and will help to understand the mechanism behind the enhancement layer mode selection.

For each combination of (QP_{BL}, QP_{EL}), 64 frames have been encoded using the Joint Scalable Video Model (JSVM) reference software version 9.10 [35], with an intra period of 32 frames. For analyzing the μ_{EL} a distinction is made between P and B pictures. Each combination of each sequence is encoded once using only P pictures and once with B pictures. The latter sequences have a GOP size of 16 frames. This results in a total of 128 encoded streams for each test sequence. Each layer has the same temporal resolution of 30 fps, adaptive ILP is used for enhancement layers. Context Adaptive Binary Arithmetic Coding (CABAC) is used as entropy coding mode.

Intra-coded pictures will not be discussed, firstly they have a low complexity and optimizing them will not result in a significant complexity reduction. Secondly, it is observed that typically only for 2-4% of the total number of macroblocks in intra-predicted pictures one has $\mu =$ *I_16×16*, whereas for all other macroblocks $\mu =$ *I_4×4*. This indicates that a profound analysis of the intra-coded pictures is not necessary. As a result, only inter-coded pictures (i.e., P and B pictures) are discussed.

The analysis performed in the remainder of this chapter is assisted by graphs that visualize the probability of each (μ_{BL}, μ_{EL})-pair. In these graphs, each bar represents the conditional probability for a random macroblock that μ_{EL} is selected, based on the a priori knowledge of the μ_{BL} of the co-located macroblock in the base layer. This conditional probability is expressed as Equation 2.2.

$$p = P(\mu_{EL}|\mu_{BL}) \tag{2.2}$$

For each μ_{BL}, the sum the probabilities of a row (all points with a constant μ_{BL}) is 1 or 0. In case the sum is 0, no base layer macroblock is encoded using μ_{BL}. When μ_{BL} is used, the sum of the probabilities that any μ_{EL} will be selected given the μ_{BL} will be 1. The macroblock type number is indicated on both axes and corresponds to those used in the H.264/AVC

μ	macroblock type name
-1	B_SKIP
0	$B\_Direct\_16\times16$
1	*B_L0_16×16*
2	*B_L1_16×16*
3	*B_Bi_16×16*
4-20 *	$B\_Lx\_Ly\_16\times8^\dagger$
5-21 *	$B\_Lx\_Ly\_8\times16^\dagger$
22	$B\_8\times8$
23	*I_4×4*
24-47	*I_16×16*‡
48	I_PCM

* even numbered
* odd numbered
† Lx and Ly can be $L0$, $L1$ or Bi
‡ each type differs in coded block pattern

Table 2.2: Summarization of the macroblock types for B pictures.

specification[6]. $\mu = -1$ is added, which represents the inferred and non-coded macroblocks (i.e., P_SKIP mode in P pictures and B_SKIP mode in B pictures). All macroblocks with BL_SKIP are considered to have the same macroblock type as defined in the base layer. Therefore the probability for BL_SKIP is incorporated in the μ_{EL} probability. For completeness, Table 2.2 and Table 2.3 summarize the names of the macroblock type for each μ in B pictures and P pictures, respectively.

2.3.2 Analysis results

Figure 2.6 and Figure 2.7 show some of the graphs created after analyzing all streams. As will be discussed, the sequences *Bus*, *City*, *Foreman*, and *Mobile* show the same characteristics, therefore for only one of these sequences a graph is shown to highlight the properties. The mutual trends can be seen throughout the different graphs of any of these sequences. The sequence *Crew* on the other hand has different characteristics, so in order to depict these different characteristics, the graphs of *Crew* are included explicitly. Note that these findings do not only apply to the *Crew* sequence, but these characteristics correspond to the type of content. Consequently,

[6] [5] Section 7.4.5 Macroblock layer semantics - Table 7-14.

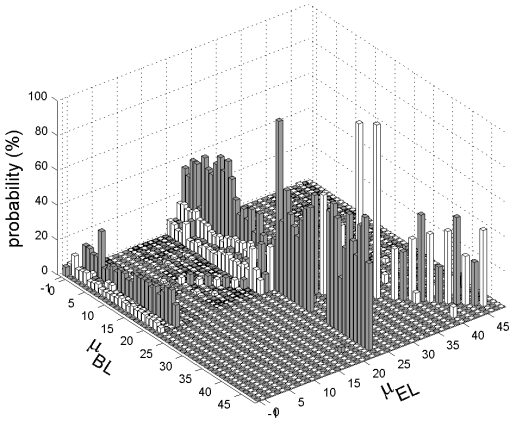
μ	macroblock type name
-1	P_SKIP
0	*P_L0_16×16*
1	*P_L0_L0_16×8*
2	*P_L0_L0_8×16*
3	*P_8×8*
4	*P_8×8*$ref0$
5	*I_4×4*
6-29	*I_16×16*‡
30	I_PCM

‡ each type differs in coded block pattern

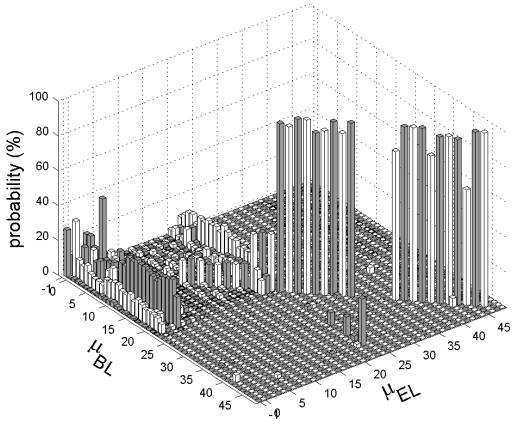
Table 2.3: Summarization of the macroblock types for P pictures.

in real-life situations, many other sequences might be found to have such characteristics. However, these sequences were not incorporated in this analysis.
Encoded streams of the sequences *Bus*, *City*, *Foreman*, and *Mobile* have similar characteristics while they have different resolutions. This indicates that the resolution of the layers does not have an influence on the probability of μ_{EL}. On the other hand, the graphs for sequence *Crew* show a different layout compared to the other sequences, while both layers have the same resolutions applied as sequence *City*. This can be seen in Figure 2.6, where only the sequence *Crew* (Figure 2.6(a) and 2.6(b)) has *I_16×16* coded macroblocks in the base layer ($\mu_{BL} = 24, \ldots, 48$), whereas sequence *Foreman* (Figure 2.6(c)) shows that none of the base layer macroblocks is intra-coded.
Even though for the *Crew* sequence less than 2% of the base layer macroblocks in inter-coded pictures are *I_16×16* coded, interesting findings are observed when $\mu_{BL} =$ *I_16×16*.

- First, in most situations $\mu_{EL} = \mu_{BL}$ holds true (Figure 2.6(a) and 2.6(b)), which is noticed by the diagonal line in the bottom right of the graphs ($\mu = 24, \ldots, 48$).
- Second, when the quality of the enhancement layer drastically increases compared to the base layer ($QP_{EL} < QP_{BL} - 9$), μ_{EL} can be *I_4×4* as well (Figure 2.6(a)).

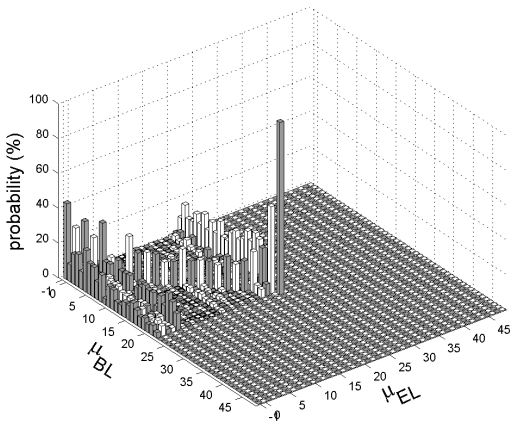


(a) *Crew* $(QP_{BL}, QP_{EL}) = (24, 12)$



(b) *Crew* $(QP_{BL}, QP_{EL}) = (18, 18)$

Figure 2.6: Correlations of μ_{BL} and μ_{EL} for B pictures (including $\mu_{BL} =$ *I_16×16*).



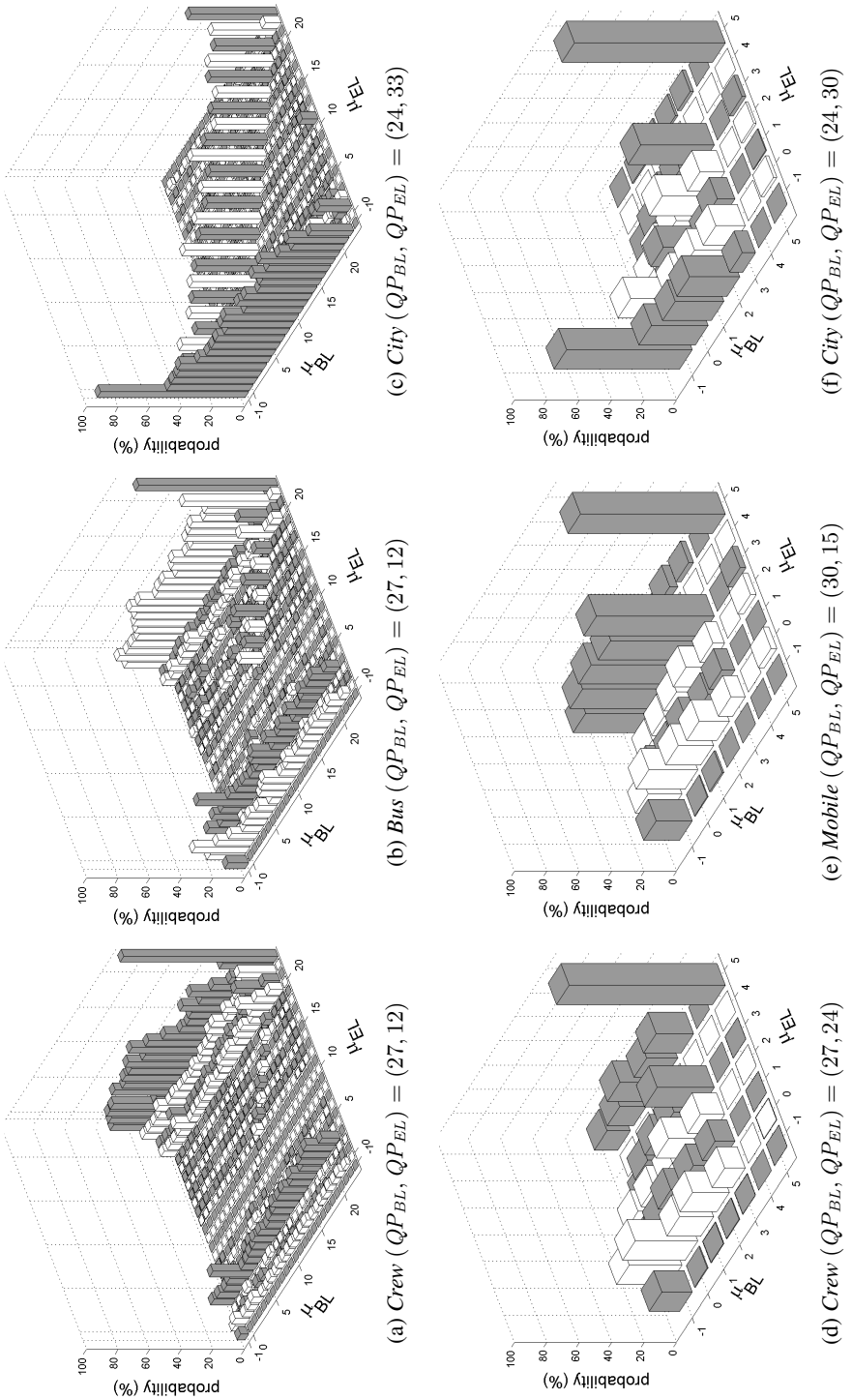
(c) *Foreman* $(QP_{BL}, QP_{EL}) = (21, 18)$

Figure 2.6: Correlations of μ_{BL} and μ_{EL} for B pictures (including $\mu_{BL} =$ *I_16×16*) (cont.).

The latter is caused by the increase in the degree of detail in the enhancement layer because of the increase in quality and resolution. More details will result in extra residual data if $\mu_{EL} = \mu_{BL}$ and poor prediction results for inter prediction, consequently more macroblocks are *I_4×4* coded.
For inter-coded pictures, a high correlation in macroblock type can be seen for both P and B pictures, but also differences between both types can be observed due to the nature of both inter-coded frame types. In the following, first the findings which count for all inter-coded pictures will be discussed, thereafter a distinction is made for B pictures (Section 2.3.3) and P pictures (Section 2.3.4).
A general observation for all graphs, is a diagonal of significant values. This diagonal corresponds to all probabilities d following Equation 2.3.

$$d = P(\mu_{EL} = \mu_{BL} \vee \mu_{EL} = BL\_SKIP\ ). \tag{2.3}$$

$P(\mu_{EL} = BL\_SKIP)$ is included in the diagonal since the base mode is inferred with BL_SKIP (by signaling *base_mode_flag* as described in Section 2.1). The exact influence of this diagonal on the selection probabilities depends on both QP_{BL} and QP_{EL}. In Figure 2.6(a) it can be seen that the probability for $\mu_{BL} = \mu_{EL}$ is significantly lower compared to Figure 2.6(b). The diagonal is more significant when $QP_{EL} \geq QP_{BL}$ because the higher quantization of the enhancement layer results in a higher impact



(a) *Crew* $(QP_{BL}, QP_{EL}) = (27, 12)$ (b) *Bus* $(QP_{BL}, QP_{EL}) = (27, 12)$ (c) *City* $(QP_{BL}, QP_{EL}) = (24, 33)$

(d) *Crew* $(QP_{BL}, QP_{EL}) = (27, 24)$ (e) *Mobile* $(QP_{BL}, QP_{EL}) = (30, 15)$ (f) *City* $(QP_{BL}, QP_{EL}) = (24, 30)$

Figure 2.7: Correlations for μ_{BL} and μ_{EL} for inter-coded macroblocks. (a)-(c): B pictures (d)-(f): P pictures.

of the syntax bits compared to the residual data (due to the RD optimization of $J = D + \lambda R$). Therefore, using ILP with the *base_mode_flag* is more efficient than signaling a new μ. Nevertheless, in every situation this diagonal can be considered as an important aspect.
As a special case of this diagonal property, μ= *I_4×4* has to be considered. In almost all graphs it can be seen that if μ_{BL} = *I_4×4* then μ_{EL} = *I_4×4* with a probability almost equal to 1[7]. This might be less explicit when the quality of the enhancement layer decreases compared to the base layer, but the probabilities for alternative types are highly distributed and insignificantly low.
A last general observation only applies to the *Crew* sequence, most likely because of the content properties. For both P and B pictures, it is observed that for inter-predicted macroblocks in the base layer ($\mu_{BL} = -1, \ldots, 22$), μ_{EL} = *I_4×4* has a significant probability as long as $QP_{EL} \leq QP_{BL}$. These findings are observed in Figure 2.6(a) and 2.6(b) where $\mu_{EL} = 23$.
To continue this discussion, P and B pictures are distinguished. In the remainder of this chapter detailed versions of the graphs are provided to enhance the readability. In particular, the correlations for *I_16×16* macroblock types are excluded from Figure 2.7, as all related findings have been tackled before. The graphs for P pictures (Figure 2.7(a) - 2.7(c)) show less macroblock types, since for P pictures less macroblock types are defined compared to B pictures (Figure 2.7(d) - 2.7(f)).

2.3.3 B pictures

- An overall observation for B-picture graphs (Figure 2.7(a) - 2.7(c)) is the frequent occurrence of two macroblock types in the enhancement layer, irrespective of μ_{BL}, (QP_{BL},QP_{EL}) or the resolution of base and enhancement layers.

These types coincide with $B\_Direct\_16\times16$ ($\mu = 0$) and *B_Bi_16×16* ($\mu = 3$). This can be seen both in Figure 2.6 and Figure 2.7(a) - 2.7(f), where these types (in the enhancement layer) span a complete column, which means that these types are frequently selected independently of μ_{BL}. The reason for their high occurrence in all sequences can be found in the fact that both are non-partitioned and both use the weighted average of two reference pictures for prediction. Being non-partitioned has increased their occurrence in the enhancement layer because of the increase in resolution of the enhancement layer. Furthermore, using two reference pictures yields mostly a better prediction compared to one reference picture. For

[7]Following Equation 2.2 this can be epxressed as: $P(I\_4\times4|I\_4\times4) \approx 1$.

the latter reason, the *B_Bi_16×16* macroblock type occurs more frequently compared to other non-partitioned and single-reference macroblock types (*B_L0_16×16* and *B_L1_16×16*).

- Partitioned macroblock types become more important in the enhancement layer when $QP_{EL} \leq 24$.

This can be seen by comparing Figure 2.7(a) and 2.7(b) with Figure 2.7(d). Due to the increase in resolution combined with a low QP_{EL}, the visual detail of the image increases. Such details are better compressed when a macroblock can be partitioned to improve the prediction. Particularly the partitioned macroblock types *B_Bi_Bi_16×8* ($\mu = 20$), *B_Bi_Bi_8×16* ($\mu = 21$), and *B_8×8* ($\mu = 22$) have a high occurrence, independent of μ_{BL}. The latter (*B_8×8*) is significant by the fact that small block sizes are possible (i.e., *4×4*, *4×8*, *8×4* and *8×8*). Meanwhile, *B_Bi_Bi_16×8* and *B_Bi_Bi_8×16* are more significant compared with the single reference macroblock types with the same partitioning (*16×8* and *8×16*; ($\mu = 4, \ldots, 18$)). The bi-predictive types are favored due to the better results of the weighted average prediction of the reference pictures.

- A direct relationship between the number of skipped macroblocks and the QP_{EL} is observed.

When $QP_{EL} \geq 18$, about 10% of the macroblocks in the enhancement layer have $\mu_{EL} = B\_SKIP$ ($\mu_{EL} = -1$), independent of μ_{BL} (illustrated by Figure 2.6(c) and Figure 2.7(c)). This value increases when QP_{EL} increases, due to the quality decrease in the enhancement layer, more macroblocks in the enhancement layer are B_SKIP-coded. As much as 35-40% of all enhancement layer macroblocks are coded B_SKIP when $QP_{EL} \geq 30$ and $QP_{BL} \ll QP_{EL}$ (as depicted in Figure 2.7(c)). This phenomenon is explained by the larger quantization of the enhancement layer, which reduces the details of both the reference picture for the enhancement layer and the current frame of the enhancement layer. Consequently, residual data is unnecessary and the predicted motion vector will result in an RD optimal predictor. Note that only when $QP_{EL} \gg QP_{BL}$, the impact of *B_Bi_16×16* is reduced, in favor of B_SKIP mode, as can be seen in Figure 2.7(c).

2.3.4 P pictures

- *P_8×8ref0* ($\mu = 4$) will not occur in the encoded streams, since CABAC entropy coding is used.

Therefore, no probabilities for $\mu = 4$ will appear in the P-picture graphs (Figure 2.7(d) - 2.7(f)). Note that *P_8×8ref0* is solely used for Context Adaptive Variable Length Coding (CAVLC); this is purely a change in signaling and does not affect the mode decision.

- As with B pictures, the number of P_SKIP macroblocks in the enhancement layer is significant when $QP_{EL} \geq 24$, except for *Crew*.

This is observed in Figure 2.7(d). When the base layer macroblock is P_SKIP, it can be seen that there is a high probability (0.7 - 1) that $\mu_{EL} = P\_SKIP$ when $QP_{EL} \geq QP_{BL}$.

- Macroblock types *P_L0_16×16* ($\mu = 0$) and *P_8×8* ($\mu = 3$) occur frequently in all sequences independent of any parameter.

Because of the increase in resolution, *P_L0_16×16* occurs more frequently thanks to the increased number of homogeneous macroblocks. The corresponding increased detail is more efficiently encoded using a fine partitioned macroblock type (i.e., *P_8×8*).

- Overall, it is observed that *16×8* partitions ($\mu = 1$) in the base layer rarely correspond to *8×16* partitions ($\mu = 2$) in the enhancement layers.

This is explained by the fact that in the enhancement layer either the same orientation is maintained as in the low resolution image; more details are included (and thus *P_8×8* will be more likely to be selected); or the texture is smoothed (*P_L0_16×16*). The analogy holds for *8×16* partitions in the base layer. It can be stated that in the enhancement layer *8×16* and *16×8* partitioned macroblocks can occur when μ_{BL} has such partitioning. This property will be referred to as the orthogonality property and can be described as in Equation 2.4, where MB is any random macroblock of the enhancement layer and $x \neq y$.

$$\forall MB : \mu_{BL} \in \{P\_L0\_x \times y\} \mapsto \mu_{EL} \notin \{P\_L0\_y \times x\} \tag{2.4}$$

This orthogonality property is depicted in Figure 2.7(d) - 2.7(f). Analogous to the orthogonality property, it is seen that both rectangular partitioned macroblock modes ($\mu =$ *P_L0_L0_16×8* and $\mu =$ *P_L0_L0_8×16*) are selected less frequently in the enhancement layer when the co-located base layer macroblock corresponds to $\mu =$ *P_L0_16×16* or $\mu =$ *P_8×8*. It seems that macroblocks which do not have a rectangle oriented partitioning in the base layer, do not tend to have such partitioning in the enhancement layer either.

2.3.5 Conclusions for the enhancement layer macroblock type analysis

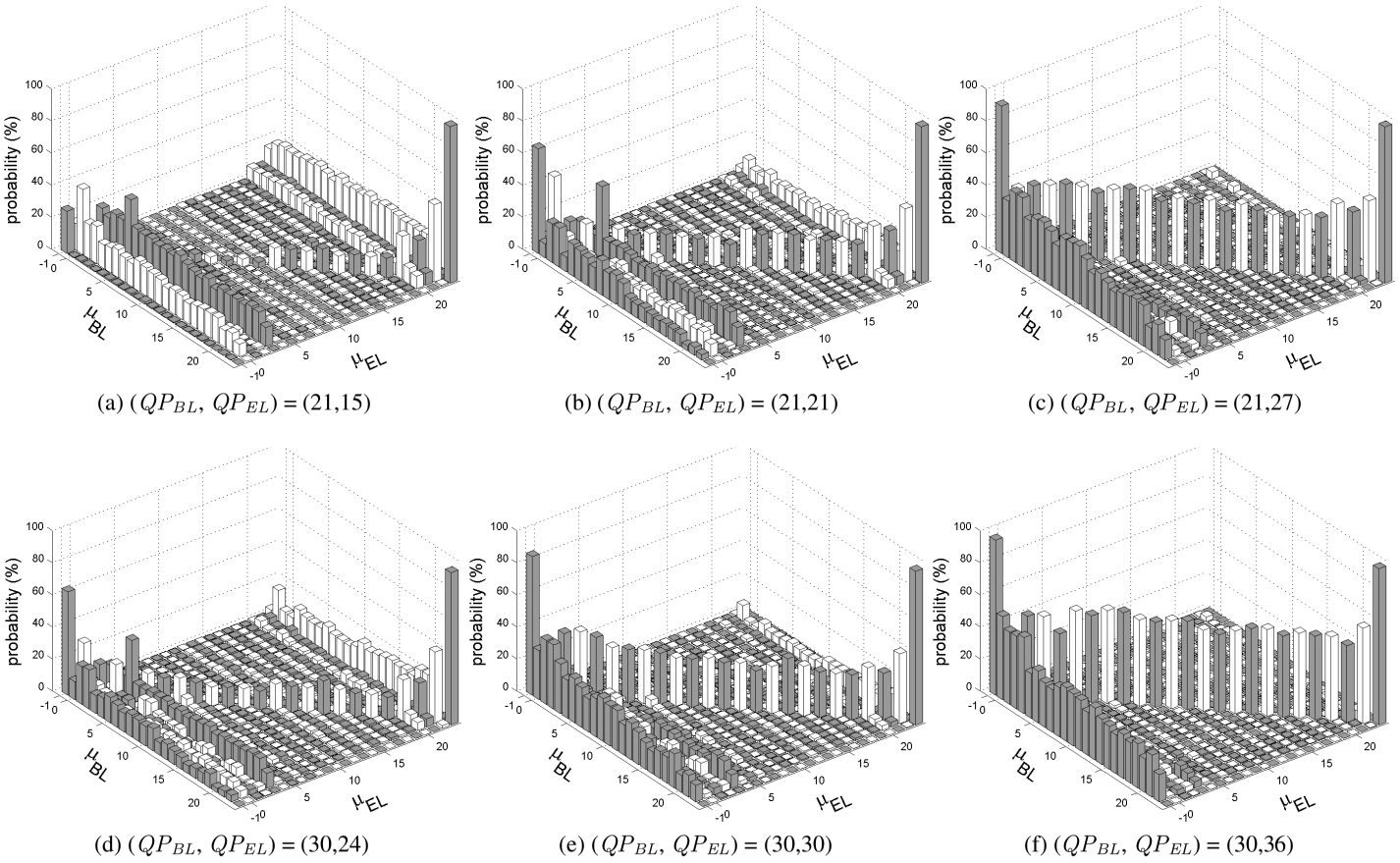
The previous analysis shows a correlation between the macroblock type of the co-located macroblocks in the base and enhancement layer. This correlation depends on the applied quantisation of the layers and the visual characteristics of the sequence. For all sequences, it is seen that the base layer macroblock type has a high probability to be selected in the enhancement layer for the co-located macroblock (diagonal property). For B pictures, $B\_Direct\_16\times16$ and *B_Bi_16×16* occur frequently , while for P pictures *P_L0_16×16* and *P_8×8* ($\mu = 3$) have a higher occurrence. Furthermore, partitioned macroblocks tend to have the same partitioning in the enhancement layer and increasing the quantization results in a higher probability for non-coded macroblock types.
The presented analysis can be used to reduce the SVC encoding process. The complexity of the enhancement layer mode decision process can be reduced if the probability for selecting each macroblock type is known given the selected base layer macroblock type. Based on the previous analysis, a model is derived in the next section. This model reduces the macroblock modes that have to be evaluated in the enhancement layer. Using the previous analysis, the proposed model can be evaluated theoretically, as is done in Section 2.4.4. An implementation of the model shows the complexity reduction and RD loss due to the inaccuracy of the model, as presented in Section 2.4.5.

2.4 Proposed fast mode decision model

Based on the previous analysis a mode decision model is designed using only the prior knowledge of QP_{EL} and μ_{BL}. This model reduces the enhancement layer encoding complexity of SVC by reducing the set of evaluated modes of the enhancement layer. The model is evaluated for both complexity and RD performance and is compared against the performance of Li's model.
Since different macroblock types are used for B and P pictures, a different model is designed for each of the types.

2.4.1 Proposed model for B pictures

In order to introduce the basic building blocks of the designed model, observations from the analysis are highlighted again (partly shown in Figure 2.8). These graphs show the average conditional probability for a CIF resolution



(a) (QP_{BL}, QP_{EL}) = (21,15) (b) (QP_{BL}, QP_{EL}) = (21,21) (c) (QP_{BL}, QP_{EL}) = (21,27)

(d) (QP_{BL}, QP_{EL}) = (30,24) (e) (QP_{BL}, QP_{EL}) = (30,30) (f) (QP_{BL}, QP_{EL}) = (30,36)

Figure 2.8: Ordinal graphs reflecting the average conditional probability $P(\mu_{EL}|\mu_{BL})$ for inter-coded macroblock types and *I_4×4* in B pictures with CIF/4CIF resolution.

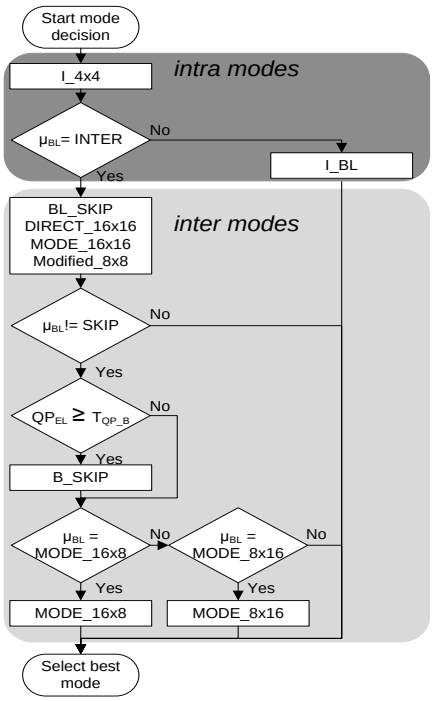


Figure 2.9: Flowchart of the proposed model for B pictures.

at the base layer and 4CIF resolution for the enhancement layer. Figure 2.9 shows the flowchart of the proposed model, which takes prior knowledge of QP_{EL} and μ_{BL} into account. This flowchart reduces the set of macroblock modes that have to be evaluated in the mode decision process of the encoder compared to the reference encoder which evaluates all modes.

For both intra- and inter-coded base layer macroblocks $I\_4\times4$ is evaluated. For inter-coded base layer macroblocks, the characteristics of $I\_4\times4$ make this a valuable type to maintain a high rate-distortion efficiency when the most optimal mode is not evaluated. For intra-coded base layer macroblocks, I_BL is evaluated in the enhancement layer. On the one hand because this mode comes with a low complexity, on the other hand because it has a high impact on the rate-distortion efficiency. Because the base layer signal is used as a predictor for the enhancement layer, only a low amount of residual data has to be encoded. None of the inter-coded macroblock modes are evaluated for intra-coded base layer macroblocks since the corresponding probability is almost zero, which is expressed by Equation 2.5. This can be seen in Figure 2.6 where the bottom left part is almost zero.

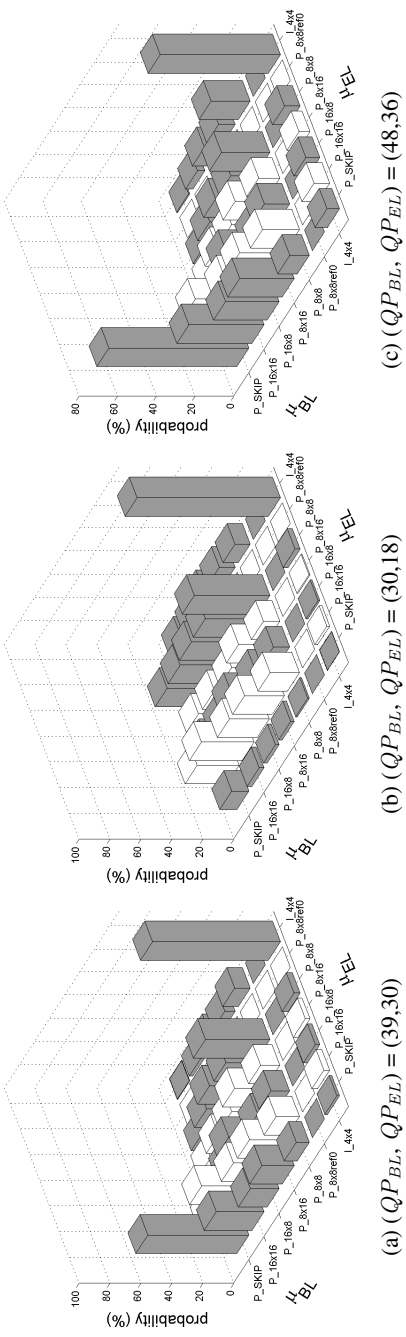
Sub-macroblock partition size	Relative complexity (%)
Direct_*8×8*	0.05
8×8	20.02
8×4	22.66
4×8	25.92
4×4	31.35

Table 2.4: Overview of the complexity of the *8×8* sub-macroblock partitions relative to the total encoding time of *B_8×8*.

$$\begin{aligned} &\forall(\mu_{BL}, \mu_{EL}) \in S : P\{\mu_{EL}|\mu_{BL}\} = 0 \\ &where\ S = \{(\mu_{BL}, \mu_{EL}) \mid 23 \leq \mu_{BL} \leq 48 \wedge -1 \leq \mu_{EL} \leq 22\} \end{aligned} \tag{2.5}$$

The analysis in Section 2.3.2 shows the *diagonal property*, which is a diagonal of significant values indicating that the base layer mode is selected during the enhancement layer mode evaluation process. Therefore, the base layer mode with ILP should be evaluated (BL_SKIP). Furthermore, three modes are always selected independent on the base layer macroblock mode ($MODE$): $B\_Direct\_16\times16$, *MODE_16×16*, *MODE_8×8*. The first two are unpartitioned and therefore require only a limited complexity. The last one has a high probability but to reduce the computational complexity, only *8×8* partitions are evaluated. Table 2.1 shows that *MODE_8×8* accounts for almost 60% of the total complexity on average. As can be seen in Table 2.4, most of this complexity is allocated to evaluate the partitions smaller than *8×8* (79.93%), while between 60-70% of the *MODE_8×8* are *8×8* -partitioned. Therefore, eliminating the sub-*8×8* partition sizes results in a limited reduction of compression efficiency but a siginificant reduction in encoding complexity.

When the base layer macroblock is a non-coded macroblock (B_SKIP), an early termination is applied. From Figure 2.8 can be seen that the modes that are likely to be selected have already been evaluated for the enhancement layer. B_SKIP has already been evaluated using BL_SKIP. If the base layer macroblock is not B_SKIP, then a regular B_SKIP (without ILP) will be evaluated in the enhancement layer based on the quantization of the enhancement layer. Comparing Figure 2.8(b) and Figure 2.8(c) with Figure 2.8(a) shows that B_SKIP has an insignificant probability for inter-coded base layer macroblocks when the QP_{EL} is below a threshold (T_{QP_B}). For B pictures, this threshold has been experimentally verified to be $T_{QP_B} = 18$.



(a) $(QP_{BL}, QP_{EL}) = (39,30)$ (b) $(QP_{BL}, QP_{EL}) = (30,18)$ (c) $(QP_{BL}, QP_{EL}) = (48,36)$

Figure 2.10: Ordinal graphs reflecting the average conditional probability $P(\mu_{EL}|\mu_{BL})$ for inter-coded macroblock types and *I_4×4* in P pictures with CIF/4CIF resolution.

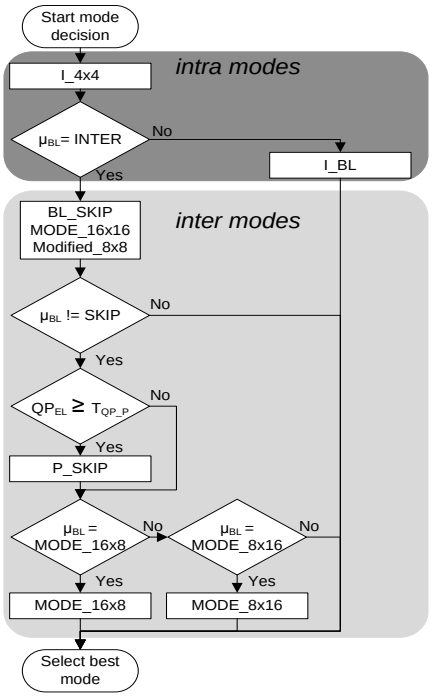


Figure 2.11: Flowchart of the proposed model for P pictures.

In the analysis, it is shown that the probabilities for MODE_*16*×*8* and MODE_*8*×*16* ($\mu = 4 \ldots 21$) are insignificant when the base layer macroblock is not coded using the same type. Therefore, these types are only evaluated when the base layer is encoded using this type.

2.4.2 Proposed model for P pictures

The conditional probability $P(\mu_{EL}|\mu_{BL})$ for P frame macroblocks is given in Figure 2.10, both ordinal axes indicate the macroblock type. Based on this analysis, the flowchart of the proposed model for P pictures is presented in Figure 2.11. The proposed model for P pictures is similar to the one for B pictures, although some differences occur. The enhancement layer macroblock types are evaluated according to the following rules.

- *I_4×4* and I_BL are evaluated because of the same reasons as described for B pictures.
- P pictures do not support MODE_DIRECT. Therefore, initially only BL_SKIP, *MODE_16×16*, and a modified *MODE_8×8* are evaluated.
- In analogy to the B pictures flowchart, for P_SKIP macroblocks in the base layer, only the aforementioned three modes have to be evaluated in the enhancement layer.
- The evaluation of P_SKIP in the enhancement layer is only required when $QP_{EL} \geq \ T_{QP_P}$. From the analysis it is seen that $T_{QP_P} = 24$ is a good value as QP threshold for P pictures. Figure 2.10 shows the high probability of P_SKIP when $QP_{EL} \geq \ T_{QP_P}$.
- As with B pictures, and according to the orthogonality property, the modes *MODE_16×8* and *MODE_8×16* are only important if such partitioning is used for the base layer macroblock.

2.4.3 Comparison with Li's model

As pointed out in Section 2.2, Li's model [30] is the best performing model in available literature and is shown in Figure 2.12. Compared to the model proposed in Section 2.3, Li's model evaluates for non-partitioned base layer macroblocks only the same macroblock mode and inter-layer residual prediction. For the non-partitioned (inter-coded) macroblock modes, the base layer macroblock mode which resulted in the second best RD-cost is additionally evaluated for the enhancement layer ($\text{MODE}_{ELpred2}$ in Figure 2.12). However, the analysis in Section 2.3 shows a high correlation between the base and enhancement layer modes, without the need for evaluating the second best base layer mode. When the model would be incorporated in a system where also the base layer is optimized, not all modes will be evaluated for this base layer. Therefore, $\text{MODE}_{ELpred2}$ might be of lesser significance for the enhancement layer encoding. Consequently, the usability of this model is strictly limited to systems that only optimize the enhancement layer.

2.4.4 Accuracy

The accuracy of the model indicates the amount of macroblocks for which the model evaluates the correct macroblock type. The accuracy of the proposed model can be calculated based on the probabilities derived from the

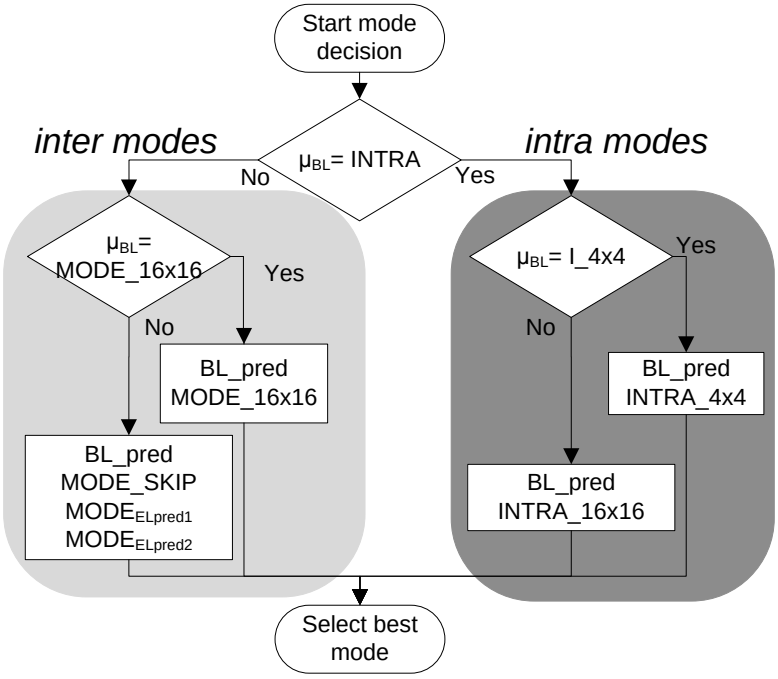


Figure 2.12: Flowchart of Li's proposed model, according to [30].

analysis. The weighted average of the probabilities $P(\mu_{EL} \mid \mu_{BL})$ results in the total amount of macroblocks that are evaluated correctly in the enhancement layer. Since the model is also applied to referenced frames, these references are different compared to the anchor. Therefore, the most optimal mode in the referencing frame might have been changed compared to the anchor. So, selecting the RD-wise optimal mode in a frame based on a non-optimal reference frame might result in a different mode compared to the anchor. Consequently, this macroblock is considered to be inaccurate compared to the model, while it should have been considered to be accurate.

Nevertheless, the accuracy of the model can give an indication about how well the model fits the (anchor's) reality and can indicate shortcomings in the model. Moreover, the accuracy also shows if the model is content independent. When the same accuracy is achieved for each sequence, then the model can be considered content independent. The total accuracy of the model is given in Table 2.5. This accuracy takes into account all analyzed sequences as elaborated on in Section 2.3.1. The weighted average (taking into account the higher resolution for sequences *Crew* and *City*) for both B and P pictures shows an accuracy of 86.5% for the proposed model.

Sequence	Accuracy of B pictures (%)	Accuracy of P pictures (%)
Bus	82.87	87.11
City	87.87	84.19
Crew	85.30	88.88
Foreman	86.66	89.40
Mobile	88.96	82.57
Weighted Average	86.47%	86.49

Table 2.5: Accuracy of the proposed model for the analyzed sequences compared to the anchor.

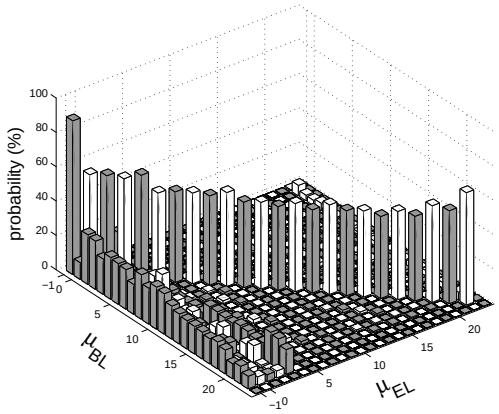


Figure 2.13: Probability $P(\mu_{EL} \mid \mu_{BL})$ for all evaluated sequences with $QP_{BL} = 18$ and $QP_{EL} = 24$.

The weighted average of the probabilities $P(\mu_{EL} \mid \mu_{BL})$ for $QP_{BL} = 18$ and $QP_{EL} = 24$ of all evaluated sequences of the next section (*Ice*, *Harbour*, *Rushhour*, *Soccer*, *Station*and *Tractor*) is shown in Figure 2.13. It can be clearly seen that for $\mu_{BL} \in \{-1, 0, 1, 2, 3\}$ no *MODE_16×8* or *MODE_8×16* is evaluated. Furthermore, for *MODE_16×8* only the orthogonal mode *MODE_16×8* is evaluated, while for *MODE_8×16* only *MODE_8×16* is evaluated. Note that the list prediction is not optimized such that macroblock types with a different list prediction as the base layer still might be selected in the enhancement layer. This can be seen by the noise close to the diagonal. Figure 2.14 illustrates the difference between the modeled probability and the real probability for one sequence (*Harbour* with $QP_{BL} = 24$ and $QP_{EL} = 12$). The modeled probability is obtained by

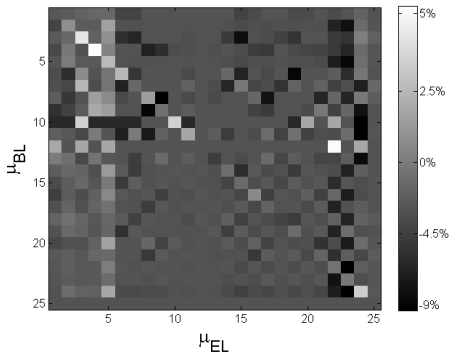
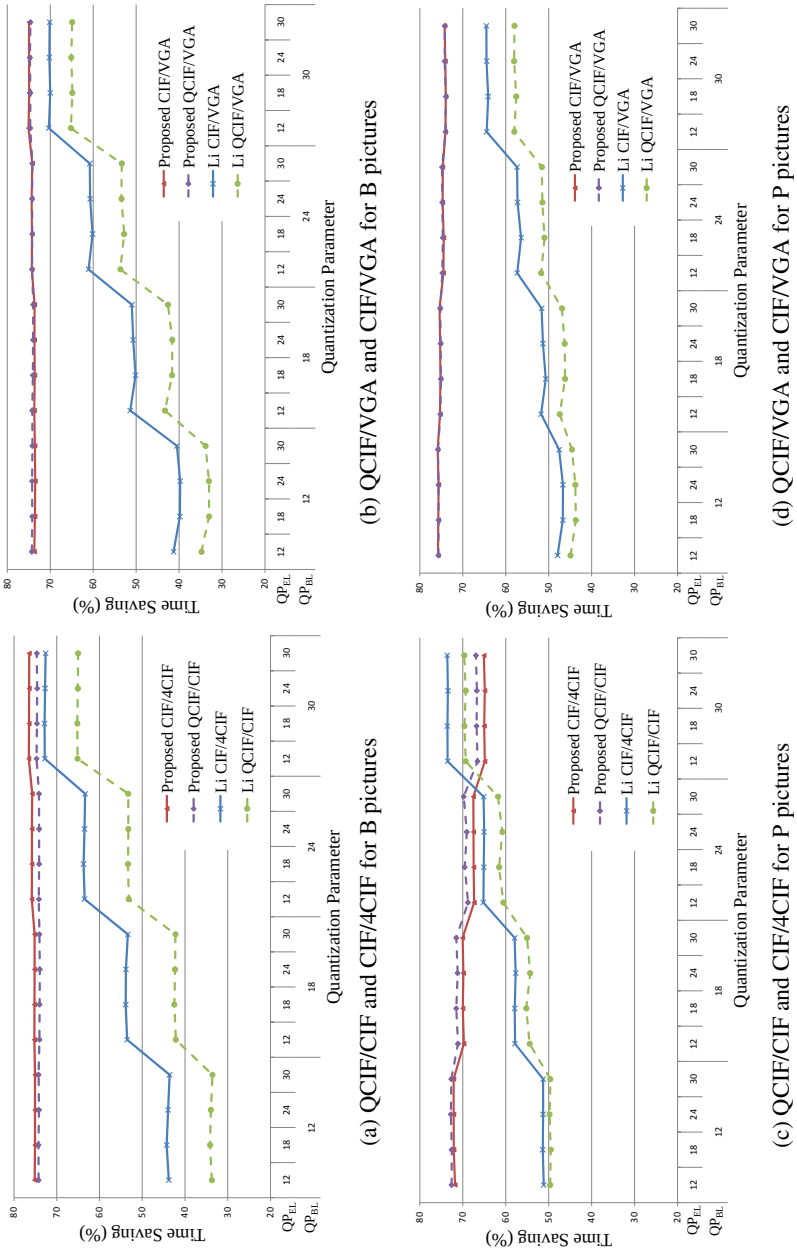


Figure 2.14: Mismatch between modeled and real probability for *Harbour* with $QP_{BL} = 24$ and $QP_{EL} = 12$.

analyzing the probabilities after encoding the sequence with the proposed optimizations adopted in the encoder. The real probabilities are obtained by encoding the sequence without any optimizations, as has been done for the analyzed sequences. Negative values indicate where the model did not select μ_{EL} compared to the non-optimized scenario. The shown probabilities are not weighted over the total number of macroblocks.

2.4.5 Experimental results

As pointed out in Section 2.2, Li's model is one of the best performing models. Since other fast mode decision models [26, 32, 33] compare with Li, the proposed model can be compared with these models too. Therefore, we will compare our results with [30]. To do so, the proposed model and Li's model have been implemented with the JSVM 9.4 reference software [35]. Only the mode decision step was modified, so both algorithms can be objectively compared. An unmodified version of the reference software is used to generate the reference streams in terms of quality, compression efficiency, and encoding complexity. To achieve the highest possible RD the motion estimation performs an exhaustive block search, which allows to accurately evaluate the influence of the RD by the proposed model.
All streams have been generated on the same machine, a dual quad core processor operating Windows XP. Experiments are done using six test sequences, which were not used for analysis and have different visual characteristics: *Ice*, *Harbour*, *Rushhour*, *Soccer*, *Station* and *Tractor*. These sequences are encoded with $QP_{BL}, QP_{EL} \in \{12, 18, 24, 30\}$. Only a part of the QP range $(0 \ldots 51)$ is used because for broadcasting the bit rates are too high when $QP < 12$, and the quality for $QP > 30$ is too low.



(a) QCIF/CIF and CIF/4CIF for B pictures

(b) QCIF/VGA and CIF/VGA for B pictures

(c) QCIF/CIF and CIF/4CIF for P pictures

(d) QCIF/VGA and CIF/VGA for P pictures

Figure 2.15: Average time saving for the proposed and Li's model for all sequences.

To examine the effect of the resolution scalability on the fast mode decision models, various resolution combinations (notated as: Res_{BL}/Res_{EL}) are applied to the (QP_{BL}, QP_{EL}) combinations. Firstly QCIF/CIF and CIF/4CIF resolutions are applied to observe the dyadic spatial scalability, corresponding to the resolutions used during the analysis. Secondly, the proposed model is verified for other resolution combinations; QCIF/VGA and CIF/VGA resolutions are simulated to observe the effect of non-dyadic scaling. For each encoded bitstream, 64 frames are encoded, with a GOP size and intra-period of 16 and 32 frames respectively.

The computational complexity is assessed first. This is done by evaluating the time saving (Time Saving (TS)) of the fast mode decision models relative to the original encoding process. This time saving is given by Equation 2.6.

$$TS\ (\%) = \frac{T_{Original}\ (ms) - T_{Fast}\ (ms)}{T_{Original}\ (ms)}. \tag{2.6}$$

The time saving is based on the number of calculations performed by the CPU. This might not give a direct indication of the energy reduction in any given system. However, in order to efficiently evaluate different algorithms, the CPU time can give an indication of the required number of calculations. In general, it can be assumed that the higher the CPU time is, the more energy is required for encoding. This energy can either be calculations in Digital Signal Processors, the number of VHDL gates or the speed that is required for CPU based systems. Since the same codebase is used for the all evaluated techniques, the difference in time saving gives an indication of the relative complexity reduction between the algorithms.

$T_{Original}$ represents the CPU time required to encode the original enhancement layer, while T_{fast} is the CPU time required to encode the enhancement layer with a fast mode decision model applied. The same machine was used for evaluating the complexity of all sequences. Therefore, the timing overhead for each sequence is comparable. Measurements showed a 0.2% deviation in execution time for the same content and parameters. Since there is not a single way to measure the reduction in complexity, the processing time is used. A reduction in processing time most likely yields a reduction of the required energy consumption.

Second, rate-distortion graphs are plotted in order to compare the results of the proposed fast mode decision model against those obtained with Li's model and to assess the influence of the proposed model on the overall RD performance of the system.

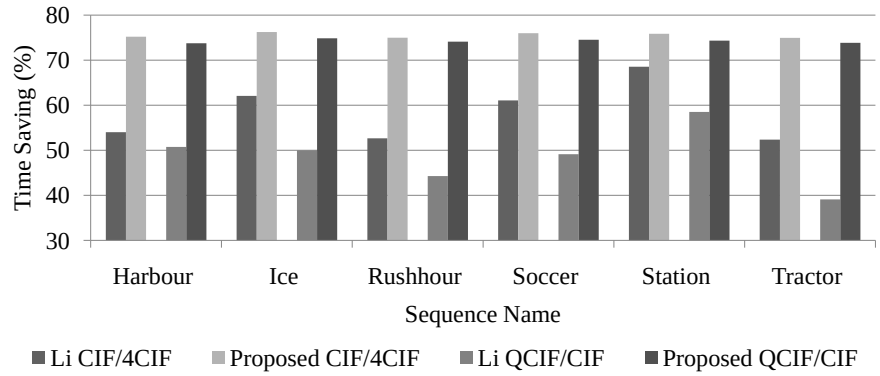
MODE	QP_{BL}= 12	QP_{BL}= 18
P_SKIP	6.96	35.57
P_L0_16×16	20.1	24.65
P_L0_16×8	7.85	10.41
P_L0_8×16	6.61	13.03
P_8×8	58.39	16.34
I_4×4	0.03	0
I_16×16	0.05	0

Table 2.6: Comparison of the relative distribution of P macroblock modes for sequence *Foreman* with QP_{EL}= 18 to illustrate the increase in complexity due to the distribution of base layer macroblock modes.

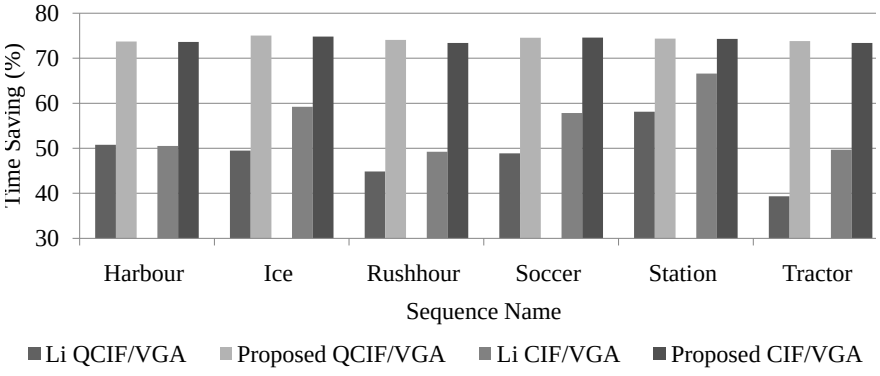
2.4.6 Encoding complexity

Figure 2.15 shows the average time savings obtained by Li's model and the proposed model, where all sequences are averaged over each (QP_{BL}, QP_{EL}) combination. It can be seen that the proposed model obtains an average complexity reduction of around 75%, which is rather constant over all quantization combinations since mostly the same number of evaluations has to be performed. For P pictures with QCIF/CIF resolution (Figure 2.15(c)), a small variation in time saving of about 6% is noticed, from 72.8% to 66.6%. Due to the smaller number of P pictures, this variation in complexity reduction will have only a minor impact. Furthermore, the complexity reduction for the proposed model does not depend on the resolution of both layers. Both resolution configurations show the same complexity reduction, while Li has a significant gap between both resolutions.

The complexity reduction for Li's model increases with a higher QP_{BL}, as was already reported but not explained by [30]. This is because the coarser quantization leads to more homogeneous macroblocks in the base layer for which a *16×16*-partitioning is selected. So, for more macroblocks only BL_SKIP and *MODE_16×16* have to be evaluated in the enhancement layer On the other hand, for sequences with a low quantized base layer around 75% of the macroblocks have either *MODE_8×8* selected or *MODE_8×8* corresponds to the second best RD optimal mode. Since Li's model takes also the second best RD optimal mode of the base layer into account for the enhancement layer mode decision, this high occurence of *MODE_8×8* results in a high computational complexity for the enhancement layer. Table 2.6 supports this, where the distribution can be seen for the base layer modes of sequence *Foreman* with two different QPs.



(a) QCIF/CIF and CIF/4CIF



(b) QCIF/VGA and CIF/VGA

Figure 2.16: Average time savings for Li's model and the proposed model taking into account all QP combinations for each sequence.

The average complexity reduction for each test sequence is presented in Figure 2.16. Again, it can be seen that the complexity reduction for the proposed model is rather constant around 75%, which indicates the independency of the model for the content of the input video. On the other hand, for the same resolution combination, Li's model shows variations of the complexity reduction up to 19%, depending on the sequence. (e.g. for QCIF/VGA, a complexity reduction of 58.12% is reported for sequence *Station* while sequence *Tractor* shows only a reduction of 39.32% in complexity). These findings are in line with those presented in [27], where variations of 12% are reported. Furthermore, variations in complexity reduction between different resolution combinations for Li's model are also observed in Table 2.7, which presents the average complexity reductions

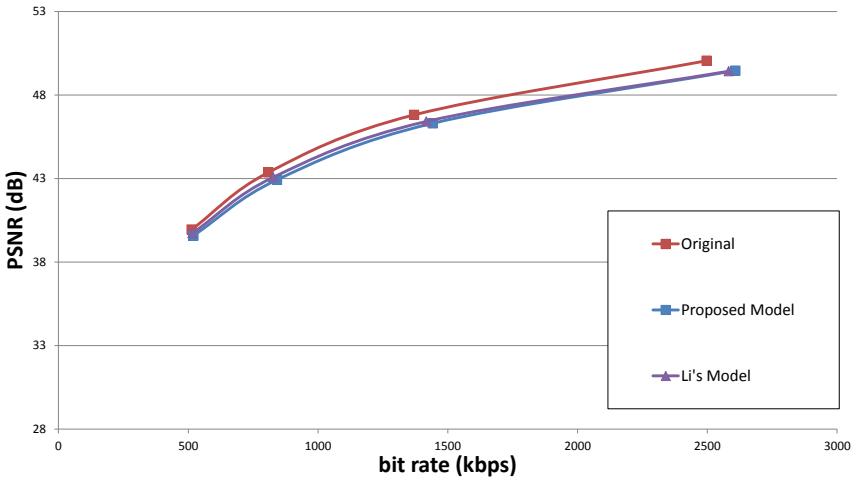
	Average time saving (%)	
Resolution	Li	Proposed model
QCIF/CIF	48.63	74.24
CIF/4CIF	58.49	75.54
QCIF/VGA	48.57	74.52
CIF/VGA	55.50	74.01
Average	52.80	74.58

Table 2.7: Overview of the average time saving for each resolution comparing Li's model and the proposed model.

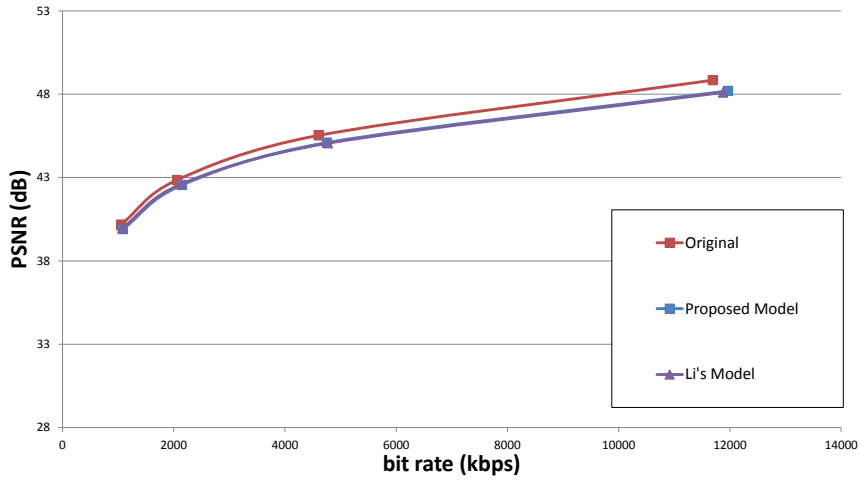
for each resolution.
The huge variation in complexity reduction for Li is due to the fact that the number of enhancement layer macroblock evaluations for many macroblocks depend on both the predicted macroblock type and the second best evaluated macroblock type. For the proposed model, only macroblock types *MODE_16×8* and *MODE_8×16* might slightly influence the complexity of the mode decision process of the enhancement layer.
Compared to Li, the model proposed by [33] shows an average complexity reduction of less than 4.5%. Furthermore, the presented complexity reduction for Li are comparable to the obtained results for the proposed model, therefore it is safe to assume that the proposed model has a lower complexity compared to [33]. In [32], complexity is further reduced with 10% for spatial scalability if their proposed model is combined with Li's model. Note that this complexity reduction heavily depends on content and quantization. The complexity reduction reported in [26] is 7% lower than that of Li, while for the combination of their model with Li's model, the complexity reduction is improved with 11% compared to Li's model. Since the proposed method reduces complexity with 22% compared to that of Li, and the complexity is independent of content, resolution, and quantization, it can be safely stated that the proposed model not only outperforms Li's model in terms of complexity, but also other existing fast mode decision models [26, 32, 33].

2.4.7 Rate distortion analysis

A rate-distortion analysis is provided to evaluate the impact on the bit rate and the corresponding image quality. RD results are shown in Figure 2.17 and Figure 2.18. The input of the base layer is obtained by downsam-



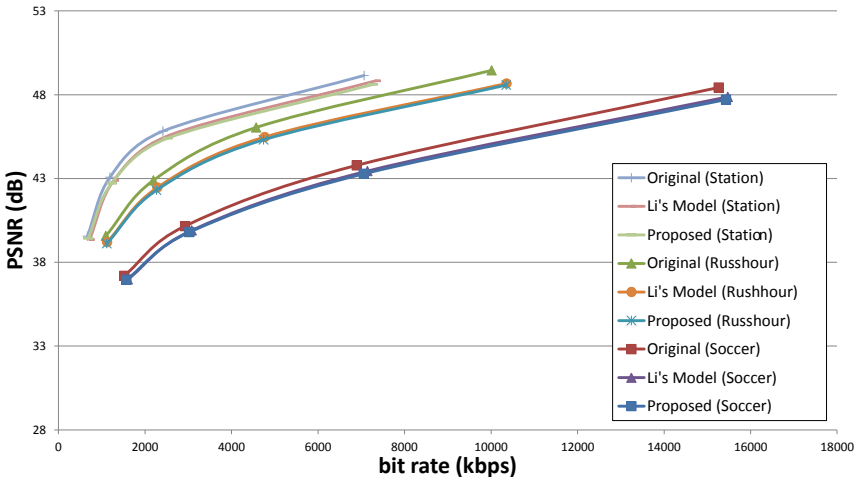
(a) RD curve for *Ice* QP_{BL}= 24 (QCIF/CIF resolution)



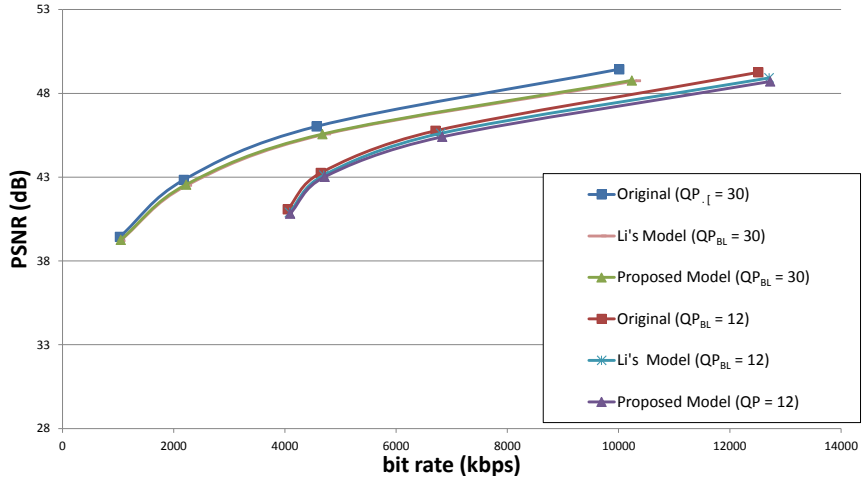
(b) RD curve for *Ice* QP_{BL}= 30 (CIF/4CIF resolution)

Figure 2.17: Rate distortion curves for QCIF/CIF and CIF/4CIF resolution highlighting the general trends ($QP_{EL} \in \{12, 18, 24, 30\}$).

pling the higher resolution input sequences (as depicted in Figure 2.2). This downsampling is non-normative and thus any downsampling filter can be used. For I_BL and residual prediction a normative SVC upsampling filter is used. Therefore, to eliminate any possible mismatches between both filters, which could introduce visual artifacts, the downsampling filter equivalent to the normative upsampling filter is used for the input sequence. The



(a) RD curve for QP_{BL}= 24 (QCIF/VGA resolution)



(b) RD curve for *Rushhour* QP_{BL}= 12 and QP_{BL}= 30 (CIF/VGA resolution)

Figure 2.18: Rate distortion curves for QCIF/VGA and CIF/VGA resolution highlighting the general trends ($QP_{EL} \in \{12, 18, 24, 30\}$).

Peak Signal-to-Noise Ratio (PSNR) of the base layer resolution is obtained by comparing this downscaled input sequence with the decoded base layer. In general, all RD curves for the generated sequences are similar to those in Figure 2.17 and Figure 2.18. The mutual relationships between the curves are preserved. A gap is noticed between the unmodified version and the fast mode versions, which corresponds to the loss in quality and increase

in bandwidth. Only a slight difference can be noticed between both fast mode decision model curves. A minor gap is noticed between the rate-distortion results obtained with the proposed fast mode decision model and the model proposed by Li. The higher complexity of Li's model will typically result in slightly improved rate-distortion, although Figure 2.17(b) and Figure 2.18(b) show that this is not always the case. Here, the proposed model shows gains in coding efficiency, along with the benefit of reduced computational complexity. For lower bit rates (higher quantization of the enhancement layer) the performance of Li's model degrades, since the curves for both models overlap or show only a very small gap.

Comparing Figure 2.17(a) with Figure 2.17(b) and Figure 2.18(a) with Figure 2.18(b) shows that for both dyadic and non-dyadic spatial scalability, the resolutions of both base and enhancement layer do not have an influence on the rate distortion performance.

From the experiments, using 16 QP combinations with 4 resolution combinations, the average bit rate shows an increase of 2.28% for Li's model and an increase of 2.23% for the proposed model compared to the reference encoder. The PSNR shows a reduction for Li's model with 0.34 dB, while the proposed model has a degradation of 0.46 dB compared to the output of the reference encoder. While generally maintaining the bit rate, the quality is slightly reduced. However, this quality degradation is not visible, because of the high PSNR values for the original encoded bitstreams.

A detailed overview of some of the results for all sequences with QPs (12,24) or (24,18) and a CIF/4CIF or CIF/VGA resolution is given in Table 2.8 and Table 2.9, respectively. In these tables, the complexity reduction or time saving (TS), delta bit rate (ΔBR) and delta PSNR (ΔPSNR) for both the model of Li and the proposed model are given, based on the bitstreams encoded by the unmodified reference encoder. Note that the positive ΔBR represents an increase in bit rate for the signal, while a negative ΔPSNR represents a decrease in quality.

The average effect on quality and bit rate is given in Table 2.10, which shows the average Bjøntegaard Delta bit rate (BDRate) and Bjøntegaard Delta PSNR (BDPSNR) [36, 37]. BDRate and BDPSNR are obtained by integrating the difference of two RD curves. The curves are based on four points and a third order polynomial is applied to create the curves. By integrating the difference between both curves over the X-axis (rate) or Y-axis (PSNR) the average difference in rate or PSNR is known between two curves. The columns *Li's model* and *Proposed model* show the BDRate and BDPSNR for the proposed and Li's model compared to an original encoder. The last columns show the BDRate and BDPSNR difference for the proposed architecture compared to Li's architecture.

		Li's model [30]			Proposed model		
		TS (%)	ΔBR (%)	ΔPSNR (dB)	TS (%)	ΔBR (%)	ΔPSNR (dB)
Harbour	(12,24)	47.17	0.52	-0.25	74.85	0.63	-0.38
	(24,18)	55.10	1.13	-0.40	75.19	1.32	-0.63
Ice	(12,24)	42.35	0.32	-0.14	75.65	0.70	-0.21
	(24,18)	69.71	2.58	-0.31	76.46	2.67	-0.38
Rushhour	(12,24)	35.69	0.16	-0.12	74.60	0.29	-0.10
	(24,18)	58.17	2.05	-0.38	74.90	1.68	-0.44
Soccer	(12,24)	49.13	0.47	-0.19	75.26	0.20	-0.23
	(24,18)	65.52	1.27	-0.38	76.39	1.28	-0.52
Station	(12,24)	48.43	-0.03	-0.16	74.99	0.71	-0.17
	(24,18)	78.21	3.90	-0.20	76.19	3.99	-0.33
Tractor	(12,24)	40.94	0.39	-0.21	74.72	0.40	-0.26
	(24,18)	55.68	2.55	-0.40	74.86	2.70	-0.57
Average	(12,24)	43.95	0.31	-0.18	75.01	0.49	-0.23
	(24,18)	63.73	2.25	-0.35	75.67	2.27	-0.48

Table 2.8: Time saving (TS), delta bit rate (ΔBR) and delta PSNR (ΔPSNR) for all sequences for (12,24) and (24,18) with CIF/4CIF resolution compared to the original encoding process.

The BDRate and BDPSNR values are in line with the previous findings. The additional gain in complexity compared to Li's model results in a slightly lower quality (-0.10dB) and a small increase in bit rate (2.1%). The BDRate and BDPSNR are shown compared to the original encoder, additionally the BDRate and BDPSNR are given for the proposed method compared to Li's model.
The presented results are generated with different content and quantization than used in [32] and [33]. This makes comparisons difficult, although major trends are observed. Compared to [32], the proposed model shows in a slightly reduced RD performance. Note that this is compensated by the significantly lower reported complexity. Compared to [33], the proposed model show similar RD results for a reduced complexity.

2.4.8 Conclusions on the SVC encoding process optimizations

A profound analysis of six sequences with varying content and quantization revealed the correlation between the mode selection of the base and enhancement layer of SVC bitstreams. Firstly, the same macroblock type between base and enhancement layer is selected regularly. Secondly, the influence of the non-coded macroblocks increases with a higher enhancement layer quantization. Finally, the non-partitioned macroblock types have a high probability to be selected and a partitioned base layer macroblock, is likely to have the same partitioning in the enhancement layer.
Based on this analysis, a model is derived. The proposed model results always in a low complexity, while maintaining a high coding efficiency. The experimental results show that the proposed model for both dyadic and non-dyadic spatial scalability performs similar to state-of-the-art mode decision techniques, in terms of rate and distortion. The proposed solution even yields a lower average bit rate increase (+2.23%) compared to Li's model (+2.28%), while quality slightly degrades (-0.46 dB vs. -0.34 dB). However, this degradation is hardly visible because of the high PSNR values of the original bitstreams. The reduction in PSNR is obtained because the model does not always predict the most optimal mode. The proposed model has an average accuracy of 86.5%. Consequently, 13.5% of the macroblocks have a less optimal prediction, resulting in a higher bit rate for the macroblock, but also a higher distortion of the reconstructed macroblock.
The encoding complexity is reduced by 75% compared to an original encoder, where Li's model obtains only a 52% complexity reduction. Consequently, this technique only needs half of the complexity compared to the state-of-the-art technique, for comparable coding efficiency. Moreover, it was seen that the proposed technique shows a nearly constant complexity

		Li's model [30]			Proposed model		
		TS (%)	ΔBR (%)	ΔPSNR (dB)	TS (%)	ΔBR (%)	ΔPSNR (dB)
Harbour	(12,24)	41.97	0.75	-0.18	73.06	0.73	-0.27
	(24,18)	51.47	1.20	-0.41	73.65	1.25	-0.60
Ice	(12,24)	39.13	1.07	-0.16	74.08	1.81	-0.25
	(24,18)	66.40	3.55	-0.32	74.79	3.8	-0.41
Rushhour	(12,24)	30.95	1.16	-0.14	73.18	1.51	-0.23
	(24,18)	53.16	2.36	-0.38	73.29	2.09	-0.45
Soccer	(12,24)	44.77	0.85	-0.17	73.65	0.79	-0.21
	(24,18)	60.89	2.42	-0.36	74.81	2.07	-0.49
Station	(12,24)	44.94	0.32	-0.22	73.60	0.92	-0.18
	(24,18)	76.74	4.30	-0.24	74.74	3.63	-0.30
Tractor	(12,24)	36.61	0.79	-0.15	73.20	1.22	-0.34
	(24,18)	52.25	2.94	-0.41	73.43	3.09	-0.55
Average	(12,24)	39.73	0.82	-0.17	73.46	1.16	-0.25
	(24,18)	60.15	2.80	-0.35	74.12	2.66	-0.47

Table 2.9: Time saving (TS), delta bit rate (ΔBR) and delta PSNR (ΔPSNR) for all sequences for (12,24) and (24,18) with CIF/VGA resolution compared to the original encoding process.

	Li's model [30]		Proposed model		Proposed vs Li's model	
	BDRate	BDPSNR	BDRate	BDPSNR	BDRate	BDPSNR
Harbour	6.00	-0.42	8.73	-0.59	2.55	-0.18
Ice	9.51	-0.36	11.40	-0.43	1.76	-0.07
Rushhour	8.82	-0.35	10.05	-0.41	1.15	-0.06
Soccer	7.05	-0.37	9.25	-0.47	2.03	-0.10
Station	7.20	-0.26	10.12	-0.36	2.78	-0.10
Tractor	8.02	-0.43	10.56	-0.57	2.36	-0.13

Table 2.10: RD performance for all sequences with CIF/4CIF resolution.

reduction, while for existing techniques the complexity reduction highly depends on resolution, content properties and quantization of the bitstream. This implies that the hardware complexity required to implement the model can be estimated more accurately for the proposed model.
The proposed model can be implemented in both software and hardware designs, but requires adaptation of the complete mode decision process. This might create some difficulties for hardware designs. Typically, hardware designs re-use existing building blocks, and most of those already have optimized algorithms to reduce the cost[8] of the system. Consequently, it might not be practical or feasible to completely implement the proposed model. To reduce complexity of existing systems, generic techniques are proposed in the next section. These techniques evaluate smaller optimizations of the mode decision process. These smaller optimizations can be mutually combined and incorporated in existing fast mode decision models.

2.4.9 Future Work

The SVC enhancement layer encoding has been optimized by means of an analytical model, which has been derived by analyzing multiple video streams. After evaluating the model, it has been verified by machine learning techniques. Since the outcome of the machine learning was identical to the proposed model, the machine learning has not been elaborated on in this chapter. However, machine learning can be an efficient tool in the future concerning SVC enhancement layer encoding. Using machine learning techniques, a large amount of data can be trained continuously. This means that the model can be trained while the encoder is operational. In a sense this allows to train the model on a specific set of video, rather than a broad

[8]This cost can be the energy consumption, computational complexity, chip surface, design complexity, ... or a combination thereof.

range. However, if the training is repeated regularly, this will yield better results. In a future implementation this can be considered. Nevertheless, it will be important to select a good set of features. During the evaluation using machine learning techniques. A broad set of features (including motion vector information, (maximum) energy of the residual signal, variance of the energy of the residual signal, and variance of the mean of the energy of each 4×4 block of the residual signal) have been evaluated, although the best performing tree was only using base layer macroblock information and identical to the analytical model.

Furthermore, as pointed out in Section 2.1, the base layer can also be optimized. In this research, this has not been evaluated to identify the RD-cost and complexity gain due to the base layer. It has to be analyzed how the enhancement layer will behave when an optimized base layer is used for encoding. In case a non-optimal block is selected for the base layer, the model might miss the (new) most optimal enhancement layer block. On the other hand, having a less optimal base layer macroblock type, might result in a different enhancement layer macroblock type. However, both might yield a global optimal RD-cost rather than an optimal RD-cost for both layers seperately.

Lastly, the mode decision invokes the motion estimation process. Since an exhaustive block search is used to accurately evaluated the RD performance, the complexity can be further reduced. As pointed out in Section 2.1, fast motion estimation techniques can be applied to the motion estimation process. This will further reduce the total encoding complexity.

2.5 Generic techniques to reduce the enhancement layer encoding complexity

When the complete mode decision process of an encoder, as presented in Section 2.4, can not be completely be adapted, smaller optimizations can be applied. Therefore, in this section techniques are proposed to reduce the encoding complexity that are complementary to other design optimizations. Based on the analysis in Section 2.3 some evident changes can be made to hardware encoder designs, in order to reduce the encoding complexity of the enhancement layer. These techniques can be generalized, influence only a small part of the design, and are likely to be compatible with existing optimizations and fast mode decision models. To identify this compatibility, the proposed generic techniques have been combined, and have been implemented on top of an existing fast mode decision model. Consequently, the required complexity of such models is further reduced.

2.5.1 Proposed techniques

The proposed modifications are generic in a sense that they can be mutually combined and used in combination with other fast mode decision models, as long as the applicable (sub-)process of the existing fast mode decision model is not already altered. This technique will work because most existing models operate mainly on a mode decision level, therefore the proposed sub-mode decision level adaptations can further improve the performance of such models. Obviously, when an existing mode decision process already alters the process of the proposed optimization, the proposed optimization cannot be applied. Nevertheless, other proposed generic techniques can still be implemented. Selective inter prediction [18] is such a technique, and will be discussed in combination with the proposed generic techniques

The proposed techniques are evaluated as a stand-alone implementation to investigate the impact on the RD performance and complexity. Furthermore, an existing mode decision model [30] is improved, while the additional complexity reduction and loss in image quality are evaluated.

Three techniques are proposed, which are derived from the previous analysis: *disallow orthogonal macroblock modes*, *only evaluate sub8x8 blocks if present in the base layer* and *only evaluate the base layer list predictions*. In the results section these techniques will be referred to as *ort*, *sub* and *list*, respectively. Where necessary, more specific analysis results are additionally presented for each proposed technique. Other techniques might also be applicable. However, given the previous analysis, these three techniques show a high potential for complexity reduction and applicability in existing fast mode decision models. Since good results are obtained for the proposed optimizations, no additional analysis has been performed.

Disallow orthogonal macroblock modes

Based on the analysis result of Section 2.3, the average conditional probabilities (p) of all quantizations and sequences have been determined. The resulting conditional probabilities are shown in Figure 2.19, where all μ have been combined in the probability for the modes. Here the orthogonal property clearly is seen for $\mu_{BL}, \mu_{EL} \in \{MODE\_8\times16, MODE\_16\times8\}$. Therefore the orthogonal mode of the base layer should not be evaluated during the enhancement layer mode decision process. This is the same observation as was already made for the proposed fast mode decision model in Section 2.4. Note that from Figure 2.19 can be seen that also $SKIP$, $MODE\_Direct$, and $I\_4\times4$ have a low selection probability when $\mu_{BL} \in \{MODE\_8\times16, MODE\_16\times8\}$. However, according to Table 2.1 these

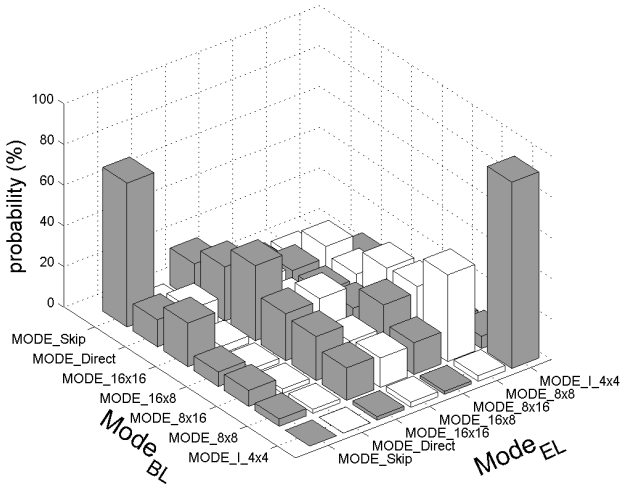


Figure 2.19: Average conditional probability for enhancement layer modes.

three modes nearly contribute to the overall complexity.

Only evaluate sub8x8 blocks if present in the base layer

Figure 2.20 shows the probability that a *sub_8×8* mode is selected as the enhancement layer macroblock mode. As can be seen, less than 40% of all *MODE_8×8* macroblocks have a *sub_8×8* partition size. Meanwhile, almost 80% of the complexity of *MODE_8×8* is required for those *sub_8×8*-partitions, as shown in Table 2.4. Therefore, *sub_8×8*-evaluation should be limited to macroblocks that were encoded as a *MODE_8×8* macroblock in base layer. When a less partitioned macroblock mode is selected for the base layer, this indicates that less details are required to be encoded in the content. Since upscaling mostly preserves this property, it is unlikely that a finer partitioned macroblock mode will be selected in the enhancement layer. Consequently, *sub_8×8*-partitions are only required to be evaluated in the enhancement layer if these are encoded in the base layer, which will result in an accuracy of 64%.

Note that this is a more flexible approach than the modified *MODE_8×8* in the previously designed model, which did not evaluate *sub_8×8* partitions at all. The proposed generic technique does evaluate *sub_8×8* -partition sizes if the base layer macroblock is *MODE_8×8*.

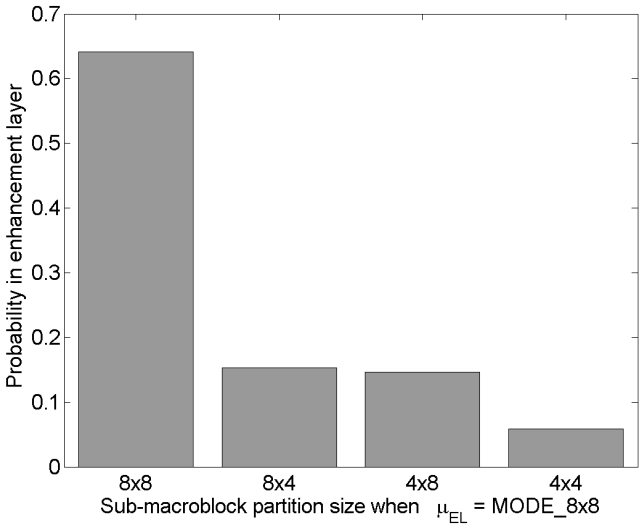


Figure 2.20: Distribution for sub-macroblock partition sizes for *MODE_8×8* in spatial enhancement layers.

Only evaluate the base layer list predictions

As introduced in Section 2.1, the prediction direction (forward, backward or bi-prediction) is defined by the prediction list (resp. L0, L1 or L0 and L1). Base and enhancement layer show a high probability for using the same prediction list if the macroblock mode is the same (Figure 2.21). For example for *MODE_16×16*, $\mu_{EL} \in \{1, 2, 3\}$, it is seen that the same type is used. Consequently, since μ_{EL} identifies the prediction list[9], the same prediction list as the base layer has a higher probability. For *MODE_16×8* and *MODE_8×16* a diagonal of higher probabilities is seen. This diagonal corresponds to $\mu_{BL} = \mu_{EL}$, so the base layer list prediction is favored.

This property can be exploited, due to the resemblance of the video content in both layers. Since the prediction direction is dependent on the video content, both layers are likely to have the same prediction list, because the content in both layers is similar. Therefore, the correspondence between the current frame and the reference frame for both layers will be similar. Consequently, the encoder will most likely use the same prediction direction for both layers.

[9] $\mu_{EL} = 1$ uses L0; $\mu_{EL} = 2$ uses L1; $\mu_{EL} = 3$ uses Bi-directional prediction.

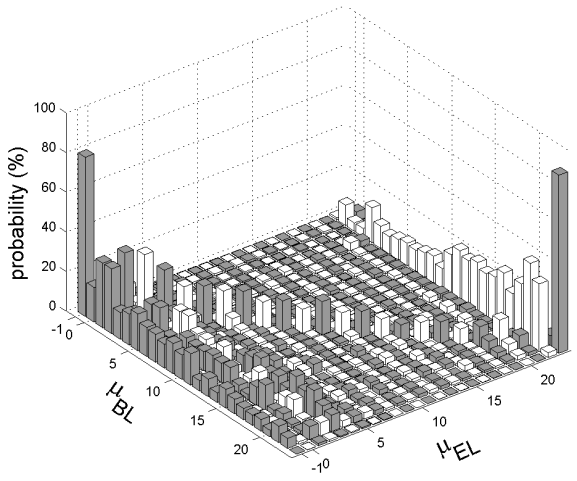


Figure 2.21: Average conditional probability (p) identifying the list prediction between both layers.

2.5.2 Results

The presented techniques yield a lower complexity for the encoder, since the mode decision process does not need to evaluate all macroblock modes, it does reduce evaluations for sub-macroblock partitions, and does not have to evaluate all prediction directions. However, each of these properties yield a low loss in quality and increase, since it might be that the most optimal enhancement layer mode is not selected. Therefore, combining these three techniques will lower the complexity, but also the RD performance. To evaluate the influences of the proposed techniques on complexity and RD performance, first they are evaluated as standalone techniques, and subsequently, the proposed techniques are combined with Li's fast mode decision model.

Four test sequences with different characteristics (*Harbour*, *Ice*, *Rushhour*, and *Soccer*), have been encoded with varying combinations of the base and enhancement layer quantizations $QP_{BL}, QP_{EL} \in \{18, 24, 30, 36\}$. The presented rate distortion performance results always have a fixed QP_{BL} and a different QP_{EL}. Dyadic spatial scalability is applied for two resolution combinations: QCIF/CIF and CIF/4CIF. For reference purposes, the test sequences have been encoded with the JSVM_9.4 reference software [35]. Furthermore, these sequences are encoded with the original encoder opti-

Method	ΔBR (%)	ΔPSNR (dB)	TS (%)
Ort	0.60	-0.05	26.95
Sub	0.20	-0.03	53.98
List	0.91	-0.06	17.76
Ort+Sub	0.53	-0.09	73.91
Ort+Sub+List	1.06	-0.13	77.15

Table 2.11: RD performance and time saving for the proposed techniques in a standalone scenario.

mized with the proposed techniques, with Li's model improved with the proposed techniques and with selective inter-layer residual prediction [18]. The complexity reduction and RD performances of the encoded sequences is evaluated. The RD performance measurements are expressed as a difference in bit rate (ΔBR) and a difference in PSNR (ΔPSNR) relative to the original encoded sequences. Comparison of the complexity is done by the time saving given by Equation 2.6. Since the same codebase is used for the original encoder, Li's optimized encoder and the proposed techniques, the difference in time saving gives an indication of the complexity reductions. The time saving is expressed as a percentage, to compare the complexity reduction of the proposed techniques independently of the hardware. Time measurements are executed on a dedicated machine with a dual quad core processor and 32 GB of RAM memory.

Results for generic techniques as a standalone solution

Table 2.11 shows the average results for the proposed techniques. When using only one improvement (single technique), the *sub_8×8* reduction method (*Sub*) results in the highest complexity reduction, virtually without degradation of the RD performance. When small complexity reductions are sufficient, this is a good candidate. Only disallowing orthogonal macroblock modes (*Ort*) or limiting the list predictions (*List*) will perform worse for both compression efficiency and complexity. Figure 2.22 shows the RD performance of the proposed single techniques for sequence *Rush-hour* with $QP_{BL} = 36$. Obviously, extending technique *Sub* with *Ort* results in an even lower complexity. This performs better than combining *Sub* with *List*, which can be derived from the single techniques because *List* yields a higher complexity for a more degraded compression performance. Combining all three techniques will have the highest time saving; however, this comes with a bit rate increase of about 1%. Improving an original

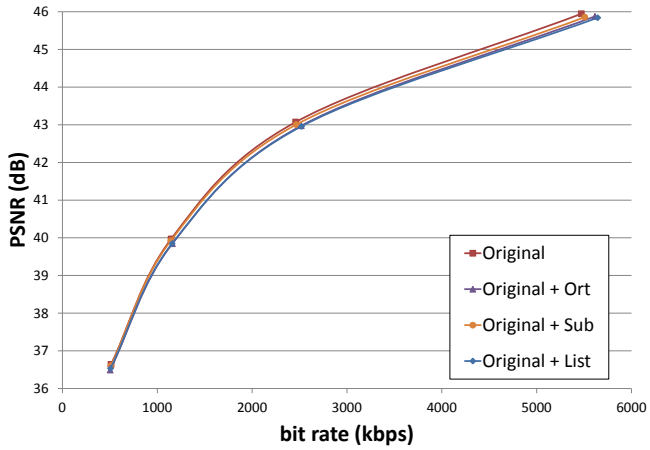
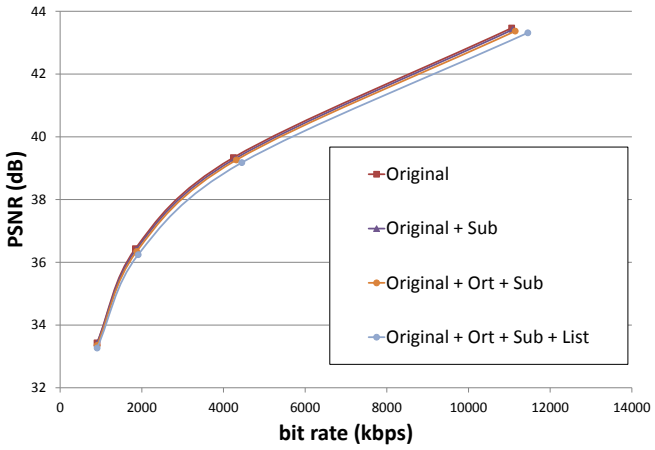


Figure 2.22: Comparison of the RD performance for the proposed generic in a standalone scenario.

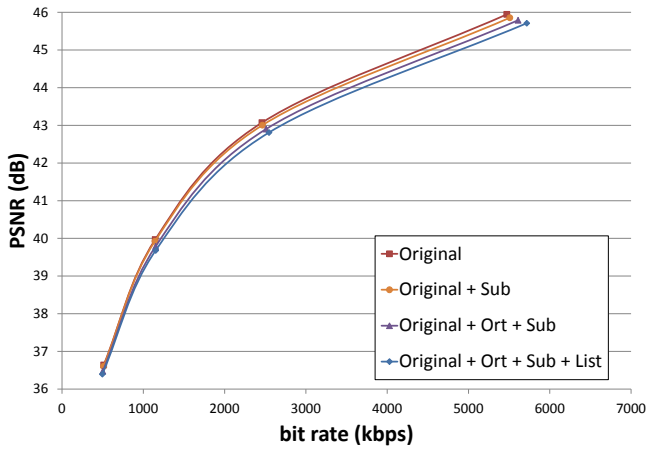
encoder with these three techniques requires only 22% of the complexity of the original encoder, while nearly an equal compression performance is achieved. Furthermore, it can be seen that combining techniques does not yield to the sum of both solutions. Combining *Ort* and *Sub* does not yield a loss of 1.51% in bit rate and 0.11dB in PSNR. Because the selected modes might slightly change in the reference pictures compared to the original encoding, the reference image will slightly change. This might result in selecting a different mode in the frame that has to be encoded. Consequently, this also influences the RD performance, such that adding the RD performance of multiple adaptations does not hold.

Since technique *Ort* and *Sub* only interfere slightly, the complexity reduction can, approximately, be summed. However, *List* has an influence on both *Ort* and *Sub*, since it modifies the list prediction of those macroblocks that still have to be evaluated. Consequently, going from *Ort+Sub* to *Ort+Sub+List* results in a complexity reduction of 17%. However, this corresponds to 4% of the complexity of the original encoder. Therefore, given the different nature of the level where the optimizations are applied, the complexity reductions of the single techniques give an indication of the complexity reduction when they are combined with other techniques.

Figure 2.5.2 shows the coding efficiency for the generic techniques used in a standalone scenario. It can be seen that only the combination of all techniques has a slightly lower RD performance. Furthermore, for high quality base layers (Figure 2.23(c)) the reduction in RD performance is

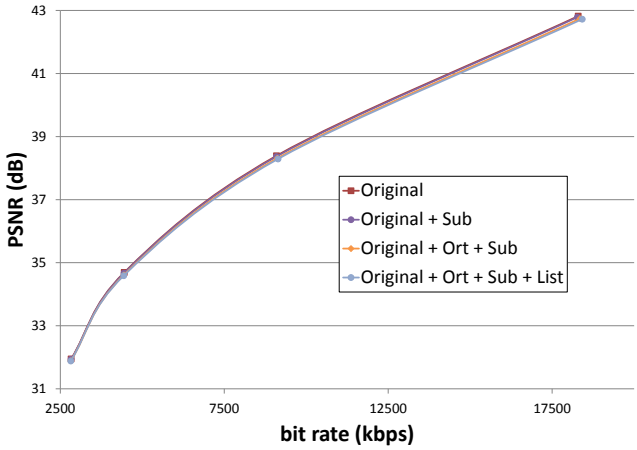


(a) *Soccer* @ QP_{BL}= 36



(b) *Rushhour* @ QP_{BL}= 36

Figure 2.23: RD performance results for combining the generic techniques without external optimizations.



(c) *Harbour* @ QP_{BL}= 24

Figure 2.23: RD performance results for combining generic techniques in a standalone configuration (cont.).

lower than low quality base layers (Figure 2.23(a) - 2.23(b)). Such a small decrease justifies the use of low complex generic techniques. When lower complexities are required, these techniques can be combined with fast mode decision models. In such cases, the RD performance will further decrease.

Generic techniques to improve existing fast mode decision models

While the proposed single techniques are useful in standalone scenarios, they can also be combined with existing fast mode decision models. Again, Li's model is used to evaluate the effects of the generic improvements for existing fast mode decision models. In [26] their reported time saving is 7% lower than Li, while for the combination of their model with Li, the time saving is improved with 11% compared to Li.
Results for combining Li with the proposed techniques can be found in Table 2.12. When using multiple generic techniques, only the results for Li+*Ort*+*Sub* are shown, since Li+*Sub*+*List* and Li+*Ort*+*List* yield a higher complexity and lower coding efficiencies, for the same reason as with the standalone techniques. Note that while adding one single technique only seems to yield small time savings, the absolute gains are comparable to those shown in Table 2.11. As can be seen, Li+*Ort*+*Sub* has only 2.6% less complexity gain compared to Li+*Ort*+*Sub*+*List*, although the absolute complexity of the latter is 17% lower compared to the former.

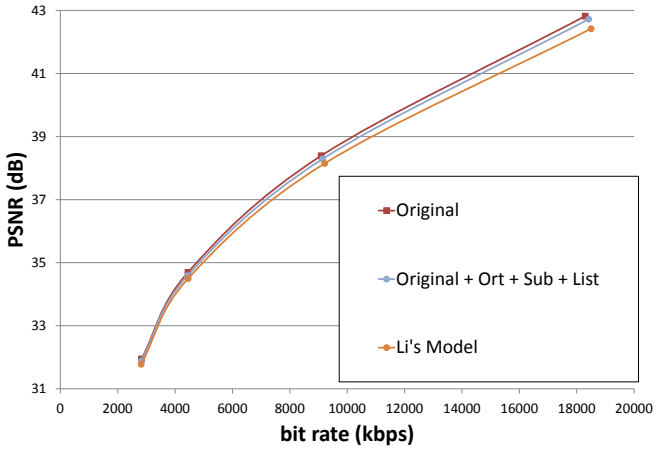
Method	ΔBR (%)	ΔPSNR (dB)	TS (%)
Li	1.40	-0.25	66.76
Li+*Ort*	1.55	-0.27	68.37
Li+*Sub*	1.39	-0.28	82.02
Li+*List*	2.13	-0.30	71.39
Li+*Ort*+*Sub*	1.50	-0.31	84.47
Li+*Ort*+*Sub*+*List*	2.14	-0.36	87.27

Table 2.12: RD performance and time saving for standalone scenario of the proposed techniques.

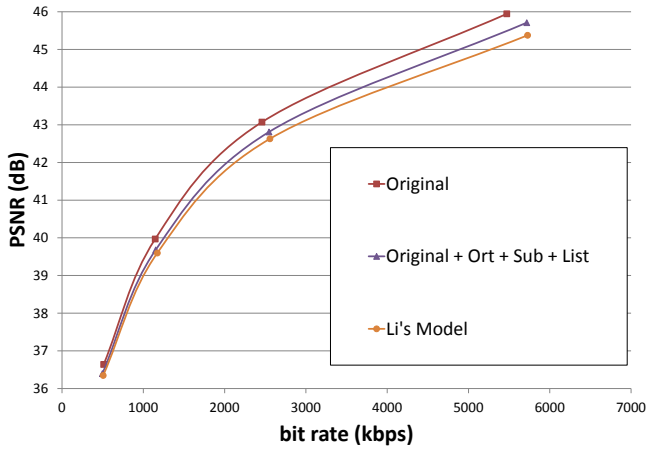
Comparing Table 2.11 with the results for the unmodified Li's model, shows that generic techniques yield better RD performance for a lower complexity. The generic techniques require 31.25% less complexity compared to Li, even though a better RD performance is measured. Figure 2.24 shows the lower RD performance of Li's model compared with the lower complexity technique of combining the proposed generic techniques. From this observation, it can be concluded that single generic techniques are preferred for small complexity reductions ($< 80\%$), while the criterion for combinations with fast mode decisions should lie with very low complexity solutions.
In Figure 2.25(a) and Figure 2.25(b) the RD performance of the combinations with Li's model are shown for sequence *Harbour* using a CIF/4CIF resolution, and for sequence *Rushhour* with a QCIF/CIF resolution. Improving Li's model with all proposed generic techniques only slightly degrades the RD performance compared to Li's model, an increase of 0.74% in bit rate and merely 0.11 dB lower PSNR are measured. Meanwhile, the former only requires half of the complexity compared to Li's model. Combining all generic techniques with Li's model degrades the picture quality with 0.36 dB, and requires only an increase of 2.14% in bandwidth. On the other hand, only 12.73% of the original complexity is needed. These results satisfy the requirements for using fast mode decision models in real-world systems. However, if only small bit rate increases are allowed, one of the other proposed techniques can be chosen, while the highest possible RD efficiency is guaranteed.

Improving with selective inter-layer prediction

As previously mentioned, selective inter-layer residual prediction [18] can be used to extend existing fast mode decision models, since it does not change the mode decision process. This makes selective inter-layer resid-



(a) *Harbour* @ QP_{BL}= 24

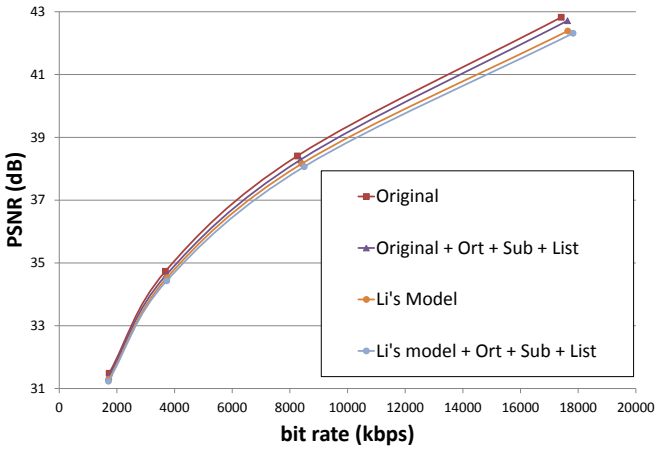


(b) *Rushhour* @ QP_{BL}= 36

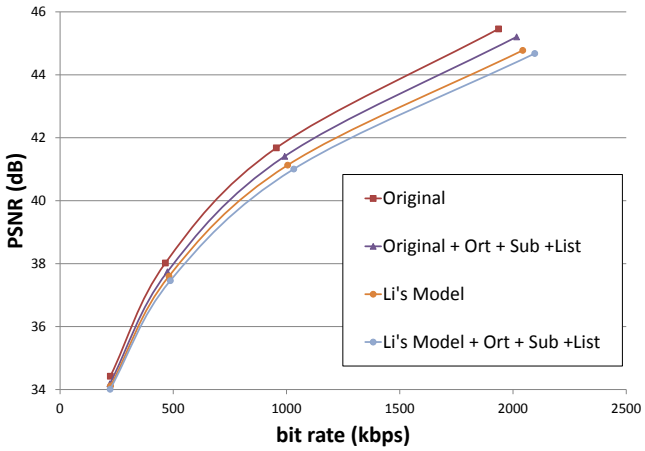
Figure 2.24: Comparison of the RD performance for the combined proposed generic techniques and Li's proposed model without modifications.

ual prediction also a generic technique. To stress the universality of generic techniques, Li's model is further improved with selective inter-layer residual prediction.

Figure 2.26 represents the RD performance for applying selective inter-layer residual prediction for Li's model both with and without the proposed



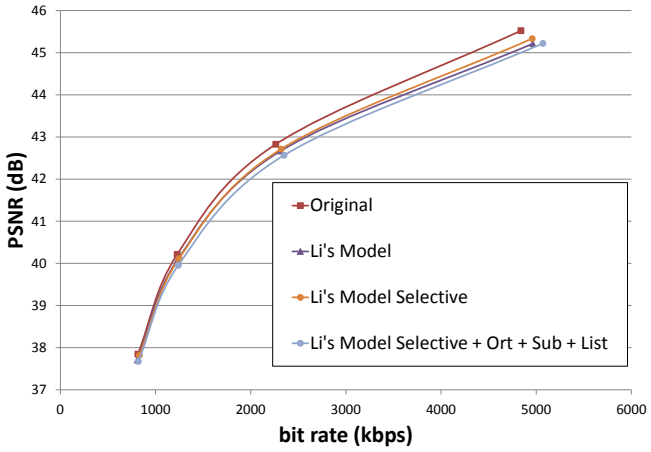
(a) *Harbour* @ QP_{BL}= 30



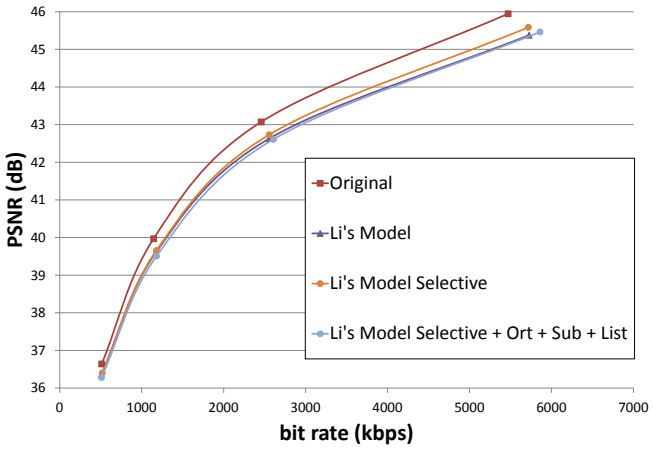
(b) *Rushhour* @ QP_{BL}= 36

Figure 2.25: Comparison of the RD performance for the combined proposed generic techniques and Li's proposed model improved with the combined generic techniques.

generic techniques. As can be seen from this figure and Table 2.13, using selective inter-layer residual prediction with only Li's model results in the same complexity reduction as Li+*List* (Table 2.12), while a slightly better RD performance is achieved. Combining selective inter-layer resid-



(a) *Ice* @ QP_{BL}= 24



(b) *Rushhour* @ QP_{BL}= 36

Figure 2.26: Comparison of the RD performance when selective inter-layer residual prediction is applied to existing mode decision models (sequence *Ice*).

ual prediction with all generic techniques further reduces the complexity (compared to Li+*Ort*+*Sub*+*List*, the complexity reduces with 11.5%), on the other hand the RD performance further degrades.

Method	ΔBR (%)	ΔPSNR (dB)	TS (%)
Li + Selective	3.03	-0.14	72.24
Li+ *Ort*+*Sub*+*List* + Selective	3.77	-0.26	88.74

Table 2.13: RD performance and time saving of selective inter-layer residual prediction in combination with Li's model.

2.5.3 Conclusions for the use of the generic techniques

Based on an analysis of encoded scalable bitstreams, generic techniques have been identified. These techniques are closely related to the proposed fast mode decision model (a modified *sub_8×8* evaluation is used compared to no *sub_8×8* and the diagonal property is also adopted from the fast mode decision model in Section 2.4). However, the combination of the generic techniques outperforms the proposed fast mode decision model in terms of complexity.

The proposed generic techniques are usable in a standalone scenario where complexity reductions are required, while a high coding efficiency is important. It is shown that these techniques yield a high compression efficiency, independently of the content, resolution or quantization. When combining these generic techniques with existing fast mode decision models, a system that requires only 12.7% of the complexity compared to a normal SVC encoder can be built. Furthermore, it is shown that these techniques can be used with existing optimizations, such as selective inter-layer residual prediction. The latter requires an even lower complexity of 11.3%.

The degradation of the RD performance for lower complexities has to be taken into account when defining the complexity of the total system. Since the techniques can be applied on a per macroblock basis, the complexity of the encoder can be scaled according to the actual constraints of the system. The presented results are compared against a state-of-the art model, which is referred to by multiple publications. Therefore, the results for the generic techniques can be compared for the complexity and RD performance with other models. Compared to Li's model, the presented method requires 31.25% less complexity by combining the generic techniques. Combining Li's model with the generic techniques results in 61.70% lower complexity, while a reduction of 66.13% in complexity is achieved if also the selective inter-layer residual prediction is used. In [26] 27.63% complexity reduction compared to Li's model is reported (for dyadic spatial scalability). Therefore, the proposed generic techniques require less complexity than [26].

Finally, the presented techniques are compatible with future improved fast mode decision models. This opens the path for the introduction of SVC encoders to allow efficient transport systems to deliver one single bitstream, carrying multimedia content for different types of end user terminals over heterogeneous networks.

2.5.4 Future work on generic techniques

For systems with a known (fixed) complexity reduction, like a fixed number of encoded streams, one of the above techniques can be implemented, such that the highest RD performance is guaranteed. When a system with varying complexity is designed, all of the above techniques can be implemented. However, only those techniques that lead to the available complexity should be used. Further reducing the complexity than required should not be done. This will guarantee the highest possible RD performance. Using the required techniques can be done on a per macroblock basis, based on the current actual load. This makes the encoder a complexity scalable encoder. Moreover, complexity scalability schemes can be investigated, not only based on the current load, but also taking into account power consumption, heat dissipation, ... ultimately leading to a green encoder.

2.6 Conclusions on the complexity reduction for scalable video coding

SVC allows to deliver a single bitstream to many devices with different characteristics. Different layers are introduced to scale the bitstream according to the characteristics of the network or the device. Transmitting an SVC bitstream results in a reduced bit rate compared to a simulcast scenario where for each device a different stream is transmitted. In order not to end up with a simulcast scenario, redundant information between layers is exploited. However, encoding such bitstreams requires a significant encoding complexity. To reduce this complexity, a fast mode decision model is proposed based on an analysis. Furthermore, generic techniques are proposed which can be combined with state-of-the-art encoding algorithms.
The computational complexity of the SVC encoder is reduced by limiting the number of macroblock types that have to be evaluated. Since this evaluation represents a significant part of the total computational complexity, the overall computational complexity of the encoder is reduced drastically. In general, a complexity reduction of 75% is achieved, while bit rate increases with 2.23% and quality degrades with 0.46%. This RD performance is com-

parable with existing models. However, the complexity reduction exceeds state-of-the art models.
Additionally, generic complexity reductions have been proposed which can be used in combination with existing fast mode decision models. Three techniques are both evaluated as standalone techniques and the combinations of the techniques are evaluated. By combining all techniques, the highest complexity reduction is achieved (77%) while a performance degradation in RD is noted with 1.06% increase in bit rate and 0.13 dB decrease of PSNR.
However, the true strength of the generic techniques is the combinatorial power, such that existing models and techniques can be extended with the proposed generic techniques. This allows the encoder to further reduce the complexity. In combination with Li's model, 2.14% increase in bit rate and 0.36 dB loss in PSNR are noted, while the complexity is reduced with 87%.
The proposed model and techniques can be implemented both in hardware and software and allows for a lower energy consumption or a faster execution. Therefore, SVC bitstreams can be encoded and transmitted at a lower computational cost. This reduces the threshold to migrate towards an SVC based system.

The research described in this chapter resulted in the following publications.

- Sebastiaan Van Leuven, Koen De Wolf, Peter Lambert, and Rik Van de Walle, *"Probability analysis for macroblock types in spatial enhancement layers for SVC"*, in Proceedings of the IASTED International Conference on Signal and Image Processing 2009, pp. 221-227, Aug. 2009, USA.

- Sebastiaan Van Leuven, Glenn Van Wallendael, Jan De Cock, Rosario Garrido-Cantos, José Luis Martínez, Pedro Cuenca, and Rik Van de Walle, *"Generic techniques to improve SVC enhancement layer encoding : digest of technical papers"*, in Proceedings of the 2011 IEEE International Conference on Consumer Electronics (ICCE), pp. 135-236, Jan. 2011, USA.

- Sebastiaan Van Leuven, Jan De Cock, Rosario Garrido-Cantos, José Luis Martínez, and Rik Van de Walle, *"Generic Techniques to Reduce SVC Enhancement Layer Encoding Complexity"*, in IEEE Transactions on Consumer Electronics, Vol. 57, nr. 2, pp 827-832, May 2011.

- Sebastiaan Van Leuven, Glenn Van Wallendael, Jan De Cock, Koen De Wolf, Peter Lambert, and Rik Van de Walle, *"An enhanced fast mode decision model for spatial enhancement layers in scalable video coding"*, in Multimedia Tools and Applications, Vol. 58, nr. 1, pp 215-237, May 2012.

References

[1] T. Wiegand, G.J. Sullivan, G. Bjøntegaard, and A. Luthra. Overview of the H.264/AVC video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7):560–576, July 2003.

[2] H. Schwarz, D. Marpe, and T. Wiegand. Overview of the Scalable Video Coding Extension of the H.264/AVC standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(9):1103–1120, Sept. 2007.

[3] H. Schwarz, T. Hinz, D. Marpe, and T. Wiegand. Constrained inter-layer prediction for single-loop decoding in spatial scalability. In *Proceedings of IEEE International Conference on Image Processing (ICIP)*, pages 870–873, Sept. 2005.

[4] C.A Segall and G. Sullivan. Spatial scalability within the H.264/AVC scalable video coding extension. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(9):1121–1135, Sept. 2007.

[5] Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG. Advanced Video Coding for Generic Audiovisual Services, ITU-T Rec. H.264 and ISO/IEC 14496-10 Advanced Video Coding, Edition 5.0 (incl. SVC extension). Technical report, MPEG and ITU-T, Mar. 2010.

[6] K. De Wolf, D. De Schrijver, S. De Zutter, and R. Van de Walle. Scalable video coding: Analysis and coding performance of inter-layer prediction. In *Proceedings of 9th International Symposium on Signal Processing and its Applications (ISSPA)*, pages 895–898, Feb. 2007.

[7] S. Liu, M. Hayes, and N. Faust. Improved pel-recursive motion estimation algorithms. In *Proceedings of IEEE SoutheastCon*, pages 940–944, Apr. 1990.

[8] J. Huang and R.M. Mersereau. Multi-frame pel-recursive motion estimation for video image interpolation. In *Proceedings of IEEE International Conference on Image Processing (ICIP)*, pages 267–271, Nov. 1994.

[9] Robert M. Armitano, Ronald W. Schafer, Frederick L. Kitson, and Bhaskaran Vasudev. Motion vector estimation using spatiotemporal prediction and its application to video coding. *Digital Video Compression: Algorithms and Technologies*, pages 290–301, Jan. 1996.

[10] E. Akyol, D. Mukherjee, and Y. Liu. Complexity control for real-time video coding. In *Proceedings of IEEE International Conference on Image Processing (ICIP)*, pages 77–80, Oct. 2007.

[11] H.F. Ates and Y. Altunbasak. Rate-Distortion and Complexity Optimized Motion Estimation for H.264 Video Coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(2):159 –171, Feb. 2008.

[12] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan. Rate-constrained coder control and comparison of video coding standards. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7):688–703, July 2003.

[13] C.-H. Miao and C.-P. Fan. Efficient mode selection with extreme value detection based pre-processing algorithm for H.264/AVC fast intra mode decision. In *Proceedings of TENCON 2011 - IEEE Region 10 Conference*, pages 316–320, Nov. 2011.

[14] C.-H. Miao and C.-P. Fan. Efficient mode selection with BMA based pre-processing algorithms for H.264/AVC fast intra mode decision. In *Advances in Multimedia Modeling*, volume 6523 of *Lecture Notes in Computer Science*, pages 10–20. Springer, Jan. 2011.

[15] C.-H. Miao and C.-P. Fan. Fast inter mode decision algorithm based on macroblock and motion feature analysis for H.264/AVC video coding. In *Proceedings of 17th European Signal Processing Conference (EUSIPCO 2009)*, pages 1804–1808, Aug. 2009.

[16] Yu Hu, Qing Li, Siwei Ma, and C.-C.J. Kuo. Fast H.264/AVC Inter-Mode Decision with RDC Optimization. In *Proceedings of International Conference on Intelligent Information Hiding and Multimedia Signal Processing.*, pages 511 –516, Dec. 2006.

[17] L.-J. Pan and Y.-S. Ho. Fast mode decision algorithm for H.264 inter-prediction. *Electronics Letters*, 43(24):1351–1353, Nov. 2007.

[18] C.-S. Park, S.-J. Baek, M.-S. Yoon, H.-K. Kim, and S.-J. Ko. Selective Inter-layer Residual Prediction for SVC-based Video Streaming. *IEEE Transactions on Consumer Electronics*, 55(1):235–239, Feb. 2009.

[19] F. Pan, X. Lin, S. Rahardja, K.P. Lim, Z.G. Li, D. Wu, and S. Wu. Fast mode decision algorithm for intraprediction in H.264/AVC video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 15(7):813–822, July 2005.

[20] T. Tsukuba, I. Nagayoshi, T. Hanamura, and H. Tominaga. H.264 fast intra-prediction mode decision based on frequency characteristic. In *Proceedings of European Signal and Image Processing Conference*, Sept. 2005.

[21] H.C. Lin, W.H. Peng, and H.M. Hang. Doc. jvt-w029: Low-complexity macroblock mode decision algorithm for combined CGS and temporal scalability. Technical report, MPEG and ITU-T, San Jose, USA, Apr. 2007.

[22] H. Li, Z. Li, C. Wen, and S. Xie. Fast mode decision for coarse granular scalability via switched candidate mode set. In *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, pages 1323–1326, July 2007.

[23] C.-S. Park, B.-K. Dan, C. Haechul, and S.-J. Ko. A Statistical Approach for Fast Mode Decision in Scalable Video Coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 19(12):1915–1920, Dec. 2009.

[24] G. Goh, J. Kang, M. Cho, and K. Chung. Fast mode decision for scalable video coding based on neighboring macroblock analysis. In *Proceedings of ACM symposium on Applied Computing*, pages 1845–1846. ACM, Mar. 2009.

[25] S.-T. Kim, K. R. Konda, and C.-S. Cho. Fast Mode Decision Algorithm for Spatial and SNR Scalable Video Coding. In *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 872–875, May 2009.

[26] S.-T. Kim, K. R. Konda, C.-S. Park, C.-S. Cho, and S.-J. Ko. Fast mode decision algorithm for inter-layer coding in scalable video coding. *IEEE Transactions on Consumer Electronics*, 55(3):1572–1580, Aug. 2009.

[27] S.-W. Jung, S.-J. Baek, C.-S. Park, and S.-J. Ko. Fast mode decision using all-zero block detection for fidelity and spatial scalable video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(2):201–206, Feb. 2010.

[28] J. Ren and N.D. Kehtarnavaz. Fast adaptive early termination for mode selection in H.264 scalable video coding. In *Proceedings of IEEE International Conference on Image Processing (ICIP)*, pages 2464–2467, Oct. 2008.

[29] J.Ren and N.D. Kehtarnavaz. Fast adaptive early termination for mode selection in H.264 scalable video coding. *Journal of Real-Time Image Processing*, 4(1):13–21, Mar. 2009.

[30] H. Li, Z. Li, C. Wen, and L.-P. Chau. Fast mode decision for spatial scalable video coding. In *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS)*, page 4, May 2006.

[31] H. Li, Z. G. Li, and C. Wen. Fast Mode Decision Algorithm for Inter Frame Coding in Fully Scalable Video Coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(7):889–895, July 2006.

[32] S.-W. Jung, S.-J. Baek, C.-S. Park, and S.-J. Ko. Fast mode decision using all-zero block detection for fidelity and spatial scalable video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(2):201 –206, Feb. 2010.

[33] C.-H. Yeh, K.-J. Fan, M.-J Chen, and G.-L. Li. Fast mode decision algorithm for scalable video coding using bayesian theorem detection and markov process. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(4):563 –574, Apr. 2010.

[34] K. De Wolf, D. De Schrijver, W. De Neve, S. De Zutter, P. Lambert, and R. Van de Walle. Analysis of prediction mode decision in spatial enhancement layers in H.264/AVC SVC. In *Computer Analysis of Images and Patterns*, volume 4673 of *Lecture Notes in Computer Science*, pages 848–855. Springer, Aug. 2007.

[35] Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG. Doc. JVT-W203: Joint Scalable Video Model 10 . Technical report, MPEG and ITU-T, Apr. 2007.

[36] G. Bjøntegaard. Doc. VCEG-M33: Calculation of average PSNR differences between RD-curves. Technical report, MPEG and ITU-T, Austin, USA, Apr. 2001.

[37] G. Bjøntegaard. Doc. VCEG-AI11: Improvements of the BD-PSNR model. Technical report, MPEG and ITU-T, Berlin, Germany, July 2008.

3

Low-complexity hybrid architectures for H.264/AVC-to-SVC transcoding

3.1 Rationale

In Chapter 2, a model and generic techniques are proposed to reduce the SVC encoding complexity. However, currently the majority of the content is still encoded using H.264/AVC. Therefore, the bitstream might have to be adapted because of **network limitations** or **device limitations**. In this chapter, an efficient adaptation (transcoding) of the bitstream is proposed. Instead of transcoding the bitstream multiple times, one transcoding step is performed. The proposed architecture transcodes the H.264/AVC input bitstream to a multi-layer SVC bitstream. Therefore, only one transcoding step is required, while the resulting bitstream can be easily adapted when required.

Variable bandwidth, heterogeneous networks, and networks with multiple types of end user devices can benefit from this scalability, as is shown in Figure 3.1. After conversion, video streams can be scaled instantly using low complexity techniques. This results in a reduced energy consumption in the network.

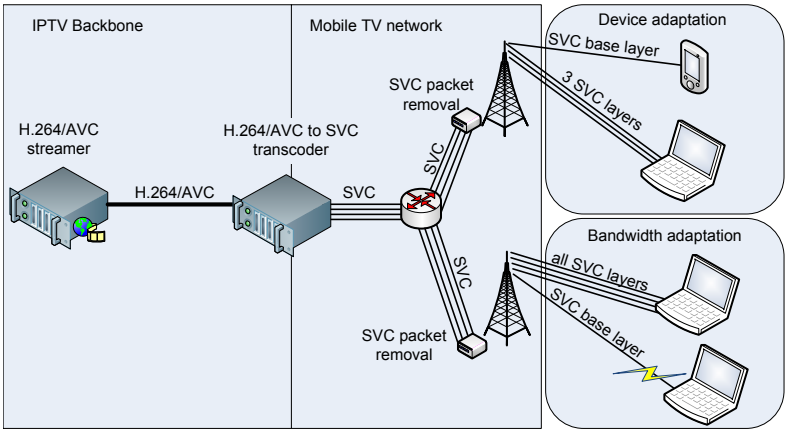


Figure 3.1: Scalable video network examples with varying bandwidth and multiple end user devices.

3.1.1 Network limitations

Scalable video coding (SVC) is a powerful tool when dealing with either changing network conditions or devices with different capabilities. Video sequences encoded with SVC, can easily be adapted to these changing requirements. As discussed in Section 2.1, the layer identification bits are signaled without entropy encoding, such that network components are able to route packets in a complexity efficient way. Resolution, quality, frame rate or a combination thereof can be reduced to cope with bandwidth fluctuations, different network characteristics (e.g., broadband vs. mobile network) or to adjust the stream to the capabilities of the receiving device. This approach reduces the quality in a controlled manner and will yield a higher Quality of Experience (QoE) compared to random packet loss. Randomly dropping packets when the available bandwidth is exceeded, results in a larger distortion compared to a controlled rate adjustment [1]. The reason is that distortions which occur in reference frames propagate through multiple frames, due to the temporal prediction. If the packets of such frames are dropped in the network, visible drift artifacts will be noticed. Therefore, scaling the bitstream in a controlled way will only drop those packets which are not referenced by any other frames. Typically, such non-referenced frames are assigned the highest temporal identifier (Tid) in hierarchical B-frame prediction, while reference frames have a lower Tid.

3.1.2 Device limitations

Moreover, also the device capabilities might require to reduce the bandwidth, independently of the available bandwidth of the channel. Firstly, receiving data requires processing, and thus energy consumption which reduces the battery life. Secondly, decoding more data results in an increased processing power. Thirdly, the device capabilities might not correspond with the characteristics of the encoded video stream. E.g., a device with a lower spatial resolution might not be able to decode and/or display HD resolution. Furthermore, the memory required to decode such bitstreams might not be available either. Additionally, devices might lack the processing power. Consequently, scaling SVC bitstreams in the network yields a higher user experience by adapting the video streams to the user requirements and network and device capabilities.

3.2 Related work

A comparison between SVC encoding and AVC transcoding shows that SVC encoding is preferred over transrating, although the encoding complexity is significantly higher [2]. However, it might not always be possible to adjust the encoding process. Consequently, transcoding still has to be performed if the constraints can not be met. However, by transcoding to SVC any future transcoding steps can be avoided. H.264/AVC-to-SVC transcoding is a relatively novel research area, although transcoding schemes have already been presented for each type of scalability.

Transcoding to temporally scalable SVC bitstreams is achieved by applying a hierarchical prediction structure [3]. H.264/AVC allows for hierarchical temporal prediction structures when encoding a video stream. However, this is mostly not desired by broadcasters in order to reduce the delay of the video delivery. The complexity of the temporal transcoding is reduced in [4] by limiting the motion vector search area based on the H.264/AVC motion vector. The temporal layer is taken into account to increase the search area since the motion vector will be larger due to the increased distance with the reference frame. An extensive research on the impact of the GOP size has been presented in [5]. In [6, 7] the temporally scalable transcoder is extented by averaging the input H.264/AVC motion vectors to the macroblock partition of the SVC layer. These temporal transcoding techniques are only applied to the highest temporal layers, since those have the highest complexity for transcoding. These techniques yield a 64% complexity reduction while a small increase of 1.4% in bit rate and a sligth decrease of 0.03dB in quality are noticed. Furthermore, machine

learning techniques have been used to reduce the mode decision complexity for temporal scalable transcoding [8]. Using machine learning a comparable RD performance is noticed for a reduction of 82% in complexity.

Transcoding to spatially scalable SVC bitstreams generates an SVC bitstream from a decoded H.264/AVC bitstream, with a reported gain of 60% in complexity due to fast mode decision [9]. This system works well for adapting bitstreams to different device characteristics. However, in the network, the bandwidth can only be limited by discarding the enhancement layer, which results in a reduction of the resolution. Therefore, the bit rate of both the resulting SVC bitstream and extracted base layer, cannot be controlled with fine granularity. Furthermore, upscaling the lower resolution video results in a low QoE .

Transcoding to CGS scalable SVC bitstreams is suggested to overcome this bandwidth issue and to preserve a high frame rate [10]. Using CGS, the bit rate can be controlled with a finer granularity and can be efficiently adapted to the available bandwidth. The open-loop transcoding architecture of [10] (shown in Figure 3.2) has an extremely low complexity, while the resulting enhancement layer quality is equal to the original H.264/AVC bitstream. This open-loop transcoding architecture creates the base layer by requantizing the H.264/AVC coefficients. To obtain the enhancement layer, the requantized base layer is subtracted from the H.264/AVC coefficients, this results in an enhancement layer which fully applies inter-layer residual prediction. Consequently, for the enhancement layer this yields the same coefficients, so the exact same decoded image is achieved. On the other hand, the extracted base layer will have drift effects because the quality reduction is performed in the transform domain, which does not maintain reference pictures for the resulting SVC bitstream. This can be seen in Figure 3.2, where no return path for a reconstructed picture to a reference buffer is provided. This transform domain transcoding results in faulty predictions which propagate through multiple frames.

Although the bandwidth of the base layer can be controlled by the open-loop requantization transcoding, the achieved base layer bit rates are much higher compared to those obtained with a cascaded decoder-encoder setup. This is because the requantization is not done for the intra predicted frames. Since such frames are referenced, measures have to be taken to reduce the drift effects. Therefore, in [10], the intra predicted frames are completely copied in the base layer, resulting in the limited scalability of the SVC bitstream. Consequently, further research towards a highly efficient transcoding system is necessary.

On the other hand, a closed-loop technique as presented in Section 3.3 is a cascaded decoder-encoder and will reduce the bit rate of the base layer

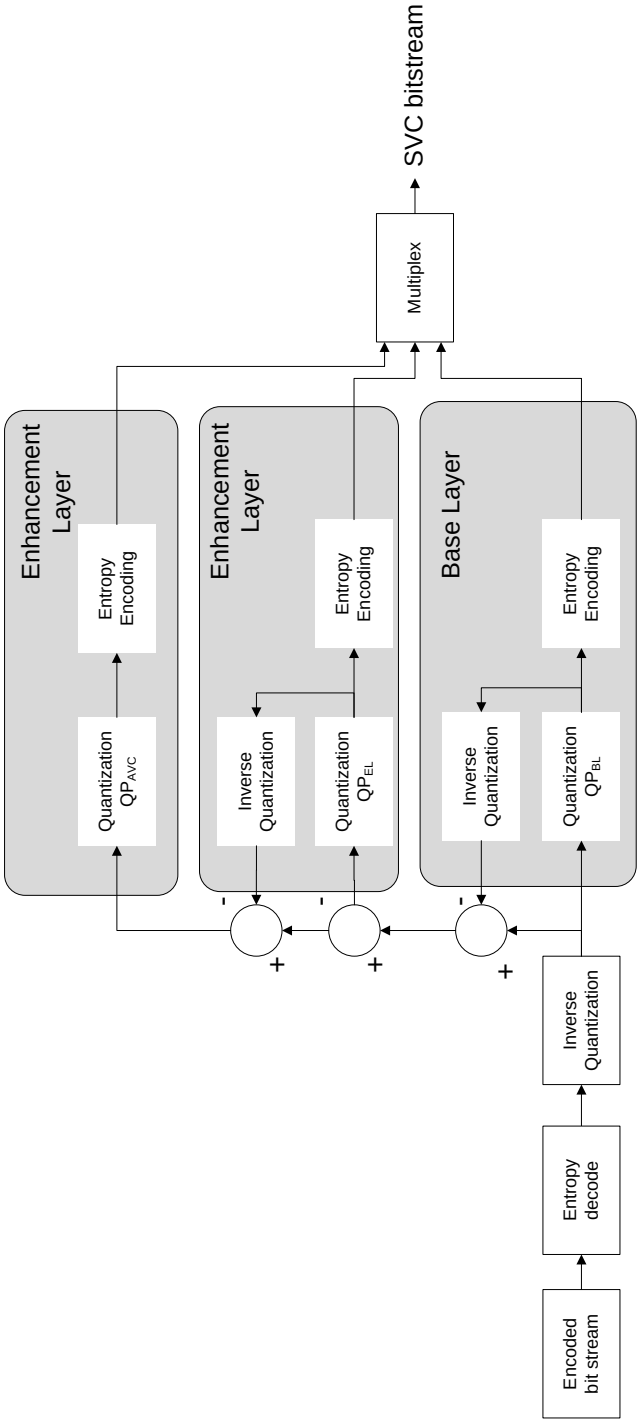


Figure 3.2: Schematical overview of an open-loop H.264/AVC-to-SVC with CGS transcoder.

and results in a higher degree of scalability. The drawback is that the SVC encoding depends on the decoded version of the H.264/AVC bitstream, consequently the quality of the closed-loop transcoded bitstream will be lower than the input bitstream. Previous work resulted in a simple closed-loop architecture, which has been presented in [11]. This architecture is based on an analysis of the macroblock modes in the original input H.264/AVC bitstream and the corresponding SVC bitstream. The analysis results in a fast mode decision model, which optimizes the encoder side of a cascaded decoder-encoder. Since this technique does not extensively exploit all information from the input H.264/AVC bitstream, this complexity can be further reduced by the closed-loop architecture proposed in this chapter.

Transcoding from H.264/AVC-to-SVC allows to exploit the benefits of scalability, while existing network infrastructure should not be replaced immediately. However, two main issues have to be tackled. Firstly, current closed-loop techniques have to be optimized in terms of complexity. Secondly, open-loop transcoding techniques suffer from drift effects, and result in a low RD performance for the base layer compared to closed-loop transcoding. In this chapter, techniques are proposed to reduce these two shortcomings of current systems. In a first approach (Section 3.3), the transcoding complexity of closed-loop transcoders is reduced, which is done by optimizing the encoding part of the transcoder. The mode decision step and the prediction lists are modified. Meanwhile, a motion vector refinement is applied, instead of a full motion vector search area.
Thereafter, the open-loop and closed-loop transcoders are combined in a hybrid transcoder (Section 3.4). The hybrid transcoder is designed to reduce the drift error due to the open-loop transcoder. Therefore, open-loop transcoding is only applied to the unreferenced frames (i.e., highest temporal layer). Meanwhile, all other frames are transcoded using the optimized closed-loop transcoder. Afterwards, the proposed hybrid transcoder is adjusted to regulate the number of open-loop transcoded frames so that the drift effects and complexity can be controlled. The resulting complexity-scalable system allows the transcoder to adjust the complexity to the available resources such as processing power and energy consumption, ultimately leading to green transcoding.
Results for the proposed closed-loop transcoder are given in Section 3.3.5. The hybrid transcoder is evaluated in Section 3.5 for both the RD performance of the complete bitstream, as well as the RD performance of the extracted base layer. Furthermore, measurements of the complexity and degree of scalability are given. Lastly, some possible additional optimization are proposed for future work (Section 3.7).

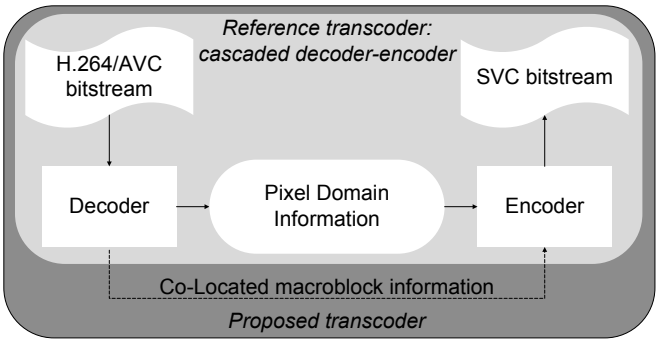


Figure 3.3: Cascaded decoder-encoder scenario.

3.3 Proposed closed-loop transcoder architecture

From an H.264/AVC encoded bitstream, an SVC CGS version is created. The quality of the enhancement layer is given by the maximum available quality of the H.264/AVC bitstream, i.e. the same quantization as the H.264/AVC bitstream is applied. To scale to lower rate points, the quantization of lower layers is increased in the SVC bitstream. Drift errors are avoided by applying closed-loop transcoding, based on a cascaded decoder-encoder scenario. Figure 3.3 shows the general concept. The encoder part is similar to Figure 2.2 where a feedback loop to a reference buffer is provided. A signaling path from decoder to encoder with co-located macroblock information of the H.264/AVC bitstream (macroblock mode, motion vectors, prediction list) is proposed to optimize the encoding process. This information reduces both the mode decision and sub-mode decision complexity by limiting the number of mode evaluations at the encoding part of the transcoder. Additionally, the prediction direction and the motion vector search range are reduced.

Since the proposed transcoder generates an SVC bitstream, both the base and enhancement layer should be generated. The proposed closed-loop transcoder optimizes for both layers the encoding process. The optimizations for the base layer are discussed in Section 3.3.1, while in Section 3.3.2 the enhancement layer optimizations are elaborated on. Furthermore, additional complexity reductions due to the list prediction are discussed in Section 3.3.3 and a complexity reduction for the motion estimation process is proposed in Section 3.3.4. Finally, results for the proposed closed-loop transcoder are given in Section 3.3.5 and concluding remarks can be found in Section 3.3.6.

3.3.1 Base layer mode decision

Since the base layer is H.264/AVC compatible, the mode decision process of the base layer is the same as for H.264/AVC. The normal mode decision process can be invoked, and for an unmodified cascaded decoder-encoder, the macroblock is encoded with the rate-distortion (RD) optimal mode, after evaluating all modes. In order to reduce the complexity, the number of evaluated modes is restricted. Therefore, the $MODE$ of the co-located macroblock of the H.264/AVC input bitstream ($MODE_{AVC}$) can be used as prior knowledge to bias the mode decision process. Since intra predicted modes correspond to a low complexity (Table 2.1), a full intra prediction step is still performed for all macroblocks. Consequently, only predictive mode evaluations are reduced, both for uni-directional (P frames) and bi-directional modes (B-frames).
The complete flowchart of the proposed base layer mode decision process is shown in Figure 3.4. Due to the higher quantization step size of the base layer, the probability for larger (sub-)macroblock partitions will typically increase (Section 2.3.2). Furthermore, *MODE_16×16*, $SKIP$ and $MODE\_Direct$ have a low complexity to evaluate (as seen in Table 2.1). Therefore, these modes are always evaluated in addition to $MODE_{AVC}$.
$MODE_{AVC}$ is evaluated because it is the most probable mode to be inherited in the base layer, since the lower quality (higher quantization step size) does not necessarily means that the most optimal mode differs between both layers. For sub-macroblock modes the same principles apply; when the H.264/AVC mode is *MODE_8×8*, the low-complexity sub-modes *sub_Direct* and *sub_8×8* are always evaluated. These have a low complexity and an increased probability for selection because these have fewer partitions. Modes *sub_4×8* and *sub_8×4* are only evaluated when these correspond to the sub-macroblock type of the co-located macroblock in the H.264/AVC bitstream (BLK_{AVC}) as indicated in Figure 3.4. Note that *sub_4×4* is never evaluated in the base layer since *sub_4×4* is affected by the increased partitioning size because of the high complexity and the reduced probability. [1]

3.3.2 Enhancement layer mode decision

Since the enhancement layer encodes the modes both with and without ILP, approximately 66% of the complexity is used for enhancement layer encoding with CGS. The enhancement layer encoding process can use the base

[1] Note that $sub\_x \times y$ is the sub-macroblock mode with size $x \times y$ for each sub-macroblock type (BLK).

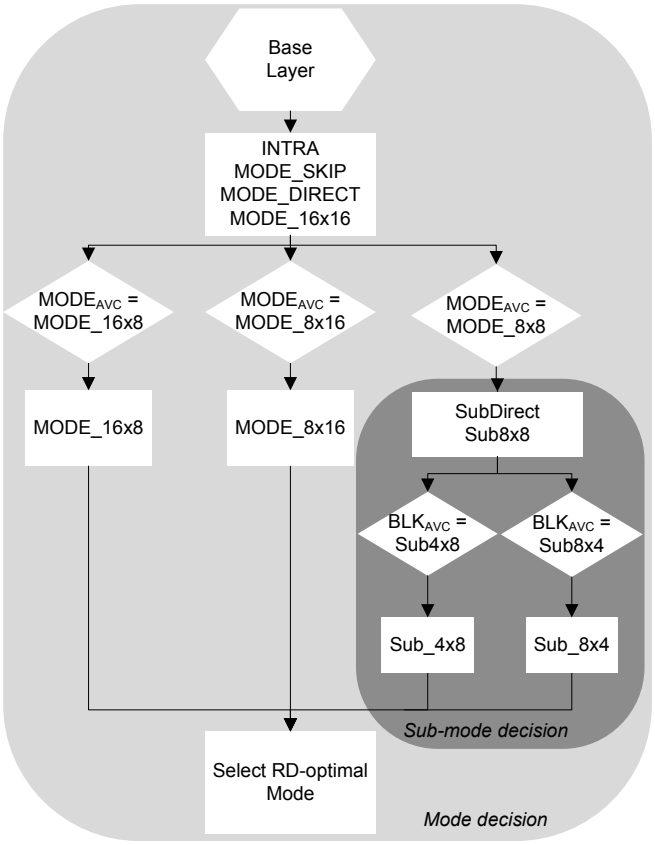


Figure 3.4: Flowchart for base layer (sub-)mode selection process.

layer information as a prediction by using ILP. The complexity for encoding the enhancement layer can be reduced, since a relation between the $MODE_{AVC}$ and the enhancement layer macroblock mode ($MODE_{EL}$) is established in [11]. This relation shows that typically $MODE_{AVC}$ or a non-partitioned macroblock mode is selected, for the enhancement layer. Therefore, the evaluated modes are limited to either the input macroblock mode ($MODE_{AVC}$), or a non-partitioned mode with a high probability. So for the base layer macroblock mode ($MODE_{BL}$) either the unpartitioned mode or $MODE_{AVC}$ will be selected. In case the $MODE_{AVC}$ is selected for the base layer mode, this mode will most likely yield an RD-optimal encoding considering the total bitstream and not only the local RD optimum for the layer. Indeed, ILP can be applied for this mode. When an unpartitioned mode is selected for the base layer, this mode might also be of interest for the enhancement layer. However, the other unpartitioned modes should not

be evaluated because the probability that these outperform $MODE_{AVC}$ and the most optimal unpartitioned base layer mode is unlikely. Additionally, $SKIP$ is evaluated because the enhancement layer reference picture might have been changed due to ILP within the reference picture. Therefore, using $SKIP$ might yield a better RD. However, no early skip termination is provided. Since the RD cost of $MODE_{BL}$ evaluated with ILP is not known, $MODE_{BL}$ might yield a lower RD cost compared to $SKIP$. To make this decision both modes should be evaluated. Additional complexity reduction is obtained, by evaluating $MODE_{BL}$ only with ILP, while a classical encoding step (without ILP) is applied for $MODE_{AVC}$. Consequently, if $MODE_{BL} = MODE_{AVC}$, one macroblock mode is evaluated completely, while $MODE_{BL} \neq MODE_{AVC}$ two macroblock modes are evaluated partly.

The sub-macroblock modes are evaluated either with or without ILP, depending whether *MODE_8×8* has been selected in the base layer or in the H.264/AVC bitstream. ILP is applied if the base layer has selected *MODE_8×8*. The complexity of *MODE_8×8* is further reduced by only evaluating *sub_Direct*, *sub_8×8*, and the co-located block size of the H.264/AVC bitstream. Note that, due to the increase in quality, for the enhancement layer encoding, *sub_4×4* might be evaluated, which might happen when the BLK_{AVC} is a *sub_4×4* partition. A schematic overview of the enhancement layer mode decision process is given in Figure 3.5.

3.3.3 Prediction direction

In B pictures (bi-)predictive or intra predictive modes can be used. When using intra prediction or bi-predictive coding, no further optimizations are applied. However, numerous macroblocks in B pictures are predictively coded by using only one of both prediction lists. Based on the similarities in content between the low and high quality content, it can be assumed that the same prediction list as the H.264/AVC macroblock is used for the SVC macroblock. Therefore, the (sub-)macroblock mode decision process only has to be performed for the corresponding prediction list. Consequently, for macroblocks which use only one prediction list, this yields a complexity reduction up to 66%, while no gain is achieved when bi-predictive macroblocks are encoded in the input H.264/AVC stream. Table 3.1 shows the probability that a uni-predicted (sub-)macroblock is present in the input H.264/AVC. The results are obtained after the transcoding has been performed by analysing the H.264/AVC input bitstreams (see Section 3.3.5 for an overview of the sequences and quantization settings). The table shows the probability (second column) that a macroblock is uni-directional pre-

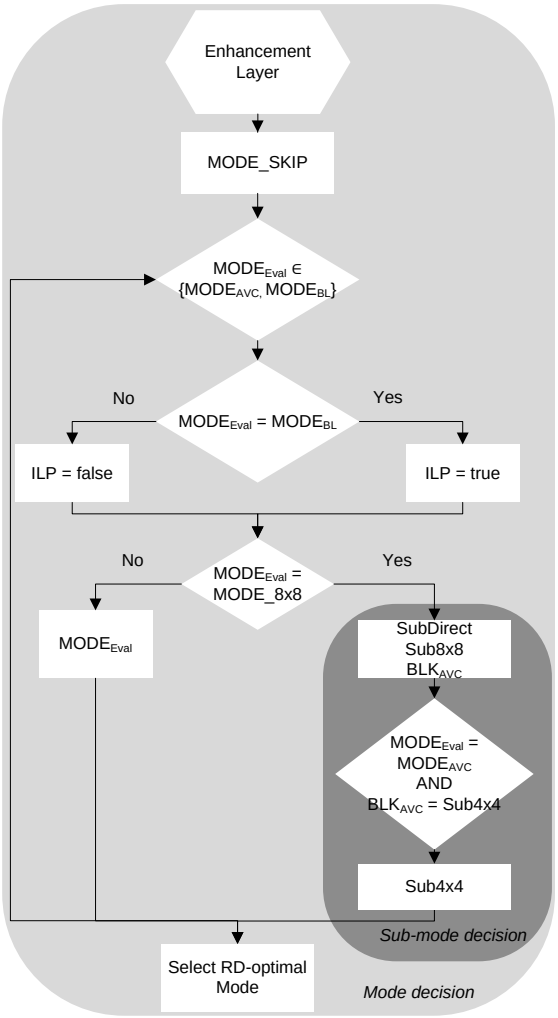


Figure 3.5: Flowchart for enhancement layer (sub-)mode selection process.

dicted (P-macroblock) or one of the partitions (e.g., $B\_L0\_Bi\_16\times8$) is uni-directional predicted in a bi-directional predicted frame (i.e., B-frame). The complexity reduction compared to fully evaluating the same macroblock is given in the third column. A total of 25.5% of all B-frame macroblocks are (partly) uni-directional predicted. Consequently, reducing the prediction direction has a significant impact on the total complexity gain.

Type	Probability	Complexity Reduction
non-partitioned	0.15	66%
both partitions uni-directional	0.04	66%
one partition un-directional	0.07	33%

Table 3.1: Analysis of the probability for a uni-directional prediction in bi-predictive coded macroblocks.

3.3.4 Motion vector estimation

Since the motion information is known from the H.264/AVC bitstream, it can be reused for the SVC bitstream. However, for both base and enhancement layer, a motion vector refinement is proposed for two reasons. First, the base layer has a reduced quality, which can result in a different motion vector compared to the enhancement layer. Second, because the base layer is used as a prediction for the enhancement layer, the enhancement layer motion vector can be different due to the ILP. Since the ILP might reduce the number of bits to signal the motion vector compared to the H.264/AVC input bitstream, inheriting the base layer motion vector might result in a lower RD cost. It should be noted that a less optimal motion vector might still result in low residual information. If the total cost of the ILP motion vector is lower than using the H.264/AVC motion vector, the motion vector with ILP will be selected.

The motion vector evaluation for the SVC bitstream uses the H.264/AVC motion vector as a starting point. Afterwards, a motion vector refinement is performed, which is evaluated within a Search Window (SW) for both base (SW_{BL}) and enhancement layer (SW_{EL}). For both layers, combinations of multiple search window sizes have been evaluated $(SW_{BL}, SW_{EL}) \in \{(1,1),(4,2),(8,1),(8,8),(16,8),(16,16)\}$ using six test sequences: *Harbour*, *Ice*, *Rushhour*, *Soccer*, *Station* and *Tractor*. $SW = (16,16)$ yields a higher complexity but, unexpectedly, does not result in a better RD. Figure 3.6 shows the RD-curves for the tested extrema $SW = (1,1)$ and $SW = (16,16)$ compared to the an unmodified cascaded decoder-encoder scenario (with a search range of 32 pixels without using an initial motion vector). Two sequences (*Ice* and *Station*) are shown, which have respectively the smallest and largest RD performance loss. As can be seen in Figure 3.6(a) and 3.6(b), $SW = (1,1)$ outperforms or equals the RD performance of $SW = (16,16)$. This is because a fast motion search is performed to limit the encoding complexity in the base layer. When using large search windows, this might let the motion vector drift further

(SW_{BL}, SW_{EL})	Delta BR (%)	Delta PSNR (dB)	Complexity vs (16,16)
(16, 16)	0.001	-0.108	1
(16, 8)	0.031	-0.111	0.93
(8, 8)	-0.004	-0.091	0.90
(8, 4)	0.001	-0.097	0.88
(8, 1)	0.035	-0.104	0.87
(4, 2)	0.026	-0.103	0.84
(1, 1)	-0.026	-0.085	0.82

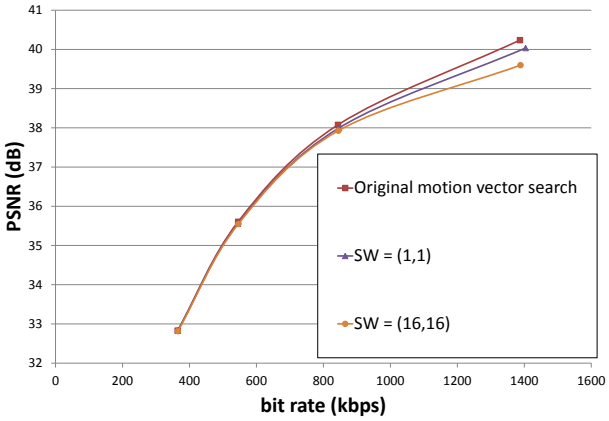
Table 3.2: RD performance of the different search window (SW) sizes for the motion refinement in the base and enhancement layer compared to a cascaded decoder-encoder. Due to RD optimization, $SW = (1, 1)$ is preferred.

away from the most optimal motion vector (inherited from the enhancement layer) since such algorithms can get trapped in local optima. Moreover, to determine the cost of a motion vector, SAD (Sum of Absolute Differences) is used rather than the number of bits after entropy encoding.

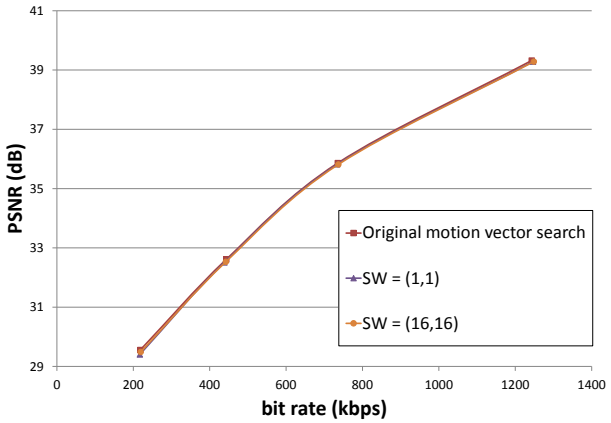
An overview of the BDRate and BDPSNR for the evaluated search window sizes is given in Table 3.2. Additionally, the complexity of the reduced search window size is given. This complexity represents the complexity of the complete transcoding system[2], not only the motion estimation complexity. Since a single pixel search window shows the best results, in the following all results are discussed with a single pixel search window size for the motion vector refinement of the base and enhancement layer. This eliminates the need for a fast motion estimation algorithm, while the RD-performance approximates the full search RD. Note that still a sub-pixel evaluation is performed, with a quarter pixel accuracy.

The motion estimation process is applied for each mode that has to be evaluated according to the flowchart in Figure 3.4, and is specified in Figure 3.7. $MODE_{EVAL}$ indicates the mode that is currently evaluated and MV_{init} indicates the initial motion vector used for the motion estimation. MV_{init} can be either MV_{AVC}, which is the inherited motion vector from the H.264/AVC bitstream, or MV_{pred} which is the predicted motion vector based on the neighboring macroblocks.

[2]Obviously, in a cascaded decoder-encoder the complexity reduction would be higher because for all macroblock modes the motion estimation complexity is reduced significantly. On the other hand, the proposed system only limits the motion estimation complexity for a limited number of macroblock modes.



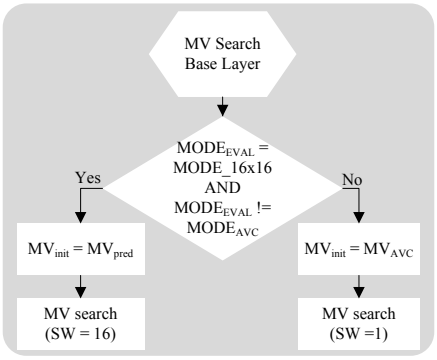
(a) Sequence *Ice* (ΔQP= 5)



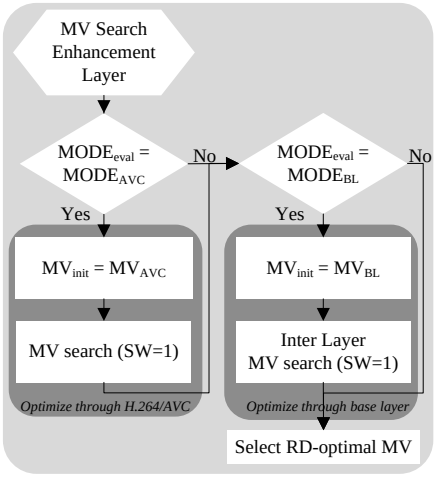
(b) Sequence *Station* (ΔQP= 5)

Figure 3.6: RD-curves for two sequences showing the impact of different search window sizes compared to a cascaded decoder-encoder with a search range of 32 pixels and without using an initial motion vector.

For the base layer a normal motion estimation step is only applied when *MODE_16×16* is evaluated and if this mode has not been selected in the H.264/AVC input bitstream. In such cases *MODE_16×16* is evaluated because the coarser quantization of the base layer might result in this mode, so no relation between the used H.264/AVC motion vectors can be exploited, e.g., by interpolating the motion vector of the different partitions. Obvi-



(a) Optimized base layer motion estimation



(b) Optimized enhancement layer motion estimation

Figure 3.7: Flowcharts of the optimized motion estimation processes for base and enhancement layer. $MODE_{EVAL}$ represents the mode which is currently being evaluated.

ously, when *MODE_16×16* is evaluated and this mode has been selected in the input H.264/AVC bitstream, the H.264/AVC motion vector is used as an initial motion vector that has to be refined. To signal the motion vector, a motion vector predictor as with H.264/AVC encoding is used. This predictor is based on median filtering of motion vectors of three surrounding macroblocks.

For the enhancement layer, a $SW = (1, 1)$ is applied around MV_{init}. This MV_{init} is either the motion vector inherited from the H.264/AVC bit-

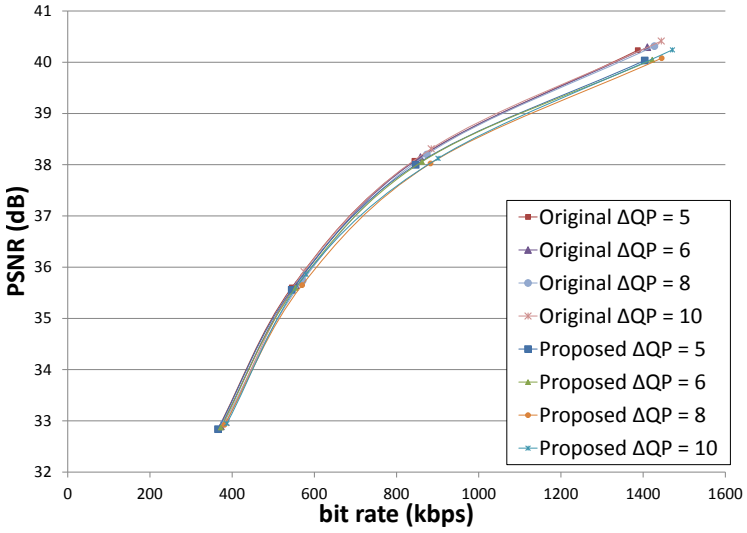


Figure 3.8: RD performance for the proposed method compared to the reference transcoder for sequence *Ice* with $\Delta QP \in \{5, 6, 8, 10\}$.

stream or the base layer motion vector. The former case occurs when the $MODE_{AVC}$ is being evaluated in the enhancement layer. In the latter case, the base layer already performed a motion vector refinement (or a completely new motion vector is evaluated in case of *MODE_16×16*). Therefore, only a refinement in a small window around the enhancement layer motion vector is required to cope with the improved visual quality and increased level of detail. When the base layer motion vector is inherited, ILP is forced since the motion vector can be predicted easily from the base layer. When $MODE_{AVC}$ is evaluated no ILP should be applied since there will not be a high correlation between base and enhancement layer. Note that both $MODE_{EVAL} = MODE_{AVC}$ and $MODE_{EVAL} = MODE_{BL}$ are evaluated. However, when $MODE_{AVC} = MODE_{BL}$, the complexity is not increased because the same mode is evaluated, once with ILP and once without ILP. The reference implementation always evaluates each mode both with and without ILP.

3.3.5 Results for the proposed closed-loop transcoder

The closed-loop architecture is evaluated using the same six test sequences with a 4CIF resolution as used in Section 3.3.4. Each sequence is encoded as an H.264/AVC bitstream with the H.264/AVC QP (QP_{AVC}):

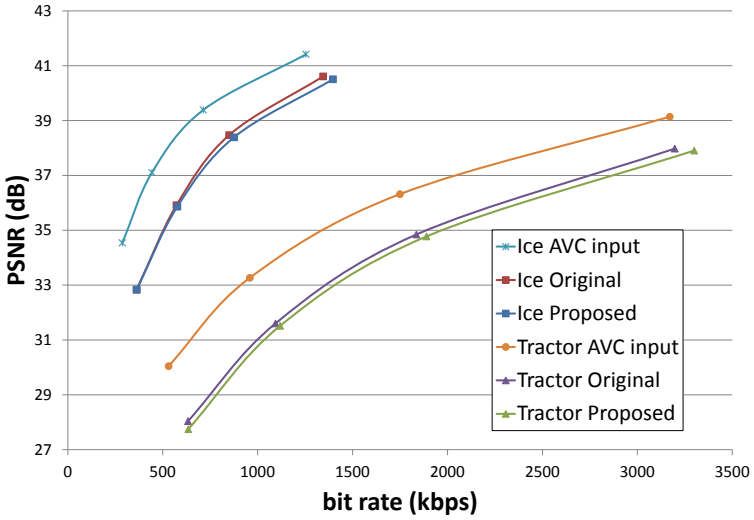


Figure 3.9: RD performance for the proposed method compared to the reference transcoder for the two worst performing sequences (*Ice* and *Tractor*) with $QP_{BL} = 47$.

$QP_{AVC} \in \{27, 32, 37, 42\}$. The input bitstream is transcoded to an SVC CGS bitstream, with a base layer QP (QP_{BL}): $QP_{BL} = QP_{AVC} + \Delta QP$ and enhancement layer QP (QP_{EL}): $QP_{EL} = QP_{AVC}$. ΔQP is the difference in quantization between the base and enhancement layer of the SVC bitstream. Depending on the amount of bits that should be reserved for the base layer, the ΔQP can be defined. To evaluate the system, different ΔQP values have been evaluated: $\Delta QP \in \{5, 6, 8, 10\}$. Additionally, also a scenario with a constant base layer quality ($QP_{BL} = 47$) for all rate points is applied.
The proposed system is based on the reference software used for SVC (JSVM_9_19_9) [12] and is compared against a reference transcoder. This reference transcoder is a cascaded decoder-encoder of the same reference software and is unmodified in terms of RD-performance or complexity. The reference transcoder is used to evaluate the proposed system for both RD performance and complexity.

Rate distortion analysis

The RD performance of sequence *Ice* is shown in Figure 3.8 for all ΔQPs. For each ΔQP the performance is comparable. However, the loss in RD

	ΔQP= 5		ΔQP= 10	
	BDPSNR	BDRate	BDPSNR	BDRate
Harbour	-0.038	1.018	-0.070	1.849
Ice	-0.089	1.658	-0.192	3.242
Rushhour	-0.038	0.821	-0.091	1.983
Soccer	-0.057	1.239	-0.174	3.877
Station	-0.051	0.964	-0.204	3.704
Tractor	-0.148	2.673	-0.413	7.295
Average	-0.070	1.396	-0.191	3.659

Table 3.3: BDPSNR and BDRate for ΔQP= 5 and ΔQP= 10.

performance compared to the reference transcoder, is slightly higher when ΔQP increases. On average, a bit rate increase of 0.53% and 1.41% for a PSNR decrease of 0.11 dB and 0.12 dB are obtained for $\Delta QP = 8$ and $\Delta QP = 10$, respectively. Figure 3.9 shows the rate distortion of the two worst performing sequences (*Ice* and *Tractor*). As can be expected for transcoding to a constant QP_{BL}, the proposed system performs better at low rate points, because of the small ΔQP for these points. For higher rate points the impact of the larger ΔQP results in a slightly lower RD performance compared to the reference transcoder.
Table 3.3 shows the BDRate and BDPSNR for the best ($\Delta QP = 5$) and worst ($\Delta QP = 10$) performance of the proposed transcoder compared to the reference transcoder. As can be seen, only small Bjøntegaard measures are reported. Consequently, the proposed transcoder results in only small bit rate differences for the same quality compared to the reference transcoder. As can be seen in Table 3.2, the average nominal bit rate for $\Delta QP = 5$ is reduced by 0.026% with a ΔPSNR = -0.085 dB[3] compared to the reference transcoder.

Complexity

The complexity of the system is evaluated as the time saving (TS) obtained by the proposed transcoding and is given by Equation 2.6. Because only the (sub)mode decision process is modified, the time saving reflects the complexity decrease for these modifications within the same code base. The complexity reductions for base layer, enhancement layer and the full sys-

[3]The worst performance compared to the reference transcoder is noticed for $\Delta QP = 10$, where a 1.41% bit rate increase and a -0.12 dB PSNR decrease are measured.

tem are given in Table 3.4. On average, only 8.3% of the complexity of a cascaded decoder-encoder is required to perform the same transcoding operation. The reason for this high complexity reduction is the fact that the list of evaluated modes is reduced significantly. For the base layer, only the H.264/AVC macroblock mode is evaluated, and additionally the non-partitioned low-complexity modes. Furthermore, the complexity for the motion estimation is limited to only a non-partitioned motion estimation, which in itself is already low-complex. This ensures a low complexity overhead and high probability to select the correct base layer macroblock mode. For the enhancement layer, the H.264/AVC mode is evaluated only without ILP, while the base layer mode is only evaluated with ILP. Meanwhile, almost no complexity is used for the enhancement layer motion estimation. Furthermore, the complexity reduction is likely to be content independent, since a large set of video content is used to cover different video characteristics while similar complexity reductions are achieved. For the whole system a difference in complexity of 0.45% is noticed. The complexity reduction will differ from real-world commercial solutions. However, JSVM is widely known and can be used as a common ground for comparison.

Comparison with existing techniques

Since there has not been a lot of investigation in the field of H.264/AVC-to-SVC transcoding, the number of algorithms to compare with is limited. To have a common ground of comparison, only techniques are considered which are able to transcode towards a quality scalable bitstream. Consequently, the presented system will not be compared with [9] and [13] since these techniques do not provide such a fine granularity for the rate points of the resulting SVC bitstream.

- **Compared to closed-loop transcoding**

Only one closed-loop transcoding algorithm has been previously proposed. This algorithm achieves a complexity reduction of 57% with only a small loss in RD performance of 6.7% in BDRate [11]. As can be seen, both the complexity as well the RD performance of the proposed closed-loop model outperforms the technique presented in [11].

- **Compared to open-loop transcoding**

As was pointed out in Section 3.1, an open-loop transcoding mechanism for H.264/AVC-to-SVC with quality scalability has already been proposed [10]. As can be seen in Figure 3.2, the open-loop transcoding only applies

	Average complexity reduction (%)		
	Base Layer	Enhancement Layer	Full System
Harbour	85.83	94.48	91.53
Ice	85.28	95.03	91.65
Rushhour	85.69	94.87	91.75
Soccer	85.43	94.72	91.52
Station	86.40	94.90	91.98
Tractor	86.76	94.44	91.73
Average	85.90	94.74	91.69

Table 3.4: Complexity reduction for the proposed closed-loop transcoding architecture.

an entropy decoding, dequantization and requantization step, the required complexity is very low. Compared to the cascaded decoder-encoder, near 100% complexity reduction is achieved. Obviously, in terms of complexity, open-loop transcoding outperforms the proposed method.
On the other hand, the rate distortion is strongly influenced. Open-loop transcoding results in a higher quality for the enhancement layer, since no decoding step is applied on the input H.264/AVC bitstream. Consequently, the original encoded quality is maintained. However, the bit rate drastically increases compared to the reference transcoder, specifically for the base layer, as can be seen in Figure 3.10. Mainly because all intra-coded macroblocks are encoded in the the base layer. Consequently, the degree of scalability is reduced, i.e. the required bit rate for the base layer is increased, and the share of the base layer in the total bit rate is increased as well. Furthermore, the closed-loop techniques do not yield drift artifacts, resulting in an increased QoE when the base layer is received.

- **Compared to fast mode decision models for SVC**

In the past, many fast mode decision models for SVC have been proposed. None of these models are optimized for encoding with the prior knowledge of an H.264/AVC bitstream. Li's model [14], one the most referred models in literature, uses base layer information to reduce the complexity of the enhancement layer encoding. In Chapter 2 generic techniques have been suggested to improve SVC enhancement layer encoding. As pointed out in Section 2.5 these techniques[4] are generic in a sense that they can be

[4]Disallowing orthogonal macroblock modes, only evaluating sub8x8 blocks if these are present in the base layer and only evaluating the base layer list predictions.

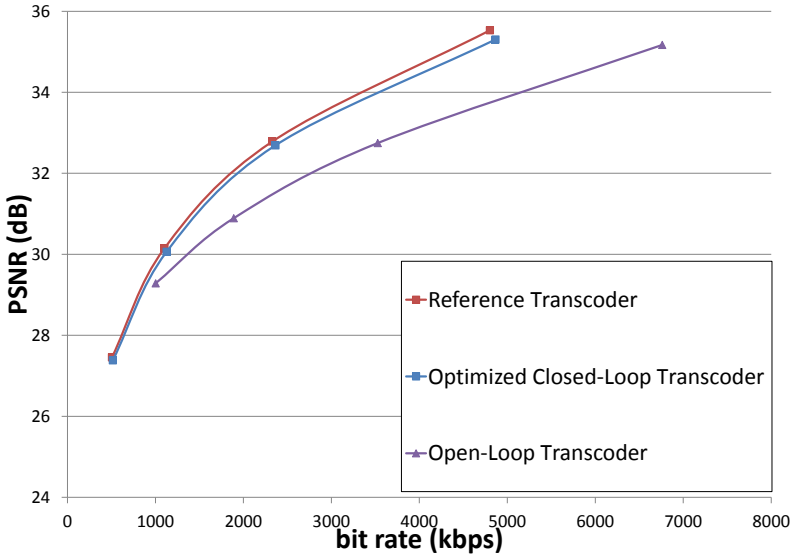


Figure 3.10: RD for extracted base layer of *Harbour* with ΔQP= 5.

adopted by existing fast mode decision models. Therefore, to improve Li's model in terms of complexity, the proposed three generic techniques have been incorporated in Li's model. This results in a low-complexity encoder, which in turn can then be used as the encoding part of a transcoder. So, the SVC encoder based on Li's model extended with the three proposed generic techniques is incorporated in a cascaded decoder-encoder.The proposed H.264/AVC-to-SVC closed-loop transcoding technique is compared with the results reported for this optimized cascaded decoder-encoder.

The complexity of the extended Li's model is only reduced for the enhancement layer, since the base layer encoding is not optimized. The required lowest complexity for the enhancement layer encoding is still 12.73% on average. Since the base layer encoding takes approximately 33% of the total complexity (due to the ILP), compared to a reference transcoder, a reduction of 54.27% is achieved compared to 91.69% for the proposed closed-loop approach. For the worst performing closed-loop scenario, $\Delta QP = 10$, a bit rate increase of 1,41% and a PSNR reduction of -0.12dB is reported. So all evaluated ΔQPs outperform the extended Li's model, resulting in an average bit rate increase of 2.14% and a PSNR of -0.36dB.

The significantly lower RD for existing fast mode decision models is due to exploiting the low quality base layer signal, which yields a less optimal enhancement layer, resulting in a low RD performance. On the other hand, exploiting also H.264/AVC information reduces the complexity but also significantly improves the global RD of the system. Since the best prediction for the high quality signal is known from the input bitstream, the base layer might be less efficient. However, this is greatly compensated by selecting the best macroblock mode for the enhancement layer. This is in line with the ideas and results for cross-layer optimization [15].

3.3.6 Conclusion for the proposed closed-loop transcoder

To reduce the complexity of the transcoding step, an optimized closed-loop transcoding scheme is described. By reducing the number of modes and optimizing the mode decision process, a low complex closed-loop transcoder is obtained. Only 8.3% of the complexity is required compared to a cascaded decoder-encoder scenario, while bit rate and quality remain stable. This complexity reduction results either in processing more bitstreams or consuming less energy with the same equipment. Compared to the existing optimized closed-loop transcoder [11], the complexity is further reduced, while the RD is improved. Additionally, the drawbacks of an open-loop encoder are tackled. No drift artifacts are introduced, the bit rate is reduced and the degree of scalability is increased.
Reducing the complexity of transcoding systems will result in cheaper hardware and lower operating costs. Even though only 8.3% of the complexity compared to a cascaded decoder-encoder scenario is required, the complexity of the transcoding step can be further reduced. By combining the proposed closed-loop transcoding with an open-loop transcoder the complexity is reduced significantly. On a frame basis either a closed-loop or open-loop transcoding step can be applied, resulting in a hybrid transcoder.

3.4 Proposed hybrid transcoder architecture

To further reduce the complexity of the transcoding process, compared to the proposed closed-loop transcoder in Section 3.3, a hybrid transcoder is proposed. The hybrid architecture combines a closed-loop and open-loop transcoder, hence the name. The quantization for each macroblock in the highest quality layer is maintained, while bit allocation for the SVC layer adjusts the QP difference (ΔQP) between layers. A higher ΔQP results in a lower bit rate for the base layer, but also in a higher difference between bit rates for both layers.

3.4.1 Closed-loop Transcoder

The hybrid transcoder applies the closed-loop transcoder as presented in Section 3.3 to transcode the reference frames. The encoding part is similar to the encoder presented in Figure 2.2. The lowest computation complex closed-loop architecture is used. So the (sub-)mode decision optimizations for for base and enhancement layer as presented in Figure 3.4 and Figure 3.5 are applied. Moreover, optimizations for the prediction direction and motion vector estimation are applied, as shown in Figure 3.7. For the motion vector refinement, a search window size of 1 is used for base and enhancement layer. A hybrid transcoder only using the closed-loop transcoding part yields the same results as the transcoder presented in Section 3.3.

3.4.2 Open-loop Transcoder

An open-loop transcoder as presented in [10] (Figure 3.2) is used, which divides DCT-coefficients over different layers by applying a inverse quantization followed by a quantization step, i.e., requantization. This requantization (for the base layer) reduces the quality of residual information, which will lower the visual quality of the frames. Because these macroblocks are referenced in the base layer, errors propagate through other macroblocks which make use of these adjusted values. The referencing macroblocks are not aware of the changed decoded output, so drift errors arise. This error propagation can be avoided by applying open-loop transcoding to non-referenced frames only, solely resulting in a reduced quality of those frame. The open-loop architecture guarantees a low complexity, but has a reduced base layer quality for the same bit rates compared to a cascaded decoder-encoder transcoder.

Moreover, in [10] it was decided not to requantize the intra-coded macroblocks. In order to control the drift, the intra-coded macroblocks are copied completely to the base layer. Although the bandwidth of the base layer can be controlled by the open-loop requantization transcoding, the achieved base layer bit rates are much higher compared to those obtained with a cascaded decoder-encoder setup, resulting in a limited scalability of the SVC bitstream. This limited scalability can be seen in the hybrid transcoding results in Figure 3.15, which shows that the base layer bit rate is almost doubled compared to the cascaded decoder-encoder scenario.

Combining the optimized closed-loop transcoder with the existing open-loop transcoder into the proposed hybrid transcoder will reduce the complexity of the closed-loop transcoder, while the quality of the base layer and the degree of scalability are increased compared to open-loop transcoding.

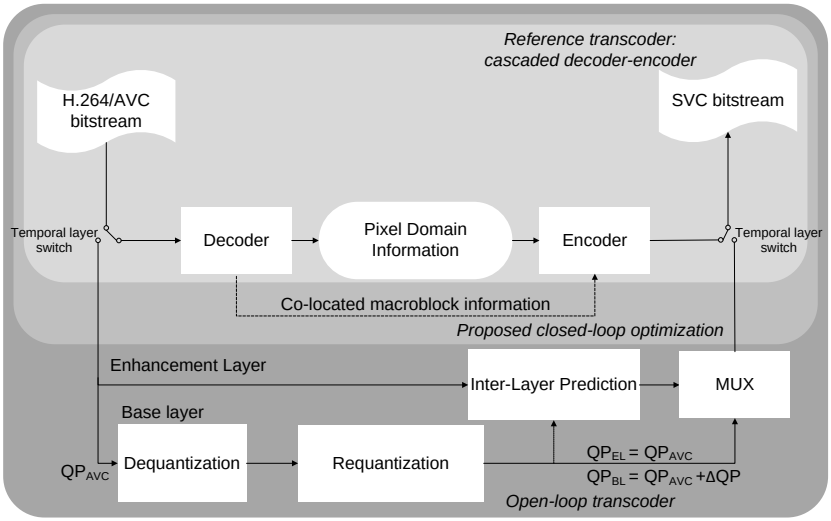


Figure 3.11: Overview of the proposed combined open- and closed-loop architecture for H.264/AVC-to-SVC transcoding.

3.4.3 Hybrid transcoder

A hybrid transcoder combines the advantages of an open- and closed-loop transcoder. It further reduces the closed-loop complexity, while improving the quality and scalability of the open-loop transcoder. In this solution, an open- or closed-loop transcoding step is applied depending on the temporal layer (identified by Tid) of the frame, as can be seen in Figure 3.11. If a frame is not referenced, (i.e., the highest temporal level) an open-loop transcoder is applied since only this frame might have transcoding artifacts, therefore no drift effects occur. Scalability is slightly reduced compared to closed-loop, although this holds only for one frame so the total effect is limited.

If a lower complexity is required, additional frames can be open-loop transcoded. This is indicated as Hybrid $Tid \geq x$, where all frames with $Tid \geq x$ are open-loop transcoded, while all other frames are closed-loop transcoded. For example with a GOP of size 8, less than half of the complexity is needed when transcoding frames open-loop with $Tid \geq 2$ compared to $Tid \geq 3$, as shown in Table 3.5. Consequently, drift effects might occur in frames that reference the open-loop transcoded ones, i.e., frames with $Tid = 3$ can suffer from drifting artifacts. However, since at most three consecutive frames are affected, drifting artifacts on the base layer will be less visible. Moreover, these three consecutive frames are in view-

Level	Type	Complexity Reduction	Frames/GOP open-loop transcoded	BDRate
1	Open-loop	99.99%	8	-19.33%
2	Hybrid $Tid \geq 1$	99.28%	7	3.80%
3	Hybrid $Tid \geq 2$	98.10%	6	6.86%
4	Hybrid $Tid \geq 3$	95.73%	4	7.04%
5	Closed-loop	91.52%	0	1.40%

Table 3.5: Complexity levels for a GOP size of 8 frames compared to a cascaded decoder-encoder. Hybrid $Tid \geq x$ means hybrid transcoding where frames with $Tid \geq x$ are open-loop transcoded. The average BDRate for $\Delta QP = 5$ is given to show the complexity versus RD trade-off.

ing order; because of the hierarchical prediction structure, the drift only occurs in two consecutive frames in coding order. The first coded frame ($Tid = 2$) is affected, as well as the previous and following frame in viewing order ($Tid = 3$), which are the following two frames in coding order. Note that the open-loop transcoded frames have a higher quality for the enhancement layer because the higher quality of the input signal remains, while closed-loop transcoding applies an additional quantization on this signal. Consequently, open-loop transcoded frames increase the average PSNR of a sequence, although drifting artifacts reduce the QoE .
If even less complexity is available, the system can further shift towards an open-loop transcoding design, by increasing the number of open-loop transcoded frames, ultimately reaching the open-loop scenario. Therefore, the proposed hybrid transcoding system is able to scale the complexity on a per frame basis ranging from optimized closed-loop transcoding to open-loop transcoding, inclusive. This way, the advantages and disadvantages of an open- and closed-loop system can be leveraged depending on the currently available resources.

3.5 Results

The hybrid transcoder is able to transcode a bitstream with constant bit rate and to divide the bit budget of the output bitstream over the different layers by adjusting the ΔQP. However, to investigate the impact of the proposed scheme on the RD performance, the input H.264/AVC bitstreams have a constant quantization. Consequently, the impact of any rate control mechanism is eliminated, which might give different distortions depending on

the H.264/AVC encoder used. The proposed system is evaluated against a cascaded decoder-encoder configuration without improvements in terms of scalability, complexity, RD performance and drift effects. The evaluation is based on the same six commonly used test sequences with 4CIF resolution as in Section 3.3.4[5]. Each test sequence was encoded as an H.264/AVC bitstream with an intra period of 32 frames while different quantization parameters were applied: $QP_{AVC} \in \{27, 32, 37, 42\}$.
The H.264/AVC input bitstreams were transcoded to SVC bitstreams having two CGS quality layers. To show the opportunities for bit allocation per layer, multiple ΔQP values have been applied between the base and enhancement layers: $\Delta QP \in \{5, 6, 8\}$ ($QP_{BL} = QP_{AVC} + \Delta QP$). The enhancement layer quantization corresponds to the maximal available quality ($QP_{EL} = QP_{AVC}$). The motion vector refinement search window size for both layers is one pixel. A GOP size of 8 frames with a hierarchical prediction structure is applied, resulting in a maximal $Tid = 3$. Consequently five levels of complexity scalability are available, as enumerated in Table 3.5. All bitstreams were generated using the Joint Scalable Video Model reference software (JSVM_9_19_9) [12].

3.5.1 Complexity

The reduction in complexity is expressed as the time saving (TS) for encoding with the improved transcoder compared to the original cascaded decoder-encoder, and is given by Equation 2.6.
The optimized closed-loop system has an average time saving of 91.5%. This implies that less than 10% of the original complexity is needed. When multiple frames of a GOP are encoded using an open-loop transcoder, the complexity will even further reduce, since the open-loop decoder nearly has the same complexity as a parser. Depending on the number of open-loop transcoded frames, the complexity can be further reduced, as shown in Table 3.5, where the last column indicates the total number of open-loop transcoded frames per GOP. Open-loop transcoding nearly equals the complexity of a parser. The difference between level 1 and 2 is the closed-loop transcoding of the key pictures (P-frames or intra-frames). Since the open-loop approach results in a high bit rate cost for intra frames in the base layer, closed-loop encoding these frames yield a higher scalability and better RD-performance for the base layer, furthermore closed-loop transcoding the P-frames will reduce the drift. Less than 1% of the complexity is needed to do so. Consequently, depending on the currently available

[5] *Harbour*, *Ice*, *Rushhour*, *Soccer*, *Station* and *Tractor*.

resources (energy, processing power,...) the transcoding design can be dynamically changed on a per frame basis to meet the constantly changing requirements. However, it is suggested not to use open-loop transcoding for intra predicted frames when this is not strictly necessary.
In case a GOP of 16 frames is used, the complexity levels as shown in Table 3.5 will be approximately the same. However, the complexity reduction for $Tid \geq 3$ in a GOP of 8 frames corresponds to $Tid \geq 4$ in a GOP of 16 frames. The complexity reduction for $Tid \geq 1$ in a GOP of 16 frames will be around 99.64%. This is roughly half of the remaining complexity since the number of intra predicted frames is halved. This number is derived by replacing the complexity for intra-predicted frames in a GOP of 8 frames that are not intra-predicted coded in a GOP of 16 frames with the complexity for open-loop transcoding, which is near the complexity of a parser.

3.5.2 Rate distortion

Entire SVC bitstream

The RD of the entire SVC bitstream is considered when all layers are taken into account. The RD-curves for the proposed transcoding schemes are shown in Figure 3.12. The open-loop RD-curve outperforms all other designs because the same visual quality as the input H.264/AVC bitstream is achieved. Closed-loop transcoded frames (both hybrid as well as the reference transcoder) will have lower PSNR values due the distorted version of the input bitstream which is used for the encoding step. Figure 3.12 shows that the optimized closed-loop transcoder has a slightly lower compression efficiency than the original closed-loop, while requiring less than 10% of the complexity. A Bjøntegaard Delta bit rate (BDRate) of -1.76% and a Bjøntegaard Delta PSNR (BDPSNR) of -0.085 dB are achieved compared to the cascaded version, which mean a nearly identical bit rate and quality. Because the additional quantization of the input signal for the closed-loop encoder compared to the open-loop encoder, the coding efficiency of the latter is higher. When combining open- and closed-loop transcoding, the bit rate and quality of the open-loop transcoded frames result in a higher RD compared to the closed-loop scenario. These situations correspond to the $Hybrid\ Tid \geq x$ curves in Figure 3.12, where frames with $Tid \geq x$ are open-loop transcoded.
Table 3.6 shows the BDRate and BDPSNR of the complete bitstream for each architecture after transcoding compared to the reference transcoder. As can be seen, open-loop transcoding gains significantly compared to the reference transcoder. This gain comes from the higher quality of the en-

hancement layer, since the same quality as the H.264/AVC input bitstream is achieved. For additional reference, Figure 3.12 shows the RD curve for the H.264/AVC input sequence.

Extracted base layer bitstream

Open-loop drift artifacts for the base layer are reduced in the hybrid scenarios, as can be seen in Figure 3.13. The best performance for the base layer is obtained by the reference transcoder (the unmodified decoder-encoder architecture). The optimized closed-loop transcoder has the second best performance, followed by the hybrid transcoding configurations. Note that no drifting artifacts can arise when only one consecutive frame is open-loop transcoded, since drift is the propagation of coding artifacts through the stream. Increasing the number of open-loop encoded frames per GOP will introduce error drift. However, these artifacts will be less visible compared to an open-loop transcoder, because important reference frames are still closed-loop transcoded. This explains the significant RD performance loss for the open-loop transcoding for the base layer. Figure 3.14 shows the drift effect of the proposed hybrid transcoder and the PSNR for that picture (frame 50 of sequence *Tractor*).
All results for the extracted base layer bitstreams are shown in Table 3.7 by the BDPSNR and BDRate values compared to an unmodified cascaded decoder-encoder. These results indicate the reduction of artifacts for the hybrid scenario compared to the open-loop scenario. Furthermore, it shows that an increasing ΔQP will only slightly reduce the performance of the proposed system.

3.5.3 Scalability

No real measure to express scalability exists. However, the ratio of the base layer bit rate to the overall bit rate should be low, to allow as many devices as possible to receive the base layer. Figure 3.15 shows the base layer bit rate in relation to the full bit rate. It can be seen that the base layer for the open-loop transcoder scenario requires a higher bit rate compared to all other scenarios. Because of the higher requirements for the lowest bandwidths, the open-loop systems have a lower degree of scalability. This is mainly because all intra coded macroblocks are completely encoded in the base layer. However, the increase in quality is not in relation to the increase in bandwidth, as shown in Figure 3.13. As expected, the degree of scalability of the closed-loop and hybrid systems is much higher, resulting in a higher QoE for end users.

		Open Loop		$Tid \geq 1$		$Tid \geq 2$		$Tid \geq 3$		Closed-Loop	
		BDPSNR	BDRate	BDPSNR	BDRate	BDPSNR	BDRate	BDPSNR	BDRate	BDPSNR	BDRate
$\Delta QP = 5$	*Harbour*	1.19	-26.70	0.17	-3.71	-0.02	0.89	-0.10	2.69	-0.04	1.02
	Ice	0.75	-11.84	-0.31	7.00	-0.48	9.95	-0.48	9.52	-0.09	1.66
	Rushhour	1.02	-17.40	-0.37	10.11	-0.51	12.70	-0.46	10.90	-0.04	0.82
	Soccer	1.17	-22.51	0.03	-0.46	-0.10	2.25	-0.17	3.85	-0.06	1.24
	Station	1.23	-13.77	-0.54	11.44	-0.62	13.12	-0.55	11.41	-0.05	0.96
	Tractor	1.53	-23.78	0.11	-1.58	-0.11	2.27	-0.20	3.86	-0.15	2.67
	Average	1.15	-19.33	-0.15	3.80	-0.31	6.86	-0.33	7.04	-0.07	1.40
$\Delta QP = 6$	*Harbour*	1.18	-26.25	0.19	-4.02	0.00	0.28	-0.08	2.25	-0.05	1.20
	Ice	0.80	-12.73	-0.24	5.75	-0.44	8.91	-0.46	9.12	-0.10	1.83
	Rushhour	1.07	-18.02	-0.33	9.37	-0.48	12.09	-0.46	10.87	-0.05	1.02
	Soccer	1.16	-22.24	0.04	-0.72	-0.09	2.14	-0.15	3.37	-0.06	1.33
	Station	1.23	-13.83	-0.54	11.73	-0.64	13.75	-0.60	12.28	-0.08	1.58
	Tractor	1.59	-24.62	0.16	-2.47	-0.03	0.83	-0.14	2.76	-0.12	2.05
	Average	1.17	-19.62	-0.12	3.27	-0.28	6.34	-0.31	6.77	-0.08	1.50
$\Delta QP = 8$	*Harbour*	1.09	-24.62	0.13	-2.82	-0.02	0.70	-0.09	2.36	-0.04	1.17
	Ice	0.84	-13.48	-0.24	5.45	-0.43	8.56	-0.49	9.35	-0.16	2.84
	Rushhour	1.03	-17.50	-0.35	9.87	-0.49	12.49	-0.47	11.27	-0.06	1.39
	Soccer	1.16	-22.14	0.03	-0.49	-0.10	2.42	-0.18	4.13	-0.13	2.86
	Station	1.14	-12.54	-0.61	13.61	-0.72	15.79	-0.69	14.09	-0.14	2.58
	Tractor	1.54	-24.13	0.10	-1.55	-0.08	1.71	-0.21	3.94	-0.20	3.46
	Average	1.13	-19.07	-0.16	4.01	-0.31	6.94	-0.35	7.52	-0.12	2.38

Table 3.6: BDPSNR and BDRate for the complete SVC bitstreams compared to an unmodified cascaded decoder-encoder.

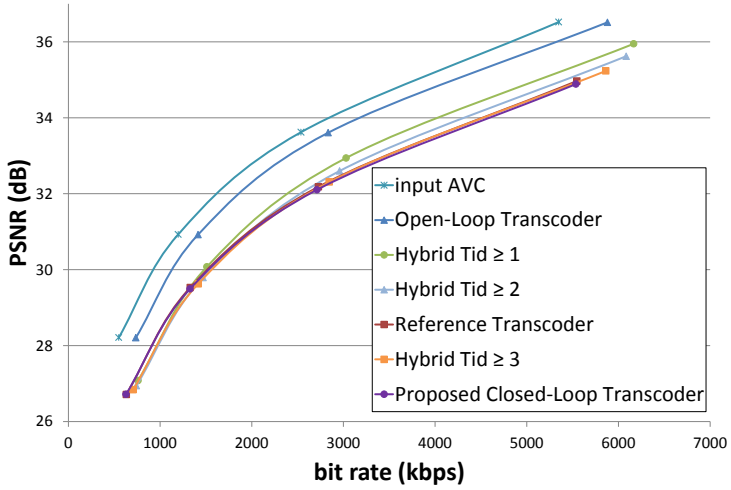


Figure 3.12: RD curves for the sequence *Harbour* with the proposed transcoding schemes for the resulting bitstream with both layers.

These conclusions should be taken into account when validating the RD performance of the complete SVC bitstream. As seen in Table 3.6, the BDRate for the open-loop scenario outperforms the other architectures. However, the lower degree of scalability, requires a higher bit rates for the base layer. Consequently, to evaluate the system, both views should be taken into considerations.

3.6 Conclusions on hybrid transcoding

An H.264/AVC input bitstream can efficiently be transcoded to an SVC bitstream with CGS while having complexity scalability at the transcoder. By combining an optimized closed-loop transcoder with an open-loop transcoder, drifting artifacts of the base layer are reduced, while the bit rate of both the base layer and the full bitstream are reduced. Both the scalability of the SVC stream and the QoE for the end user are increased. Using the optimized closed-loop transcoder, the complexity, and thus energy consumption, is only 8.48% of the original cascaded decoder-encoder scenario. This complexity can be decreased by increasing the amount of open-loop transcoded frames. When only intra predicted frames are still closed-loop transcoded, only 0.72% of the complexity is required. Meanwhile, the de-

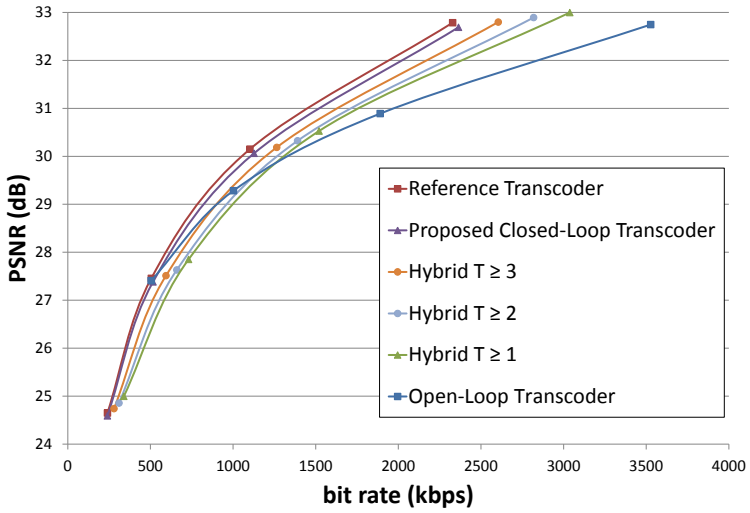


Figure 3.13: RD curves for the extracted base layer of sequence *Harbour* for the original and proposed transcoding schemes.

gree of scalability of the hybrid and closed-loop systems is much higher than open-loop architectures, resulting in a higher QoE for the end users. The proposed system can be applied for constant bit rate transcoding, while bit allocation for the layers is possible by adapting the ΔQP.

The low complexity allows to significantly reduce the energy consumption in the network. Furthermore, an adaptive system can be designed that scales transcoding architecture to the available complexity of the system. This allows to reduce the hardware investment. Indeed, not for each stream that might to be transcoded in the future, a new chip needs to be available, but by scaling the complexity, the design of the chips can be adjusted and the system scaled to the current load. This leads towards green ICT where both the hardware and energy cost are reduced.

3.7 Future Work

The hybrid transcoding architecture can be optimized and extended in different domains. Complexity optimizations are mainly to be found in the closed-loop transcoder, since this is the part of the design which still requires the highest complexity.

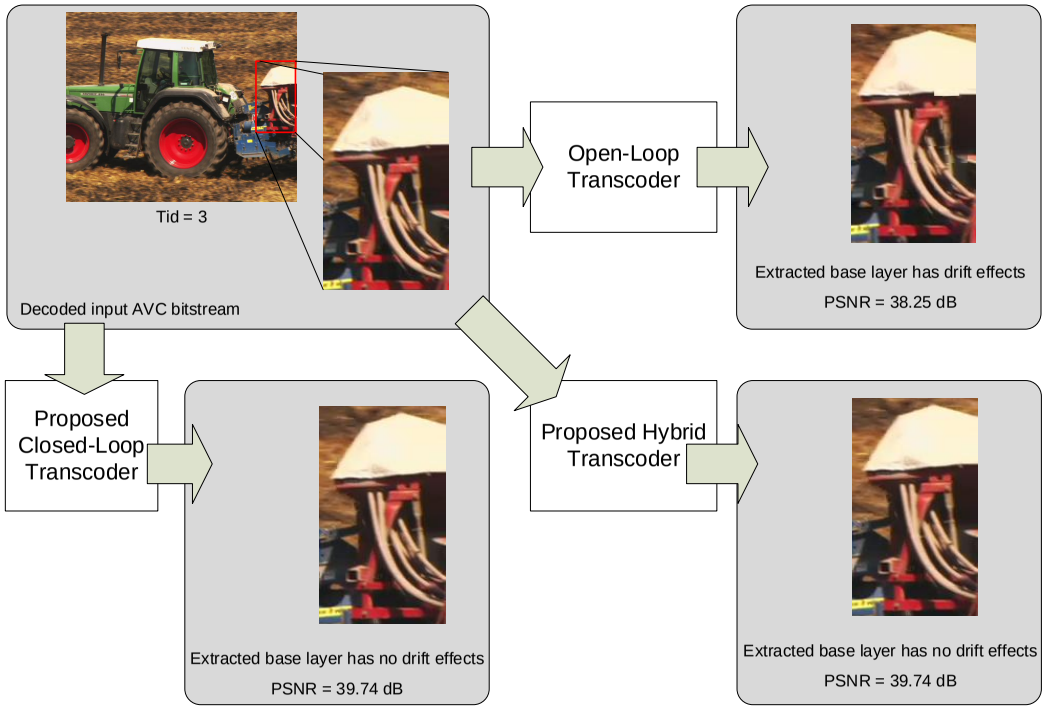


Figure 3.14: Illustration of the drift effects when open-loop transcoding is applied to sequence *Tractor* with $QP_{AVC} = 22$ and $\Delta QP = 8$.

		Open Loop		$Tid \geq 1$		$Tid \geq 2$		$Tid \geq 3$		Closed-Loop	
		BDPSNR	BDRate	BDPSNR	BDRate	BDPSNR	BDRate	BDPSNR	BDRate	BDPSNR	BDRate
$\Delta QP = 5$	*Harbour*	-1.22	44.71	-0.78	24.09	-0.64	19.45	-0.44	12.93	-0.17	4.84
	Ice	-0.49	12.72	-0.68	15.77	0.58	12.77	-0.50	10.93	-0.31	6.48
	Rushhour	-0.82	25.11	-1.04	26.25	-0.95	23.69	-0.69	16.68	-0.21	4.77
	Soccer	-0.55	17.45	-0.64	16.79	-0.52	13.43	-0.39	9.87	-0.19	4.61
	Station	0.16	-2.67	-0.73	13.00	-0.59	10.63	-0.39	7.03	-0.13	2.31
	Tractor	-1.19	30.78	-0.74	15.81	-0.76	15.99	-0.57	11.94	-0.32	6.47
	Average	-0.69	21.35	-0.77	18.62	-0.67	15.99	-0.50	11.56	-0.22	4.91
$\Delta QP = 6$	*Harbour*	-1.26	48.11	-0.94	29.72	-0.78	23.88	-0.51	15.31	-0.16	4.46
	Ice	-0.62	16.33	-0.88	19.85	0.76	16.46	-0.63	13.19	-0.32	6.47
	Rushhour	-0.87	27.18	-1.32	33.60	-1.21	30.34	-0.87	21.07	-0.22	5.10
	Soccer	-0.68	22.38	-0.86	22.46	-0.69	17.87	-0.46	11.59	-0.17	4.10
	Station	0.01	-0.09	-0.96	17.07	-0.76	13.77	-0.48	8.90	-0.14	2.43
	Tractor	-1.26	33.45	-0.94	19.81	-0.85	17.63	-0.60	12.32	-0.24	4.77
	Average	-0.78	24.59	-0.99	23.75	-0.84	19.98	-0.59	13.73	-0.21	4.55
$\Delta QP = 8$	*Harbour*	-1.26	51.42	-1.36	45.05	-1.16	37.78	-0.77	24.03	-0.18	5.17
	Ice	-0.60	15.18	-1.31	28.54	-1.16	24.51	-0.91	18.34	-0.39	7.26
	Rushhour	-0.86	26.25	-1.96	51.19	-1.79	47.94	-1.28	32.96	-0.28	6.48
	Soccer	-0.67	22.12	-1.18	32.17	-0.98	26.12	-0.65	16.51	-0.18	4.19
	Station	-0.10	2.38	-1.45	27.03	-1.15	21.99	-0.72	14.17	-0.16	2.94
	Tractor	-1.25	35.74	-0.45	31.18	-1.29	27.44	-0.90	18.24	-0.27	5.35
	Average	-0.79	25.52	-1.45	35.86	-1.26	30.97	-0.87	20.71	-0.24	5.23

Table 3.7: BDPSNR and BDRate for extracted base layer bitstreams compared to unmodified decoder-encoder shows a slight decrease in performance for higher ΔQP values.

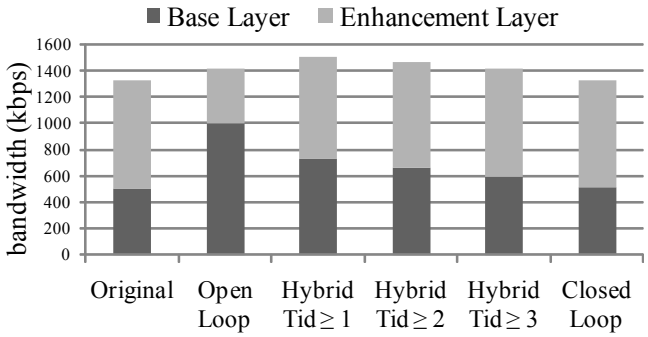


Figure 3.15: Degree of scalability comparison for closed-loop, open-loop and hybrid transcoding (sequence *Harbour* with QP_{BL}= 37 and QP_{EL}= 32).

Motion vector prediction optimization can be done in the base layer for *MODE_16×16* to reduce the motion vector search complexity in the closed-loop transcoding architecture. As indicated in Figure 3.4, *MODE_16×16* is always evaluated. When *MODE_16×16* has not been selected in the input H.264/AVC bitstream, a normal motion estimation is performed with a search window of 16 pixels around a predicted motion vector. This predicted motion vector is derived from the motion vectors of surrounding (sub-)macroblocks. However, the average motion vector of the H.264/AVC (sub-)partitions can be used as an initial motion vector if the variance of the H.264/AVC input motion vectors is acceptable.

Mode decision optimization can be achieved in different ways. If the motion vectors of the (sub-)partitions in the H.264/AVC input stream are not well correlated (and thus have a high variance and no motion vector prediction optimization can be achieved) it might be concluded that an unpartitioned mode will most likely not achieve a better RD compared to a partitioned mode. Consequently, for partitioned modes in the H.264/AVC input bitstream, *MODE_16×16* should only be evaluated with a reduced search window, or not be evaluated at all.

Furthermore, if $SKIP$ has been selected in the H.264/AVC input bitstream, it is unlikely that any other mode will be selected for the base layer. However, due to the changed reference frames of the base layer, a stop criterion can be placed to make sure this will be the case and residual information is not required. So both $SKIP$ and $MODE\_Direct$ can be evaluated for the base layer, and only when $MODE\_Direct$ yields a better RD cost compared to $SKIP$, also *MODE_16×16* can be evaluated. In the latter case, the predicted motion vector from the base layer can be used as an initial motion vector for *MODE_16×16*.

Improving the architecture can lead to a hybrid transcoder where the open- and closed-loop transcoders are switched on a per macroblock basis. Therefore, macroblocks that are referenced by other macroblocks will be closed-loop transcoded, while all other macroblocks will be open-loop transcoded. However, this requires buffering to analyze the future macroblocks and frames.
An even lower complexity could be achieved by also taking into account the residual energy of the H.264/AVC input macroblocks. When this energy is low, there is a lower probability for drift errors. Consequently, such macroblocks can be open-loop transcoded by default, independently whether or not these macroblocks are being referenced. Furthermore the mode decision optimization can be implemented more drastically, such that $SKIP$ macroblocks in H.264/AVC are by default open-loop transcoded.
These macroblock level changes can be applied to all frames, or could only be applied to frames with $Tid = 1$, while frames with $Tid > 1$ are open-loop transcoded by default. This will bring the complexity of the hybrid transcoder down to around 0.5% of the cascaded decoder-encoder configuration. Allowing 200 bitstreams to be transcoded simultaneously at the cost of one with a non-optimized transcoder.
Transcoding to MGS has not been investigated yet. This might be easily achieved by using MGS vectors, which allows to define the number of residual values for each layer. Consequently, such a system can open-loop transcode the whole bitstream, with an even lower complexity, since no entropy decoding, requantization and entropy encoding is required. Again, drift artifacts are introduced for the base layer, but research should investigate the impact of those. No closed-loop transcoding optimizations can be performed[6] since the enhancement layer has the same macroblock partition as the base layer, in fact the enhancement layer does not transport relevant syntactical information for the macroblocks.
Cross-layer optimizations for SVC have been proposed in the past [15]. Given the current developments, those can be extended towards HEVC. In this chapter, it was confirmed that selecting a sub-optimal mode for the base layer, slightly reduces the scalability of the base layer, although the total bit rate would be lower. Since the mode decision uses the optimal RD for a macroblock in the current layer, it is not aware of the impact for the prediction of other layers. This yields RD towards a local minimum, while the global minimum might not be reached. Consequently, the selected mode of the enhancement layer might require more bits to compensate for a less optimal prediction than the saved bits in the base layer.

[6] Other than decoding and encoding with the same partition and motion vector to reduce the drift effects.

Therefore, Equation 2.1 could be extended to Equation 3.1. Here, the RD is optimized by minimizing the RD cost J while taking into account the distortion and rate of each layer ($\boldsymbol{p}_0$ and $\boldsymbol{p}_1$), where D_0 and D_1 are the distortion for the base and enhancement layer respectively, and R_0 and R_1 are the rate of the base and enhancement layer respectively. It should be noted that the distortion of the enhancement layer should consider the distortion of the base layer, otherwise the minimal cost can be achieved by reducing the base layer quality, ultimately eliminating the base layer. Furthermore, a weighting factor (w) should be considered, which balances the impact of both layers and can define the rate allocation for each layer. Moreover, also a different λ might be used for each layer (λ_0 for the base layer and λ_1 for the enhancement layer), to adjust for the impact of the rates because of to lower rates in the enhancement layer due to better predictions[7].

$$\begin{aligned} J = \min_{\{\boldsymbol{p}_0, \boldsymbol{p}_1 | \boldsymbol{p}_0\}} (1-w) \quad & .\big(D_0(\boldsymbol{p}_0) + \lambda_0 . R_0(\boldsymbol{p}_0)\big) + \\ w \quad & .\Big(D_1(\boldsymbol{p}_1|\boldsymbol{p}_0) + \lambda_1 . \big(R_0(\boldsymbol{p}_0) + R_1(\boldsymbol{p}_1|\boldsymbol{p}_0)\big)\Big). \end{aligned} \tag{3.1}$$

This cross-layer optimization can be investigated for the future extensions for HEVC. If this approach seems to be appropriate, the scalable extension for HEVC can be designed according to the found conclusions.

[7]Note that for H.264/AVC λ has been experimentally defined by $\lambda = 0.85 \times 2^{\frac{QP-12}{3}}$ [16]. However, this experiment was done for single layer H.264/AVC. Since the enhancement layers of SVC yield different statistical properties due to the improved predictions, the Lagrangian multiplier λ of the RD cost formula might need to be re-evaluated. Furthermore, for HEVC λ is left unchanged. Consequently, this also changes the statistical properties of the rate and distortion.

The research described in this chapter resulted to the following publications.

- Rosario Garrido-Cantos, Jan De Cock, José Luis Martínez, Sebastiaan Van Leuven, Pedro Cuenca, Antonio Garrido, and Rik Van de Walle, *"Video Adaptation for Mobile Digital Television"*, in Proc. of the IEEE Third Joint IFIP Wireless and Mobile Networking Conference (WMNC), Oct. 2010, Hungary.

- Glenn Van Wallendael, Sebastiaan Van Leuven, Rosario Garrido-Canto, Jan De Cock, José Luis Martínez, Peter Lambert, and Rik Van de Walle, *"Fast H.264/AVC-to-SVC Transcoding in a Mobile Television Environment"*, in Proc. of the 6th International Mobile Multimedia Communications Conference, Aug. 2010, Portugal.

- Rosario Garrido-Cantos, Jan De Cock, José Luis Martínez, Sebastiaan Van Leuven, Pedro Cuenca, Antonio Garrido, and Rik Van de Walle, *"On the Impact of the GOP Size in an H.264/AVC-to-SVC Transcoder with Temporal Scalability"*, in Proc. of the 8th international conference on advances in mobile computing and multimedia (MoMM), Nov. 2010, France.

- Sebastiaan Van Leuven, Jan De Cock, Glenn Van Wallendael, Rik Van de Walle, Rosario Garrido-Cantos, José Luis Martínez, and Pedro Cuenca, *"A Low-Complexity Closed-Loop H.264/AVC to Quality-Scalable SVC Transcoder"*, in Proc. of the 17th IEEE International Conference on Digital Signal Processing (DSP), July 2011, Greece.

- Sebastiaan Van Leuven, Jan De Cock, Glenn Van Wallendael, Rik Van de Walle, Rosario Garrido-Cantos, José Luis Martínez, and Pedro Cuenca, *"Combining Open- and Closed-Loop Architectures for H.264/AVC-TO-SVC Transcoding"*, in Proc. of the IEEE International Conference on Image Processing (ICIP), pp. 1661-1664, Sept. 2011, Belgium.

- Rosario Garrido-Cantos, Jan De Cock, José Luis Martínez, Sebastiaan Van Leuven, and Pedro Cuenca, *"Motion-Based Temporal Transcoding from H.264/AVC-to-SVC in Baseline Profile"*, in IEEE Transactions on Consumer Electronics, Vol. 57, nr. 1, pp 239-246, Feb. 2011.

- Rosario Garrido-Cantos, Jan De Cock, Sebastiaan Van Leuven, Pedro Cuenca, Antonio Garrido, and Rik Van de Walle, *"Fast Mode Decision Algorithm for H.264/AVC-to-SVC Transcoding with Temporal*

Scalability"', in Proc. of 18th International Conference on Advances in Multimedia Modeling, MMM 2012, Jan. 2012, Austria.

- Rosario Garrido-Cantos, Jan De Cock, Sebastiaan Van Leuven, Pedro Cuenca, Antonio Garrido, and Rik Van de Walle, *"Fast Mode Decision Algorithm for H.264/AVC-to-SVC Transcoding with Temporal Scalability"*, in Lecture Notes in Computer Science, Advances in Multimedia Modeling, Vol. 7131, pp 585-596, 2012.

- Sebastiaan Van Leuven, Jan De Cock, Glenn Van Wallendael, Rosario Garrido-Cantos, and Rik Van de Walle, *"Complexity Scalable H.264/AVC-to-SVC Transcoding"* in Proc. of the 2013 IEEE International Conference on Consumer Electronics (ICCE), pp. 328-329 , Jan. 2013, USA.

- Rosario Garrido-Cantos, Jan De Cock, José Luis Martínez, Sebastiaan Van Leuven, Pedro Cuenca, and Antonio Garrido, *"Low complexity transcoding algorithm from H.264/AVC-to-SVC using Data Mining"*, EURASIP Journal on Advanced Signal Processing. Accepted for future publication.

- Sebastiaan Van Leuven, Jan De Cock, Glenn Van Wallendael, Rosario Garrido-Cantos, and Rik Van de Walle, *"A Hybrid H.264/AVC-to-SVC Transcoder with Complexity Scalability"*, submitted to IEEE Transactions on Consumer Electronics.

References

[1] J.-S. Lee, F. De Simone, N. Ramzan, Z. Zhao, E. Kurutepe, T. Sikora, J. Ostermann, E. Izquierdo, and T. Ebrahimi. Subjective evaluation of scalable video coding for content distribution. In *Proceedings of ACM Multimedia*, pages 65–72, Oct. 2010.

[2] H. Liu, Y.-K. Wang, and H. Li. A comparison between SVC and transcoding. *IEEE Transactions on Consumer Electronics*, 54(3):1439–1446, Aug. 2008.

[3] A. Dziri, A. Diallo, M. Kieffer, and P. Duhamel. P-picture based H.264 AVC to H.264 SVC temporal transcoding. In *Proceedings of International Wireless Communications and Mobile Computing Conference (IWCMC).*, pages 425–430, Aug. 2008.

[4] R. Garrido-Cantos, J. De Cock, J.L. Martínez, S. Van Leuven, P. Cuenca, A. Garrido, and R. Van de Walle. Video adaptation for mobile digital television. In *Proceedings of Third Joint IFIP Wireless and Mobile Networking Conference (WMNC)*, pages 1–6, 2010.

[5] R. Garrido-Cantos, J. De Cock, J.L. Martínez, S. Van Leuven, P. Cuenca, A. Garrido, and R. Van de Walle. On the impact of the GOP size in an H.264/AVC-to-SVC transcoder with temporal scalability. In *Proceedings of 8th international conference on advances in mobile computing and multimedia (MoMM)*, pages 1–8. ACM, 2010.

[6] R. Garrido-Cantos, J. De Cock, J. L. Martínez, S. Van Leuven, P. Cuenca, A. Garrido, and R. Van de Walle. An H.264/AVC to SVC Temporal Transcoder in baseline profile : digest of technical papers. In *Proceedings of IEEE International Conference on Consumer Electronics (ICCE)*, pages 339–340. IEEE, Jan. 2011.

[7] R. Garrido-Cantos, J. De Cock, J. L. Martínez, S. Van Leuven, and P. Cuenca. Motion-based temporal transcoding from H.264/AVC-to-SVC in baseline profile. *IEEE Transactions on Consumer Electronics*, 57(1):239–246, 2011.

[8] R. Garrido-Cantos, J. De Cock, J.L. Martínez, S. Van Leuven, P. Cuenca, A. Garrido, and R. Van de Walle. Fast mode decision algorithm for h.264/avc-to-svc transcoding with temporal scalability. In *Advances in Multimedia Modeling*, volume 7131 of *Lecture Notes in Computer Science*, pages 585–596. Springer Berlin / Heidelberg, 2012.

[9] R. Sachdeva, S. Johar, and E. M. Piccinelli. Adding SVC spatial scalability to existing H.264/AVC video. In *8th IEEE/ACIS International Conference on Computer and Information Science*, pages 1090–1095, June 2009.

[10] J. De Cock, S. Notebaert, P. Lambert, and R. Van de Walle. Architectures for fast transcoding of H.264/AVC to quality-scalable SVC streams. *IEEE Transactions on Multimedia*, 11(7):1209–1224, July 2009.

[11] G. Van Wallendael, S. Van Leuven, R. Garrido-Cantos, J. De Cock, J. L. Martínez, P. Lambert, P. Cuenca, and R. Van de Walle. Fast H.264/AVC-to-SVC transcoding in a mobile television environment. In *6th International ICST Mobile Multimedia Communications Conference*, page 12. ICST, 2010.

[12] Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG. Joint Scalable Video Model. Technical report, MPEG and ITU-T, Jan. 2010.

[13] R. Garrido-Cantos, J.-L. Martínez, P. Cuenca, and A. Garrido. An approach for an AVC to SVC transcoder with temporal scalability. In *Hybrid Artificial Intelligence Systems*, volume 6077 of *Lecture Notes in Computer Science*, pages 225–232. Springer Berlin Heidelberg, 2010.

[14] H. Li, Z. Li, C. Wen, and L.-P. Chau. Fast mode decision for spatial scalable video coding. In *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS)*, page 4, May 2006.

[15] H. Schwarz and T. Wiegand. R-D optimized multi-layer encoder control for SVC. In *Proceedings of IEEE International Conference on Image Processing (ICIP)*, pages 281–284, Oct. 2007.

[16] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan. Rate-constrained coder control and comparison of video coding standards. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7):688–703, July 2003.

4 Hybrid 3D video coding

4.1 Rationale and related work

The increasing availability of high-quality 3D content and the enhanced functionality of 3D systems are posing challenges to the compression and transmission of 3D video signals. Currently, 3D content is widely available in digital cinema environments. An increasing amount of consumers have a 3D compatible television at their home and Blu-ray 3D is gaining momentum. Current display technologies [1], such as stereoscopic displays, require the end-user to wear glasses, either passive or active. With both technologies, two slightly different viewpoints are projected by the television screen, and in combination with the glasses, only one view is perceived by each eye, resulting in a 3D perception of the scene. Different manufacturers apply different technologies, although two main technologies can be distinguished: passive and active glasses.

4.1.1 Stereoscopic 3D display technologies

Passive glasses have polarized filters and require the display to apply a circular polarization of the projected light. Therefore, filters are placed on top of the display which polarizes the light of each pixel. However, since two filters are required, one for each view, all pixels have to be divided between both filters. Consequently, only half the resolution in one dimension can be used for each view, either half the width or height of the image is available

for each view. So, in 3D only half of the available pixels are perceived by each eye.
Active glasses on the other hand, allow to perceive a full HD resolution in 3D. The display has a double frame rate, and alternatively projects a frame for each view. The glasses let the light from the display through for the corresponding eye of the projected view, while the light is blocked for the other eye. To do so, the glasses use LCD technology to darken the glass in front of the eye in order to block the light. The display communicates with the glasses to synchronize the shutters. Therefore, each eye perceives only the corresponding view, while the other view is blocked. However, this system has some drawbacks: the users have to wear heavier glasses; the glasses have to be charged; the communication with the TV is mainly done using infra-red, which requires a line of sight[1]; and the display will be more expensive due to the refresh frame rate.
Glasses appear to be a bothersome threshold for many end-users. Therefore, in the near future, the market introduction of so called autostereoscopic displays is to be expected.

4.1.2 Autostereoscopic 3D display technologies

Autostereoscopic 3D displays allow to perceive a 3D scene without glasses. The display projects multiple views[2], which are blocked by parallax barriers or redirected by lenticular lenses or a lens array. Using parallax barriers, the light of the pixels is blocked for certain positions, so the user only sees on each eye the pixels corresponding to that view. For lenticular lenses, no light is blocked, but lenses are placed over the display which redirect the light in such a way that different pixels are visible from different positions[3]. The lenticular lenses have the same lens properties for each pixel in a given column. To increase the quality of the perceived image, the a lens array is placed over the display. This allows to modify the lens properties (i.e., the diffraction index of the light) for each pixel independently. The result is that for a given position, out of the total set of views, only two views will be captured by the viewer's eye. When the position of the end-user is changed, different views will reach the eye, creating a more realistic 3D experience and giving the impression of free viewpoint television [2]. A schematical illustration of an autostereoscopic display with three views is

[1] This line of sight can be easily blocked by people passing, or when looking away from the screen, the glasses will switch off. In both cases, glasses have to re-synchronize and the 3D experience is interrupted.

[2] Displays with 28 views are currently in a protoype stage.

[3] Similar to the system used for 3D paper prints.

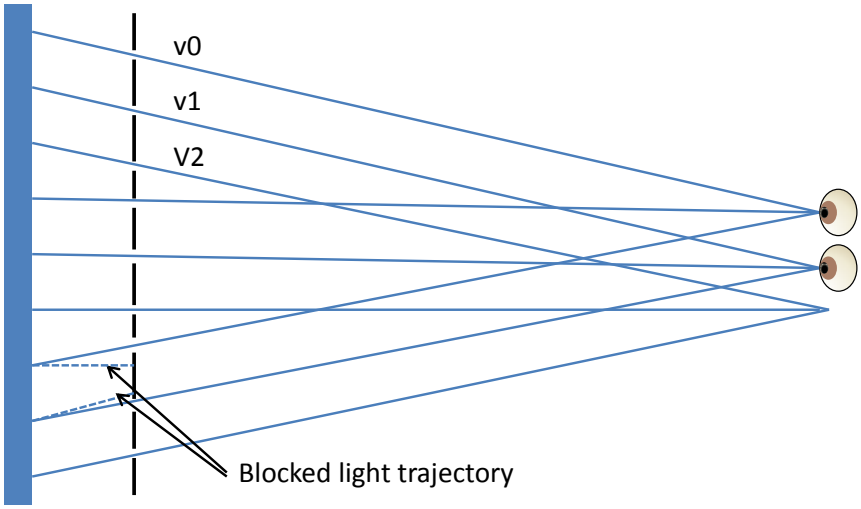


Figure 4.1: Schematical overview of a stereoscopic display with parallax barriers for three views. For simplicity of the figure, only the light paths for the center cone are shown.

given in Figure 4.1 for parallax barriers, while Figure 4.2 shows lenticular lenses and Figure 4.3 shows a lens array for two rows.
The light passed through the parallax barriers and the light diffracted by the (lenticular) lenses of each pixel from the same view converges in one point. The practical implementation of these lenticular lenses is more complicated. The light of each lens is diffracted to multiple spatial positions, resulting in so-called *'cones'*. Figure 4.4 illustrates this concept. Because of these cones, the same 3D scene can be perceived by multiple users.
To allow autostereoscopic systems to work properly, all projected viewpoints have to be available at the display device. Since transmitting all these views is impractical, only a subset of the projected views are transmitted. The other required views are generated by the display. This is done using view synthesis [3–5], which calculates a large number of intermediate viewpoints. The view synthesis uses the transmitted views and corresponding depth maps[4] to recreate a virtual 3D scene. The intermediate views are constructed by calculating the resulting image in this 3D scene. Typically three views and corresponding depth information are required to obtain acceptable results for the view synthesis [6]. Therefore, an encoding and transmission system for multiple views including depth

[4]The depth maps can be transmitted, but might also be calculated at the display device.

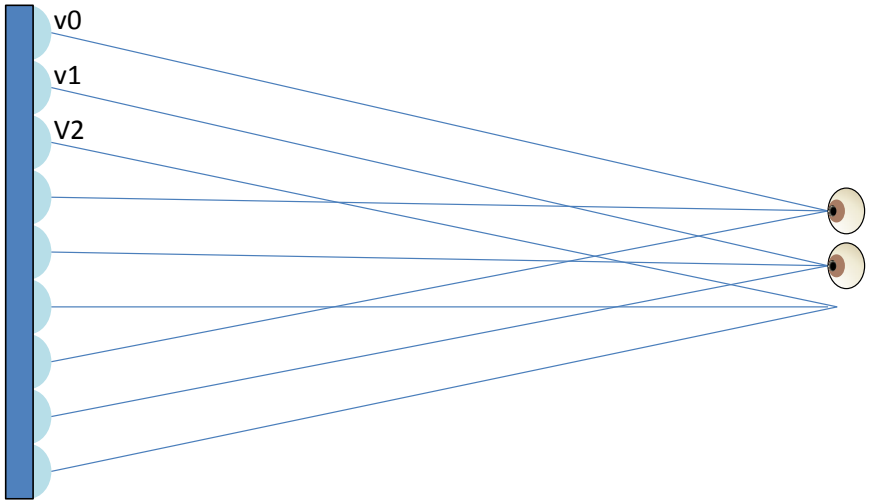


Figure 4.2: Schematical overview of a stereoscopic display with lenticular lenses for three views. For simplicity of the figure, only the light paths for the center cone are shown.

has to be developed. However, two main issues arise. Firstly, compatibility towards existing systems that support mono and stereo video is highly recommended [6]. Secondly, to limit the load on the network, the additional bit rate required to transmit 3D video should be as low as possible.

Due to the availability of monoscopic (2D) and stereoscopic technologies and the near-future availability of autostereoscopic displays, appropriate coding solutions for 3DTV have to be considered which can support a wide range of 3D functionality, and which can preferably coexist in a compatible way with existing systems. Furthermore, currently different ad-hoc solutions and standards are available to represent and encode 3D video. If no measures are taken, a proliferation of different technologies is a fact.

4.1.3 3D coding technologies

For 2D video compression, H.264/AVC is currently widely used, while for stereoscopic video the multiview extension of H.264/AVC (Multiview Video Coding (MVC) [7]) is gaining interest, e.g., for Blu-ray 3D where the Stereo High Profile of MVC is supported [8]. Guaranteeing forward

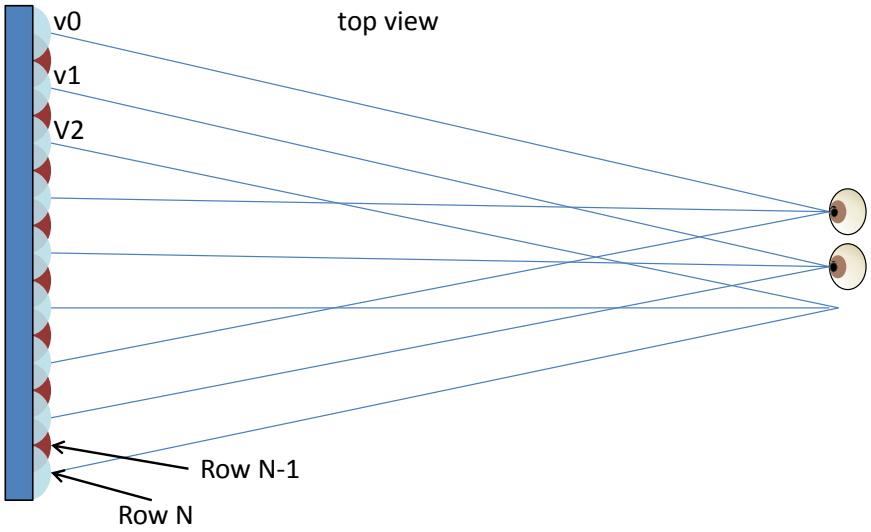


Figure 4.3: Schematical overview of a stereoscopic display with a lens array for three views. For simplicity of the figure, only the light paths for the center cone are shown.

compatibility[5] for a new 3D video standard allows current network and decoding equipment to handle the 3D video bitstream and create a monoscopic output. Therefore, the functionality of the current 2D transmission and storage systems can be incorporated in a future 3D standard. Existing systems are able to receive a basic sub-stream and continue operations, while upgraded systems can benefit from additional 3D functionality. This allows operators to improve the service they deliver, with a limited cost and guaranteed interoperability for existing systems.
Another important aspect of 3D video systems is compression efficiency. As indicated in Chapter 1, prognosis for bandwidth usage in the near-future indicate a huge amount of traffic. Limiting as much as possible the bit rates for video compression will help to handle all data and manage to keep the cost per bit low. The bit rate for simulcast (sending all texture and depth views as independently encoded views) will be unacceptably high.

[5]Forward compatibility allows the existing H.264/AVC compliant devices to (partly) decode a bitstream of the new standard. Where backward compatibility allows a new standard to be fully compliant with the existing H.264/AVC. H.264/AVC devices can decode the MVC center view, allowing these devices to be forward compatible. Meanwhile, new MVC devices allow to decode an H.264/AVC bitstream, making these devices backward compatible.

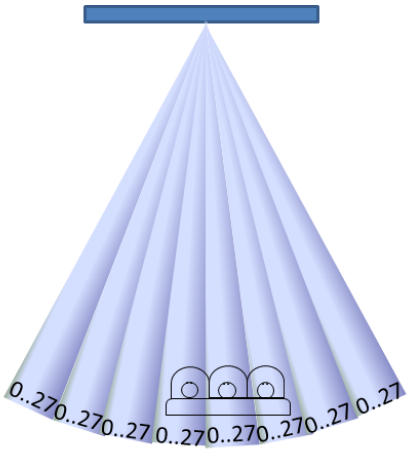


Figure 4.4: Schematical overview of a stereoscopic display with 28 views, where the repetitive cones are indicated. In each of the cones the 3D scene can be perceived.

Therefore exploiting redundancy between base view streams and additional 3D video data (such as dependent views or depth maps) is one possibility to lower the overall bit rate. In MVC, inter-view prediction is applied, which allows a previously encoded view to serve as a predictor for other views. In turn, the data transmitted with H.264/AVC or MVC can be reused for predicting the additional 3D video data, yielding a lower total bit rate in the network for full-fledged 3D coding systems. Designing efficient prediction schemes which include this additional data is the goal within JCT-3V (Joint Collaborative Team on 3D Video).

With the advent of HEVC [9], which has been standardized within the Joint Collaborative Team on Video Coding of MPEG and VCEG (JCT-VC), the successor of JVT, the bit rate can be roughly halved [10] for the same perceptual quality compared to H.264/AVC. As a result, it seems natural to consider HEVC for 3D coding systems, either for fully HEVC-based systems, or to supplement MVC or H.264/AVC based systems.

A brief overview of multi-view video coding structures is given next. These technologies have either been standardized, or are currently under investigation by MPEG and VCEG for 3D video coding. In the remainder of this chapter, these structures will be used as building blocks or as a reference to compare the performance of the proposed hybrid architecture. Therefore, the remainder of this section will only discuss 3D solutions offered as an extension of the H.264/AVC and HEVC standards.

Frame compatible H.264/AVC coding

Frame compatible coding formats for 3D video (also referred to as MPEG Frame-Compatible (MFC)) combine stereoscopic views such that the multiplexed HD signal can be reconstructed by legacy 2D decoders, and can be interpreted as a 3D signal by display devices. An overview of such approach is shown in Figure 4.5. The advantage is that regular devices can still be used and the frame compatible 3D video is handled as regular video. One approach for frame compatible coding is temporal interleaving. A left and right image are encoded alternatively. Doing so, either the frame rate of the resulting video sequence will be doubled, or only half of the frame rate for each view will be used. Other common approaches include spatial interleaving of the stereoscopic views into a single 2D image. At the encoder side, both stereo views are sub-sampled, after which they are combined into a single (2D) video signal according to a predefined arrangement[6]. The resulting signal is coded and transmitted to the decoder. After decoding, demultiplexing and upsampling at the receiver side, the stereoscopic signal can be displayed. A number of common frame packing arrangements are used to transmit a stereo pair: horizontal side-by-side; vertical side-by-side; checkerboard pattern; column interleaved and row interleaved [11]. The latter two interleave the columns and rows of each (half-resolution) view into the new frame compatible format. In H.264/AVC, the frame compatible format can be signaled using the Frame Packing Arrangement Supplemental Enhancement Information (SEI) message [12]. An SEI message contains additional information to hint the decoder. A broad range of SEI messages are available such as reference buffer information, film grain characteristics, and (post-) processing information. All these messages serve to ameliorate the viewing experience. Although, SEI messages are defined by the standard, they are a non-normative part such that a decoder is not required to decode these messages, or to take action upon them after decoding. Using the frame packing arrangement SEI message, the decoder correctly interprets the used frame compatible format, and can rearrange the incoming video to a suitable 3D representation. If this SEI message is not interpreted by the decoder the video is decoded in a regular way and a 2D display will visualize the content. An example of a resulting image is shown in Figure 4.5. When the SEI message is not available (or decodable), a frame compatible representation is visible in 2D. This yields an unpleasant viewing experience. Note that when using an interleaved arrangement, the packed frame content will be even more unpleasant to watch in 2D.

[6]Since the sub-sampling is non-normative, different sub-sampling techniques can be used.

When side-by-side frame compatible techniques are used, for each view the number of pixels in one dimension has to be halved to fit the HD image. Consequently, no full-HD 3D video will be reconstructed. These issues can be solved by applying full resolution enhancement layers on top of a frame compatible base view [13].
Frame compatible techniques are inherently less efficient compared to multiview video techniques at the same spatial resolutions. This is because multiview video coding allows a decoded view to be used as a predictor for the following views. For frame compatible techniques, only motion vectors pointing to regions in previously encoded frames can be used. Consequently, the same frame in the other view cannot be referenced. Furthermore, the motion vector difference between the motion vector and the predicted motion vector (based on neighboring motion vectors) might be large because the motion vector might be pointing to the other view. The column and row interleaved techniques reduce this motion vector cost, although it is assumed that performance will decrease because of less optimal predictions between adjacent pixels.

Multiview extension of H.264/AVC (MVC)

MVC was mainly developed for efficient compression of different viewpoints from the same scene, by exploiting correlation between the different views (inter-view prediction). This inter-view prediction mechanism is similar to how single-view compression takes advantage of temporal correlation between successive frames [14]. In a monoscopic block-based encoder like H.264/AVC, temporal correlation is reduced by a motion compensation process, while for MVC the same process is applied with a neighboring view. This process is then referred to as disparity compensation. An example multiview configuration with three views is given in Figure 4.6. In this figure, horizontal arrows indicate temporal prediction within a view. Vertical arrows indicate inter-view prediction between different views.
As an extension of H.264/AVC, MVC provides forward compatibility for its monoscopic variant. The base view within MVC is always encoded independently from the other views, and can be extracted and decoded by legacy H.264/AVC decoders [7]. This allows to roll-out 3D video without requiring to upgrade all network infrastructure or end user devices instantly, but a gradual adoption of MVC is possible.

3D coding extension of H.264/AVC

One of the tracks in MPEG 3D Video in response to the recent Call for Proposals [15] has been dedicated to an extension of H.264/AVC which offers

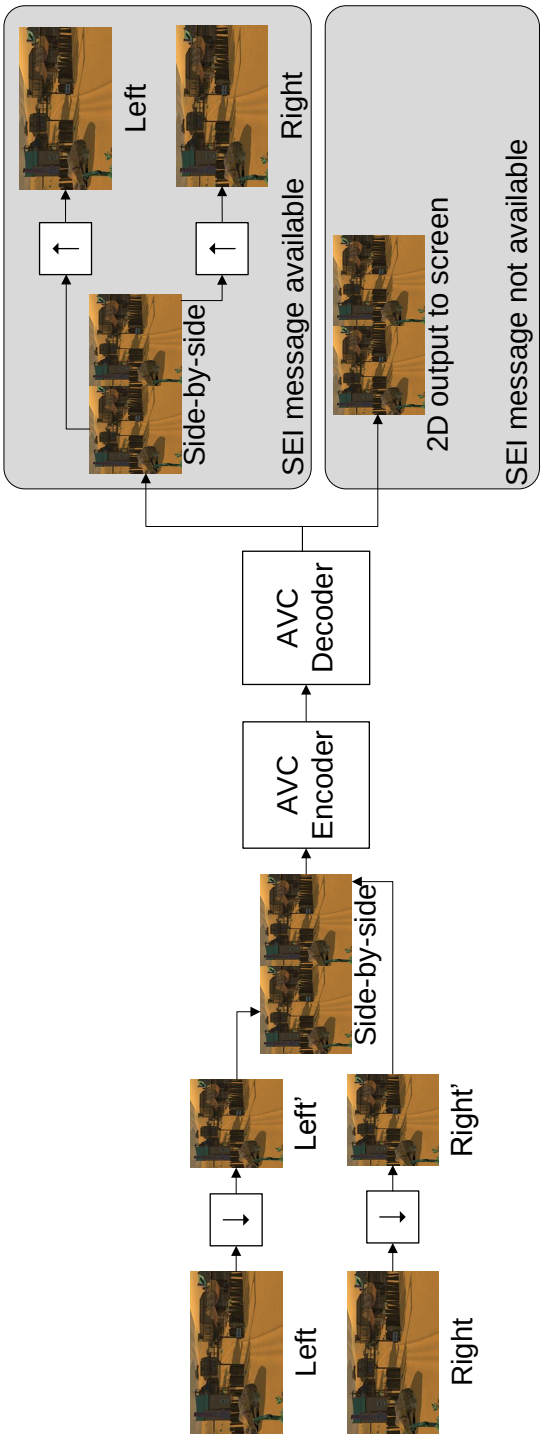


Figure 4.5: Overview of a side-by-side frame packing arrangement for stereoscopic 3D. The 2D output to the screen will be visual in side-by-side arrangement when the SEI message is unavailabe or not decodable.

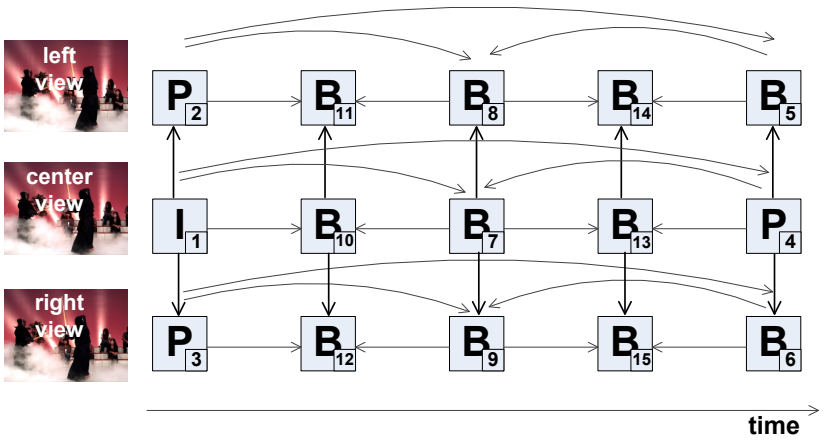


Figure 4.6: Coding structure of a multiview coding scenario (applicable to both MVC or Multiview HEVC) combining three related views. The encoding order is indicated with a number. Arrows visualize inter-frame or inter-view prediction.

superior compression efficiency over MVC, and which includes the possibility to incorporate depth maps in the coded streams [16]. This track within MPEG 3D Video is forward compatible with single-view H.264/AVC, and currently outperforms MVC-based coding (video-plus-depth) by roughly 30%. This track is scheduled to result in a Final Draft Amendment by mid 2013, and is being tested with the AVC based 3D video Test Model (ATM) reference implementation [17]. 3DV-ATM is based on the Joint Model (JM) reference software for H.264/AVC and supports additional tools for improved coding efficiency. Within ATM, two profiles are currently being investigated.

The ATM-High Profile (ATM-HP) is MVC compatible and introduces high-level adaptations and joint texture-depth encoding. This joint texture-depth encoding allows to encode both texture and depth in a single bitstream, since for MVC no signaling for depth information has been provided. Additionally, camera parameters to interpret the depth information are encoded too. However, no low-level tools are incorporated. This can be seen in Figure 4.7, where the depth views are encoded similarly but independent of the texture views since no depth information is required for the encoding

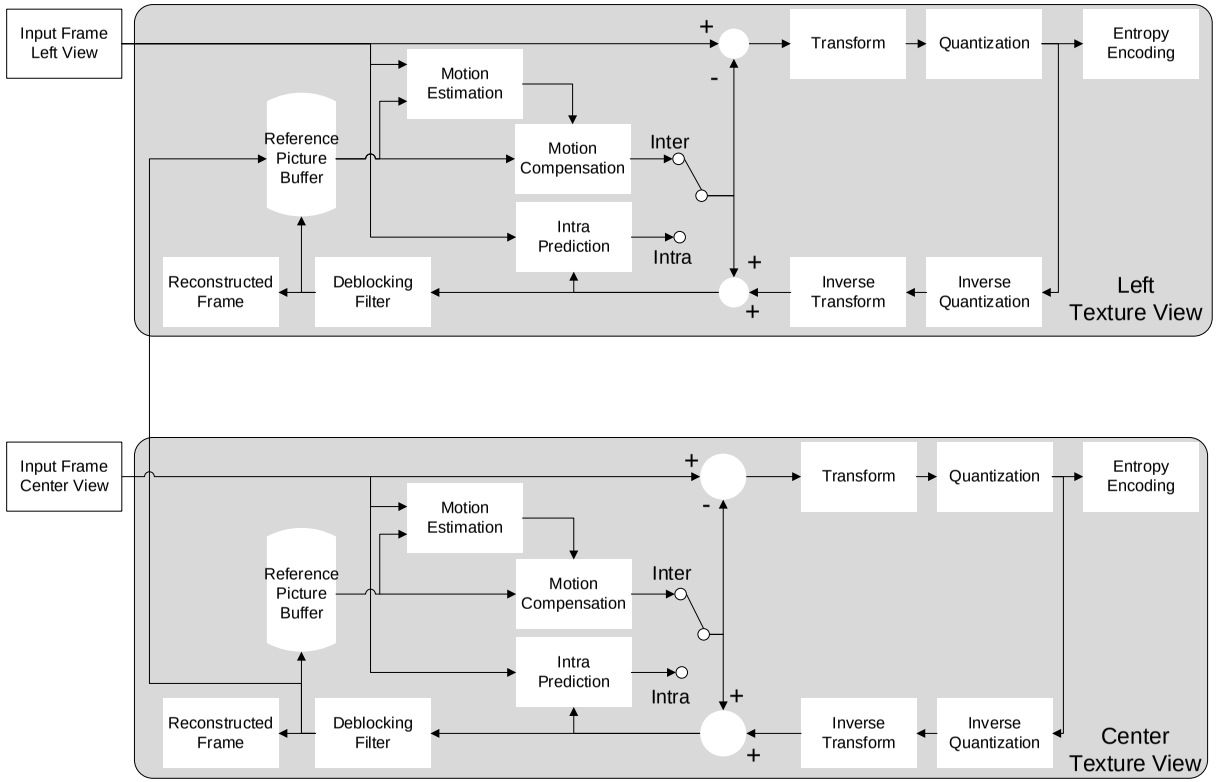


Figure 4.7: Schematic block structure of an ATM-HP encoder without applying low-level changes to the basic encoder design. Only one loop from the decoded base view to the reference picture buffer of the other view is provided.

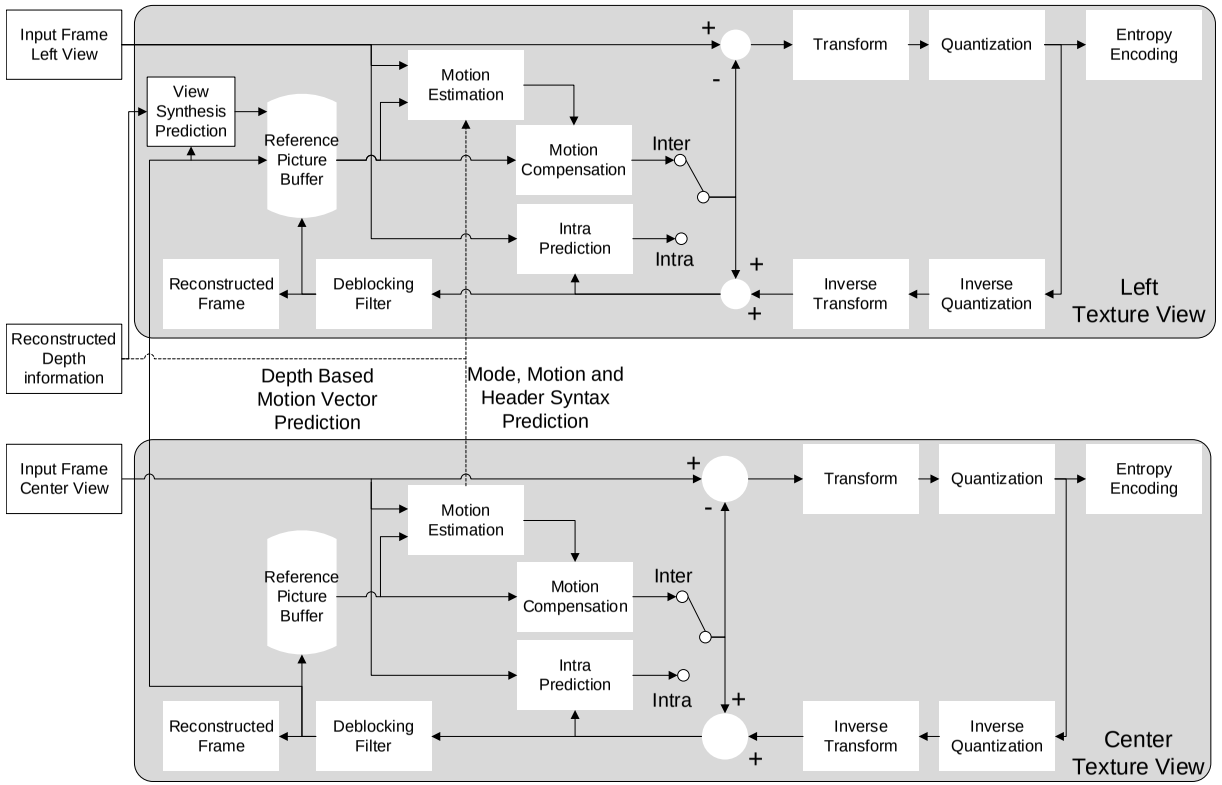


Figure 4.8: Schematic block structure of an ATM encoder applying low-level changes to the basic encoder design.

process. Note that next to temporal prediciton, only inter-view predictions in the pixel domain are possible, yielding an MVC compatible design.
The ATM-Enhanced High Profile (ATM-EHP) on the other hand is only H.264/AVC compatible, but introduces more low-level adaptations such as in-loop joint inter-view depth filtering, motion prediction from texture to depth, prediction slice header syntax elements, depth based motion vector prediction. For the texture views, in-loop view synthesis-based inter-view prediction and depth-based motion vector prediction have been adopted in the reference software. This track is currently in a working draft state [18]. In Section 4.3 the 3DV-ATM implementation is used as a reference point for the proposed hybrid 3D video coding architecture.
Figure 4.8 shows a basic ATM-EHP encoding scheme. Compared to the ATM-HP encoder in Figure 4.7 also depth information, syntactical information and synthesised views can be used as a predictor for the side views. The ATM-EHP profile is capable of exploiting additional tools. To reduce the complexity of the drawing, only texture views have been represented. However, to fully exploit the benefits of ATM-EHP, the texture-depth coding order is important. When all depths are encoded before the side texture views, the texture side views can be predicted more efficiently due to the available depth information (*Reconstructed Depth information* in Figure 4.8). This depth information can be used to perform view synthesis prediction, which is based on a synthesized intermediate view based on the available texture and depth information. The resulting (extrapolated) synthesized texture view is used as an additional prediction picture.

Multiview video coding extension of HEVC (MVHEVC)

HEVC, which is standardized in 2013, has shown to provide significant objective and subjective quality gains over H.264/AVC. Similar to MVC, a multiview extension of HEVC can be created by flexibly arranging reference picture lists without including new coding tools [19]. This allows to use previously encoded views as a prediction, which only have to be stored in as a reference pictures. Therefore, the whole disparity compensation process is identical to motion compensation, and no additional tools have to be incorporated
In the proposed hybrid coding solution, single-view HEVC has been adapted to a multi-view variation (MVHEVC) such that it matches the features of MVC (when compared to H.264/AVC). Consequently, the MVC prediction structure as described in Figure 4.6 is still applicable for MVHEVC.
The most important realization for MVC compared to H.264/AVC was enabling inter-view prediction from an earlier decoded view. In MVHEVC,

this concept can be enabled similarly to MVC by the possibility to include pictures from an earlier decoded view in the reference picture lists. The adaptation required to enable this feature can be found in the Reference Parameter Set (RPS) signaling of HEVC [20][7]. In the RPS, reference frames are indicated with a Picture Order Count (POC) difference relative to the current POC. To access another view at the same time instance, a POC difference of zero is enabled in the RPS. Because different views at the same time instance can occur, an additional view index must be signaled. In the proposed approach, only the closest view is used for inter-view prediction.
inter-view prediction was implemented adaptively at Prediction Unit (PU) level[8]. More specifically, each PU indicates the chosen reference frame by means of an index in the reference picture lists. In these lists one or several previously encoded views occur, making it possible to choose inter-view prediction instead of temporal prediction. At the PU level, the encoder can choose in an RD optimal way if inter-view prediction or inter-frame prediction should be applied.
With this multiview compression scheme, forward compatibility with HEVC is guaranteed similarly to the forward compatibility provided by MVC. The same remark can be made for the view scalability aspect of MVHEVC. Additionally, in the proposed MVHEVC compression scheme, a complexity restriction is enforced by limiting the inter-view prediction within the same access unit.
The proposed MVHEVC compression scheme is graphically represented in Figure 4.9. From this figure, it is clear that the center view is configured as the compatible HEVC view. When the target application only supports stereo vision, the center view can be easily extracted together with either the left or right view. Indeed, both left and right views do not have dependencies other than the center view; therefore, partial decoding of the 3D bitstream is possible. A detailed schematic overview of the MVHEVC encoder is shown in Figure 4.10. Again, it is seen that the center view is encoded first, and only the reconstructed output is used for encoding the left

[7]RPS is similar to Memory Management Control Operations (MMCO) in H.264/AVC. However, RPS is a more improved version, which is more robust to data losses, is more flexible and easier to interpret for the decoder.

[8]The HEVC encoding structure is significantly different compared to H.264/AVC. Macroblocks are replaced by coding units (CUs). A quad-tree partitioning is used for each coding unit, starting from a size of 64x64 pixels. Each evaluated block is called a coding unit, independently of the size. Each coding unit is split further to obtain four new blocks until a minimal size of 8x8 pixels for each coding unit is reached. For each coding unit size, different PU sizes are evaluated. These PUs are comparable to the macroblock partitions in H.264/AVC. For each PU a motion vector is evaluated. The PU size with the lowest RD cost is selected, hence this also corresponds to a certain CU size.

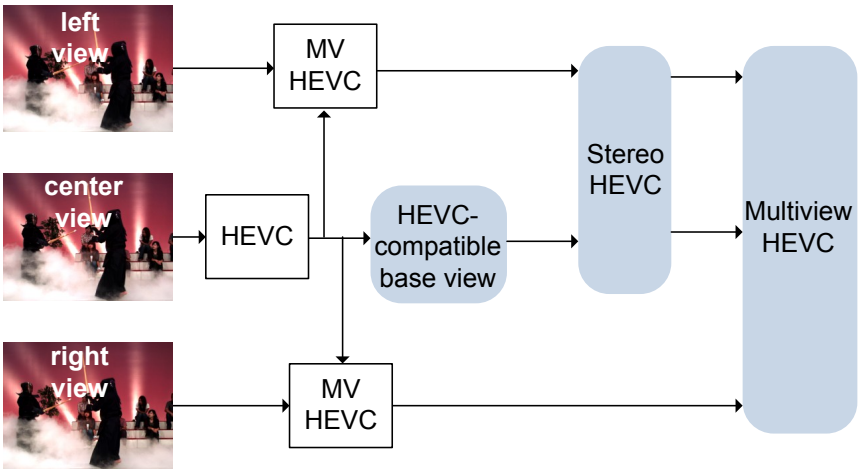


Figure 4.9: A Multiview HEVC (MVHEVC) architecture, which is similar in concept to MVC. The depth is encoded independently using the same architecture. High-level syntax indicates whether the NAL unit corresponds to texture or depth. The monoscopic and stereoscopic sub-streams are indicated.

view. Note that this is similar to MVC, and has a lower design complexity compared to SVC, since no low-level tools, and syntax inheritance between the layers has to be performed (cfr. encoder scheme in Figure 2.2).
For the multiview HEVC extension a gain of 37% in bit rate compared to simulcast HEVC is reported [19] for the texture, while on average gains of 5% for depth are obtained.

3D coding extension of HEVC

Within JCT-3V, multiple tracks are investigating 3D coding extensions of HEVC. Besides a multi-view extension of HEVC as presented above (without low-level changes), different solutions are examined which include specific provisions for depth map coding and the inclusion of low-level coding tools [21, 22]. The current test model under consideration includes tools such as disparity-compensated and view synthesis based inter-view prediction, inter-view motion prediction, and specific tools for coding of depth maps [23]. Disparity compensation is performed similar to motion compensation, but using the base view as a reference instead of a temporal prediction. View synthesis based inter-view prediction on the other hand, will

warp the base view with the decoded depth map, such that a low-complex view synthesis operation is performed. The resulting image is used as an additional predictor. No motion compensation is used for the view synthesis based inter-view prediction. Inter-view motion vector prediction allows to use the motion vector from the co-located block in the base view as a predictor for the current block. This reduces the syntax data for transmitting the motion vector. The compression efficiency of the depth maps is increased by re-using encoded data of the texture base view. The HEVC based 3D video Test Model (HTM) implementation is used as one of the reference points for the hybrid system.
A schematic overview of an HTM encoder is given in Figure 4.11. The main parts are comparable to the ATM encoder (see Figure 4.8)[9]. Some of the tools of ATM or also found in HTM, such as view synthesis prediction, joint texture-depth encoding and using depth maps for inter-view motion estimation. Additionally, HTM allows for some more tools such as the generation of a predicted depth map based on texture motion vectors (not shown) and inter view residual prediction. Again, to allow all tools to work most efficiently, the depth should be encoded after the texture center view and prior of the texture side views. Finally, HTM allows to use motion vector inheritance for depth map encoding, which has been suggested by [24] and allows to infer the motion vector from the texture. Nevertheless, if the evaluation of these tools does not yield a significant increase in compression efficiency compared to a basic set of tools, JCT-3V might still decide to remove some of these tools.

4.2 Proposed hybrid 3D architectures

The previously described 3D video coding technologies based on H.264/AVC either lack the high quality 3D perception (MFC) or have a limited coding effiency compared to HEVC based systems (MVC, ATM). On the other hand, HEVC based techniques (MVHEVC and HTM) have a high coding efficiency, but are not forward compatible with H.264/AVC. Therefore, HEVC based systems can not be incorporated in the network immediately, without the high cost of upgrading existing network infrastructure (such as encoders, streaming servers, transcoders, etc.) and the decoder install base.
In order to enable a system which offers compatibility to currently exist-

[9]Note that the HTM encoder is based on HEVC and consequently uses different algorithms to encode the video. However, for the sake of the discussion, the details of HEVC are not elaborated on.

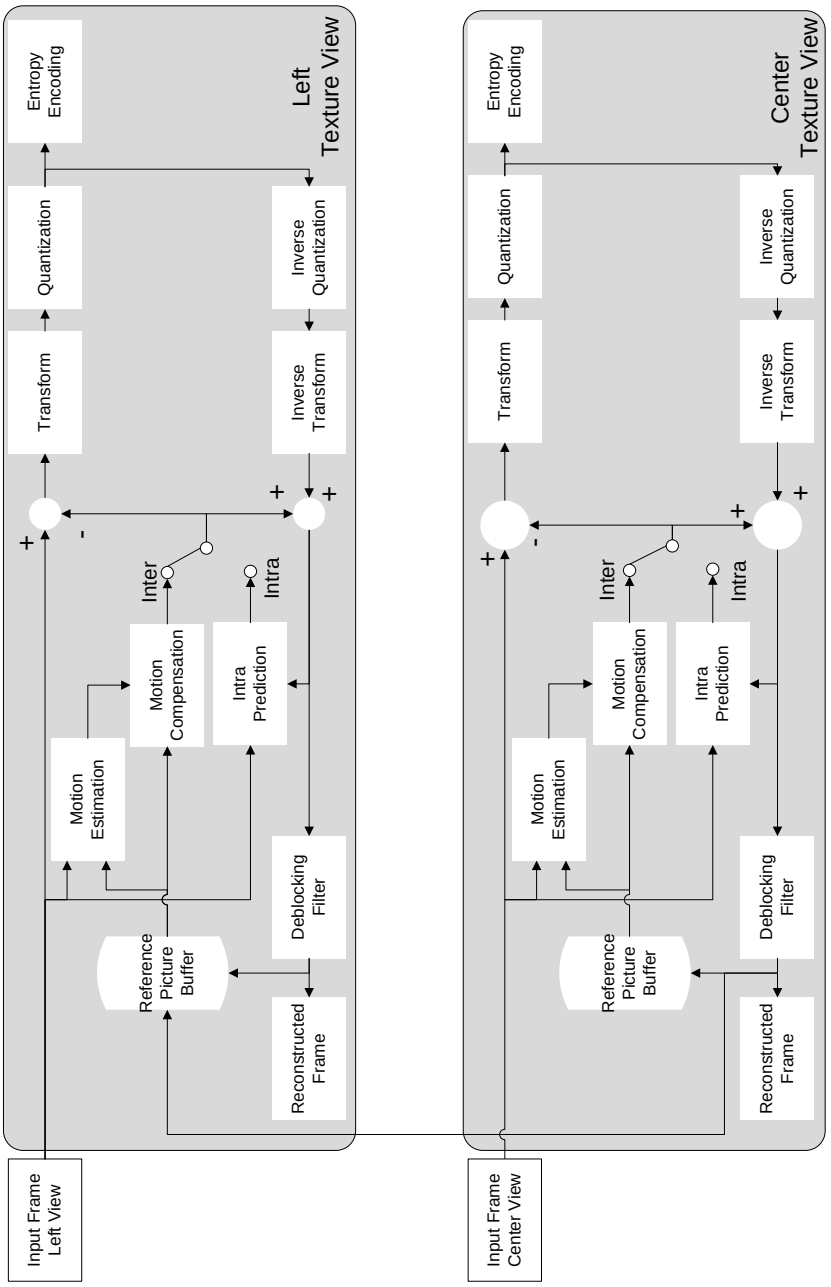


Figure 4.10: A schematic overview of the MVHEVC encoder, only allowing inter-view prediction as an additional tool for multiview coding.

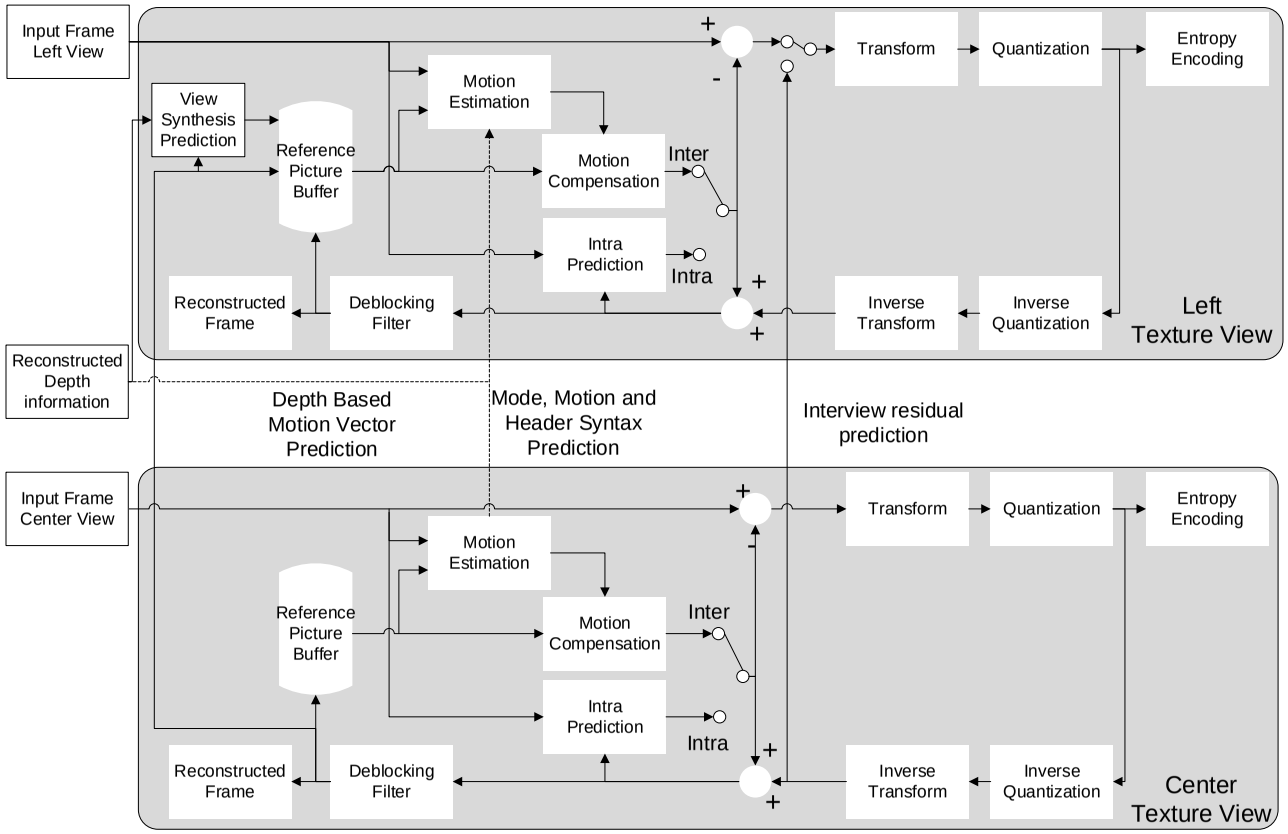


Figure 4.11: Schematic block structure of a HTM encoder applying low-level changes to the basic encoder design.

ing H.264/AVC based systems, 3D functionality, and a low overall bit rate, a hybrid architecture is proposed. The architecture is hybrid in a sense that the center view and side views are applying a different encoding standard. This is achieved by combining either H.264/AVC or MVC [25, 26] encoding for the center view and HEVC encoding for the left and right views. This architecture reduces the bandwidth by exploiting redundancy with base view streams (which are decodable by existing systems), while functionality of those systems is maintained on the mid-term. Furthermore, the additional functionality comes at a low cost due to the use of HEVC. Since current systems did not require depth information, the depth information is encoded completely in HEVC. This is done either as a MVHEVC system or with sub-Coding Unit (CU) changes (Section 4.1.3). This reduces the required bandwidth for the depth, without any functional limitations towards existing systems. The depth map is an 8-bit value (comparable to the Luma component of texture) which is an unsigned value. The interpretation of this data is done by the camera parameters, which are transmitted in a syntactical representation. These camera parameters contain information such as the distance of the closest and furthest object so the values of the depth map can be mapped in a 3D space. Currently, it is investigated if linear depth map representations are more favorable over non-linear representations. The quantization of the depth maps is correlated to the quantization of the texture views. However, additional research in this domain still has to be performed to find a good trade-off between the bit budgets for texture and depth.

Two compatibility scenarios are differentiated and for each of those hybrid architectures are proposed. The first scenario maintains forward compatibility with monoscopic video (H.264/AVC), the second scenario targets forward compatibility towards MVC and frame compatible coding. The former, allowing forward compatibility for H.264/AVC, results in a system where the base view of 3D video can still be transmitted using current 2D technologies and therefore no separate broadcasting infrastructure for 2D and 3D is required. The latter introduces forward compatibility for stereoscopic 3D. This allows for 2D and stereoscopic 3D systems to maintain operational while additional 3D video data is transmitted, without the need of a separate 3D broadcasting service.

Both proposed systems are in contrast with fully HEVC based 3D video. For fully HEVC based 3D scenarios, a simulcast transmission of H.264/AVC or MVC bitstreams is required. Therefore, the encoding complexity is limited since the encoder only has to encode the center view once (for H.264/AVC in stead as for both H.264/AVC and HEVC)[10]. Further-

[10]Note that the complexity of H.264/AVC and HEVC can not be compared since this is

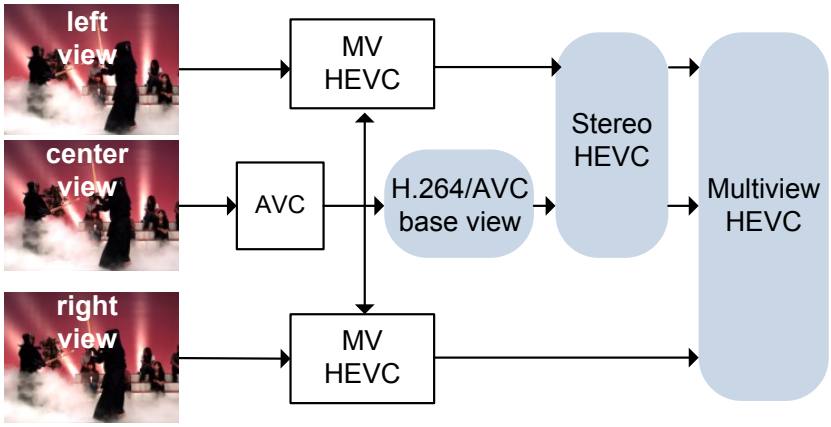


Figure 4.12: Architecture of the proposed Hybrid HEVC solution (encoder) providing forward compatibility with an H.264/AVC bitstream. The H.264/AVC bitstream represents a 2D version of the video sequence and is decodable by current legacy hardware. The monoscopic and stereoscopic sub-streams are indicated.

more, for the decoder side a hybrid architecture will also reduce the (design) complexity. For decoding multiview video, a shared memory is used. This is similar for both systems. However, devices have to be backward compatible with current technologies. Therefore, the hardware for an H.264/AVC decoder will be present in the system. Consequently, for hybrid 3D architectures, this H.264/AVC decoder can be re-used, while HEVC based systems require one HEVC decoder more to decode the center view. Given this more efficient use of hardware, and the lack of low-level tools proposed for hybrid 3D video compression, it can be concluded that the design complexity is reduced compared to fully HEVC architectures.

4.2.1 Monoscopic compatibility

Monoscopic compatibility for 3D video allows the current H.264/AVC infrastructure (network infrastructure, access networks, set-top boxes, decoders, storage systems, ...) to be be used for delivery and visualization of 2D video. Meanwhile, new or upgraded decoders are able to decode the full 3D bitstream such that, e.g., autostereoscopic displays can generate synthesized views.

implementation dependent.

Figure 4.12 shows the proposed hybrid architecture for 3D video with three views, where compatibility towards monoscopic video is maintained. The center view is encoded using H.264/AVC. The decoded center view output is used for inter-view prediction by both side views. Therefore, the side HEVC encoders have an additional reference picture available that can be used for prediction, as was the case for MVHEVC (Section 4.1.3). Since there is not necessarily a straightforward mapping between macroblocks and CUs, the potential gain by using inter-view syntax prediction will be limited. The decoded center view picture is stored in a shared memory buffer, which is accessible by the left and right views. The HEVC encoder indicates with a flag (*inter_view_prediction_flag*) for each PU whether or not inter-view prediction is used. This *inter_view_prediction_flag* is transmitted for each PU. Note that by applying this mechanism only to the pixel domain, no mapping issues between macroblock boundaries (H.264/AVC) and coding unit boundaries (HEVC) need to be solved.

4.2.2 Stereoscopic compatibility

In order to allow forward compatibility with recent developments in 3D technologies for consumer electronics, the proposed hybrid 3D architecture can also support stereoscopic compatibility. Two hybrid systems with stereoscopic compatibility based on (i) a frame-compatible base layer and (ii) a stereo MVC pair are discussed.

Frame-compatible 3D has been rapidly adopted in the market as a first phase in 3D delivery, thanks to its compatibility with installed base video transmission and display systems. A hybrid system can be constructed which is based on a frame-compatible scenario (in this case, side-by-side), as shown in Figure 4.13, where MFC pack is the multiview frame compatible packing. The center and left views are first horizontal sub-sampled with a 13-tap downsampling filter[11]. The coefficients of this seperable filter are given in Equation 4.1. The resulting (horizontally) downsampled center and left view are then packed together in a side-by-side arrangement. After decoding, the reconstructed half resolution center and left view are upsampled using an 11-tap filter, for which the coefficients are given in Equation 4.2. The upsampled center and left view serve as prediction for their respective full-resolution versions using MVHEVC. Similarly, the full-resolution center view serves as prediction for the right view. The depth maps are again completely encoded using MVHEVC.

[11]This filter has been provided by Philips in the scope of our joint MPEG-3D Call for Proposals activities. The filter design will not be further elaborated on.

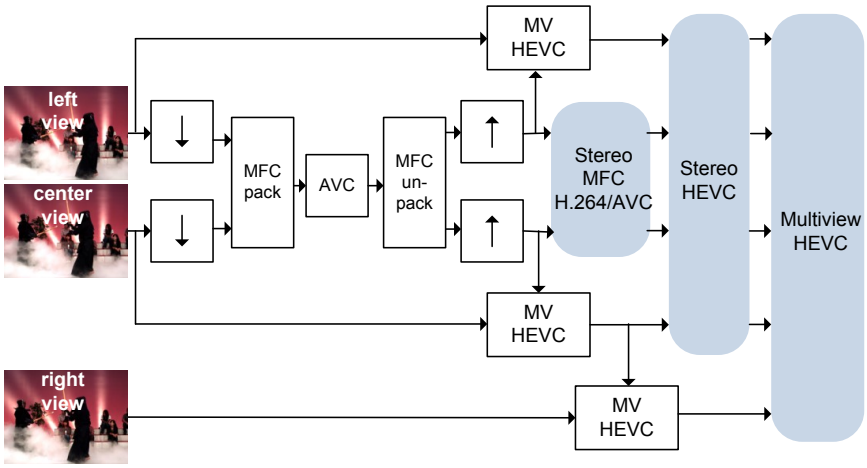


Figure 4.13: Architecture of the proposed Hybrid HEVC solution providing forward compatibility with an H.264/AVC bitstream representing a frame compatible stereo 3D version of the video sequence. Stereoscopic AVC compatible and hybrid sub-streams are indicated.

$$f_1 = [2, 0, -4, -3, 5, 19, 26, 19, 5, -3, -4, 0, 2]/64 \tag{4.1}$$

$$f_2 = [3, 0, -17, 0, 78, 128, 78, 0, -17, 0, 3]/256 \tag{4.2}$$

Since Blu-Ray 3D uses the Stereo High Profile of MVC, MVC is also proposed to be used as a basis for 3D video, as indicated in Figure 4.14. In this figure, the general case is included in which an optional spatial resolution difference may exist between the MVC coded version and the MVHEVC extension (indicated in gray). During migration of 3D systems, this could be done to save additional bandwidth for the legacy MVC version of the 3D stream, or to promote the high-quality hybrid 3D stream. Also, existing MVC content (e.g. in 720p format) could be efficiently extended to a higher resolution (e.g. 1080p) using this hybrid extension. As a special case, the resolution could be matched to that of frame-compatible systems (which also corresponds to the actual viewing resolution in polarized 3D displays). In that case, this architecture produces additional coding efficiency over the previous architecture by exploiting inter-view prediction between the center

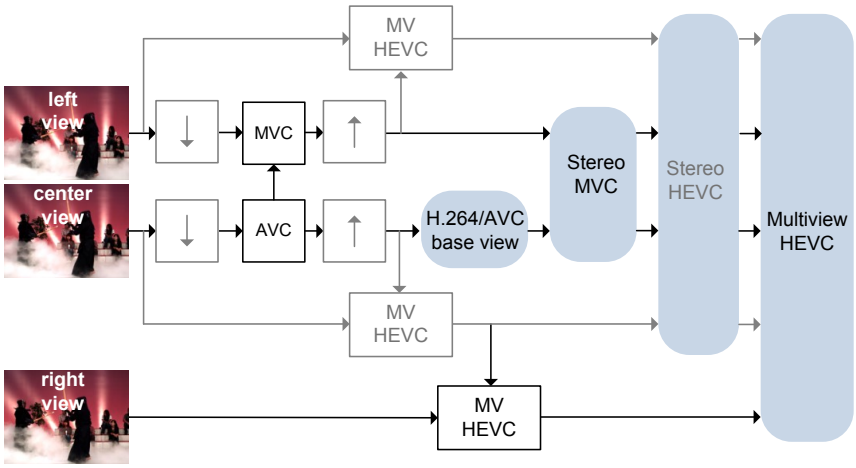


Figure 4.14: Architecture of the proposed Hybrid HEVC solution providing forward compatibility for MVC. The MVC bitstream represents a stereo 3D version of the video sequence compatible with current standards. The monoscopic and stereoscopic sub-streams are indicated.

and left views. Which of the views are used for MVC is of a lesser importance. In the remainder, the center and left view are used as MVC views, but center and right can be used as well. Moreover, the order of the views is not of importance for the architecture. Note that since MVC offers forward compatibility to H.264/AVC, the proposed stereoscopic compatible architecture is also forward compatible with monoscopic video.

After decoding, the (optionally) lower resolution MVC encoded views are upsampled. The proposed upsampling filter in Equation 4.2 is used, although any upsampling filter can be used as long as the up and downsampling filter does not result in a phase shift and the encoder and decoder use the same upsampling filter.

In the stereo compatible architectures, all MVHEVC encoder units share the same encoding architecture. This means that from a chip design perspective, the HEVC encoders can be replicated. For software design, the same instances can be initiated. So there is no need for a different encoding architecture for the upsampled HEVC views and for the inter-view predicted HEVC views, since both use the multiview HEVC architecture.

No depth information is required for legacy stereoscopic 3D, therefore, all depth maps can be encoded using a fully HEVC based architecture without harming compatibility. The center depth map uses an unmodified HEVC

encoder, whereas the satellite depth maps make use of a multiview HEVC to allow for the inter-view prediction. For all depth maps the full resolution is used as an input. This research does not aim at the high level syntax on how the depth and texture information should be signalled. In the following, both depth and texture are signalled independently in simulcast.

Quantization Parameter Considerations

As previously mentioned in Section 4.1, HEVC yields a significant bandwidth gain for the same subjective quality over H.264/AVC. Since both H.264/AVC and HEVC have the same quantization process, a QP difference can be introduced between the center view (H.264/AVC) and the satellite views (HEVC) without subjective quality loss. Since, theoretically, a QP difference of 6 corresponds with a doubled step size of the quantization, the bit rate will be halved. Off-line experiments with expert viewing have verified that using a $\Delta QP = 6$ between the center view and the side views: $QP_{HEVC} = QP_{AVC} + 6$, roughly obtains the same subjective quality for all views, with approximately half the bit rate of the side views compared to a ΔQP= 0.
Also, asymmetric quantization could be used, since human 3D perception does not detect small quality difference between views [27–29]. Therefore, a small difference in quantization (ΔQP) between the views could be introduced. The center view will have the same quality as used now for broadcasting, so the proposed architecture does not reduce the quality compared to the current industry standards. However, the left and right view could have a reduced quality. Since the depth information has specific characteristics, the quantization of the depth maps could also be adjusted, depending on whether or not the depth information is coded in full or half resolution. The common coding conditions for HTM, for example, define a table with a quantization parameter mapping between texture and depth [30]. Note that using unequal quantization for the views might result in eye fatigue. Several possible solutions for those kind of problems have been proposed [31–33]. The presented architecture does not apply asymmetric quantization settings but provided support to such features.

4.2.3 Notes on practical implementations

An important aspect of the proposed hybrid architectures over non-hybrid architectures is the concurrent use of both HEVC and H.264/AVC hardware. In current consumer electronics, both MPEG-2 and H.264/AVC chips are often available in a single set-top box. However, only one of the two is

used depending on the technology applied by the broadcaster. In the future, it is expected that hardware will have H.264/AVC as well as HEVC chips incorporated. Consequently, a hybrid solution will exploit the available hardware resources more efficiently by re-using the H.264/AVC hardware for the center view.

Compared to a multiview HEVC system (Figure 4.9), the proposed monoscopic compatible hybrid system (Figure 4.12) reduces the required number of on-chip HEVC decoders. Therefore, the production cost of such a system is reduced compared to multiview HEVC, which requires at least one additional HEVC decoder for the center view. Even if the design does not require for each view one HEVC decoder, but uses only one HEVC decoder at a higher speed, the proposed hybrid system gains significantly. Since it allows for a slower execution, both the design complexity (slower hardware) and energy consumption benefit from decoupling the base view with H.264/AVC.

Both MVHEVC and the proposed hybrid architectures, require additional synchronization between encoders. However, this can be limited to signaling when the side views can start encoding. No syntax information between the encoders must be communicated. Since current chips are designed with a shared memory, reference pictures are placed in this shared memory, which can be communicated along to the side views. This communication and synchronization is also required for MVHEVC, where the decoded HEVC output is used as a prediction. So, using H.264/AVC as a base layer is not imposing a higher complexity for the hardware.

The forward compatible hybrid architectures with MVC and frame compatible do introduce the same amount of HEVC encoders compared to MVHEVC. So the chip area will be similar to MVHEVC since the H.264/AVC chip should already be incorporated for backward compatibility. However, these systems also incorporate a downsampling and upsampling filter, which requires a slightly higher design complexity. However, this complexity comes with additional functionality. It has to be decided by the broadcaster if current hardware should incorporate the additional functionality for compatibility with current stereoscopic 3D television sets, or if the chip design should be as efficient as possible.

While the decoding complexity, energy consumption, or chip area might be reduced, one important issue remains. Since the encoder encodes different views simultaneously with different standards, timing is an important issue. Obviously, encoder design can also benefit from the fact that an H.264/AVC encoder might be available, at least the H.264/AVC design is profoundly tested, optimized and low in production cost. Nevertheless, a timing issue arises for encoding the dependent views. To encode these dependent

views, information from the center view has to be available. Therefore, the original side views should be stored in a buffer while the center view is encoded first, such that the decoded center view can be used as a predictor by the side view encoders. In case of a multi-core design, the left and right view can start encoding when the center view is finished encoding the first *search_range* macroblock rows. Only the first *search_range* macroblock rows have to be available because this is the maximum distance the motion vector of the side view can use when the center view is a predictor.
The timing between both encoders is an essential element to allow for efficient encoding. However, the same timing issue also arises for regular MVC, and multiview HEVC architectures. Consequently, solving these issues can be considered a general problem not specific for a hybrid solution.
The decoder introduces similar issues as the encoder. The basic MVHEVC approach requires three HEVC decoders. Meanwhile, these decoders have to be capable to store the decoded picture buffer in a shared memory, as is the case with current MVC decoders (used for e.g. Blu-ray).
Since the hybrid architecture combines H.264/AVC and HEVC technology, a single complexity number can not be given. All different solutions will come with there own implementation, so no common code base can be used to estimate the complexity. However, it is clear that the hybrid approach reuses the existing hardware encoders and limits the chip area. Therefore, it is safe to state that the design complexity is limited for hybrid architectures, while the benefits of the HEVC compression are exploited and backward compatibility for H.264/AVC is maintained.

4.3 Results

An objective evaluation of the proposed hybrid architectures (mono and stereo compatible) compared to the non-hybrid architectures and simulcast H.264/AVC and HEVC is performed in this section. Both proposed hybrid architectures have been implemented based on 3DV-ATM for the center view and a multi-view extension of HEVC Test Model (HM) [34] for the depth and texture side view(s) without using any low-level encoding tools. HM3.0 has been modified such that an additional uncompressed sequence can be used as additional input for the prediction step. The resulting implementation allows for inter-view prediction as described in Section 4.1.3.
The center view (for monoscopic compatibility) or center and left view (for stereoscopic compatibility) have been encoded using 3DV-ATM. For stereoscopic compatibility, the results for the MVC-based architecture discussed in Section 4.2.2 are included for the typical case without subsampling. The center view is used a predictor for the right view using

the modified HEVC encoder, which treats the center view as a temporal reference.
The compression efficiency is compared between non-hybrid and hybrid architectures. Compression efficiency is measured by the average rate distortion difference between two architectures. Evaluations are performed based on the MPEG 3D Video common test conditions [30] and Annex A of the Core Experiments description [35]. These test conditions stipulate the seven sequences to be used (*Poznan_Hall2*, *Poznan_Street*, *Undo_Dancer*, *GT_Fly*,
Kendo, *Balloons*, and *Newspaper*) and specify which views are to be encoded. The coding order is center-left-right. A GOP-size of 8 and intra-period of 24 frames is used. HEVC based architectures have to use the same resolution for depth and texture coding, while AVC based solutions have to reduce the depth map resolution.
Since six different architectures are evaluated for the non-hybrid architectures (simulcast H.264/AVC, simulcast HEVC, ATM-HP, ATM-EHP, MVHEVC, and HTM), and different test conditions for ATM and HTM have been described, the settings for this experiment are slightly adapted. ATM uses a reduced resolution depth map, encoded with the same quantization as the texture, while HTM uses the same resolution for texture and depth, but a higher quantization for depth maps. To allow a fair evaluation, all architectures have been encoded with the following conditions:

1. the same resolution for texture and depth
2. no unequal view quantization is applied
3. QP for depth is derived by Table 4.1 (as specified in [35])
4. Quantization for H.264/AVC based architectures:
   $QP_{AVC} \in \{21, 26, 31, 36, 41\}$
5. Quantization for HTM based architectures:
   $QP_{HTM} \in \{25, 30, 35, 40, 45\}$

Since the encoding performance of HEVC is significantly improved compared to H.264/AVC, the higher QPs will yield comparable rate points. Therefore, applying a different set of QPs for ATM and HTM will result in closely related RD curves and realistic conclusions. For the proposed hybrid solution, QP_{AVC} is used for the center view, while the left and right view have a higher quantization $QP_{side} = QP_{AVC} + 4$[12]. The depth

[12]Note that for the hybrid solutions $QP_{side} = QP_{AVC} + 6$ was proposed (see the

$QP_{texture}$	51	50	49	48	47	46	45	44	43
QP_{depth}	51	50	50	50	50	49	48	47	47
$QP_{texture}$	42	41	40	39	38	37	36	35	34
QP_{depth}	46	45	45	44	44	43	43	42	42
$QP_{texture}$	33	32	31	30	29	28	27	26	25
QP_{depth}	41	41	40	39	38	37	36	35	34

Table 4.1: Translation table for the quantization parameter for depth map encoding (QP_{depth}) given the texture quantization parameter ($QP_{texture}$).

view applied the same quantization for all views, according to the specified QPs in the common test conditions.

Seven sequences (*Poznan_Hall2*, *Poznan_Street*, *Undo_Dancer*, *GT_Fly*, *Kendo*, *Balloons*, and *Newspaper*) are encoded for three views (Left, Center and Right) including the corresponding depth views. The first four have a 1920x1088 resolution while the latter three have a 1024x768 resolution. These sequences are used within MPEG for evaluating standardization proposals, and the only native resolution content with three texture and corresponding depth views widely available for these resolutions[13].

Metrics for objective evaluation of stereoscopic images [36, 37] are still being developed. For multiview 3D, the problem is even more complex, and current research focuses on the evaluation of two rendered views [38, 39]. However, there is not yet a broad consensus for single objective evaluation measures to compare the coding efficiency for 3D video. Therefore, to indicate the RD for the texture information the sum of the texture bit rates is used together with the average PSNR of the texture views. For the RD of the whole system (texture + depth) the bit rate of texture and depth for each view is used, while only the average PSNR of the texture views is considered. The depth PSNR is not taken into account, because the PSNR of the depth maps has limited meaning and is generally high, even for such low bit rates. The coding performance of the depth is not taken into account seper-

remarks on Quantization Parameter Considerations in Section 4.2.2). The initial response to the call for proposals was evaluated with these settings. However, to be as close as possible to the common test conditions, for future evaluations $QP_{side} = QP_{AVC} + 4$ has been used. It was argued by Ghent University and Philips to change these settings during the MPEG meetings, although no consensus could be reached for the $QP_{side} = QP_{AVC} + 6$ settings.

[13]Note that the sequence *Loverbird1* is not longer included in the test conditions because the visual content is too distorted too give in reliable results.

	Texture Coding		Total bitstream	
	BDRate [%]	BDPSNR [dB]	BDRate [%]	BDPSNR [dB]
Poznan_Hall2	-52.56	2.12	-50.73	2.01
Poznan_Street	-50.78	2.06	-48.98	1.96
GT_Fly	-55.78	2.76	-55.88	2.81
Undo_Dancer	-57.72	3.02	-58.30	3.10
Kendo	-44.82	2.88	-41.03	2.47
Balloons	-47.15	3.30	-43.88	2.97
Newspaper	-44.14	2.52	-38.66	2.08
Average	-50.42	2.67	-48.21	2.49

Table 4.2: Average bit rate and PSNR gain for each sequence using the proposed hybrid architecture (mono compatible) compared to H.264/AVC simulcast. A negative value for BDRate indicates less bit rate and thus a gain, a positive BDPSNR indicates an increase in quality and thus a quality gain.

ately because the encoded depth is low in bandwidth. For each encoded sequence, RD-curves have been created by generating four rate points. From these rate points, for each sequence the BDRate and BDPSNR can be calculated, by using the average PSNR and combined bit rates. These measures are also used for objective 3D video evaluation within JCT-3V.

4.3.1 Comparison with simulcast

The proposed architecture is first compared with two simulcast scenarios: H.264/AVC and HEVC. Using simulcast, each texture and depth view is transmitted independently. Consequently, six encoded bitstreams are transmitted where no redundancy between the views is exploited. The benefit of using simulcast is that off-the shelf components can be used in the hardware design. Either one encoder or decoder is used at a higher speed, or multiple units are used concurrently. Since no changes in prediction structures and syntax elements are introduced, a low-cost implementation and production can be achieved. However, this comes at the expense of an additional cost in terms of bandwidth and efficiency.

	Texture Coding		Total bitstream	
	BDRate [%]	BDPSNR [dB]	BDRate [%]	BDPSNR [dB]
Poznan_Hall2	3.79	-0.09	3.54	-0.09
Poznan_Street	-38.04	1.41	-36.14	1.33
Undo_Dancer	-39.57	1.58	-39.57	1.6
GT_Fly	-21.82	0.75	-23.86	0.85
Kendo	-6.47	0.32	-4.97	0.24
Balloons	-12	0.64	-10.45	0.56
Newspaper	-25.77	1.3	-21.11	1.02
Average	-19.98	0.84	-18.93	0.79

Table 4.3: Average bit rate and PSNR performance for each sequence using the proposed hybrid architecture (mono compatible) compared to HEVC simulcast. A negative value for BDRate indicates less bit rate and thus a gain, a positive BDPSNR indicates an increase in quality and thus a quality gain.

H.264/AVC simulcast

The H.264/AVC simulcast results have been obtained by using the 3DV-ATM reference software for each view independently. 3DV-ATM is chosen to allow a fair comparison with other architectures, so that implementational differences are avoided. The proposed hybrid architecture (mono compatibility) yields around 50% in bit rate reduction, as can be seen in Table 4.2.

HEVC simulcast

As illustrated in Figure 4.15(a) for sequence *GT_Fly*, using H.264/AVC as a center view yields a significant loss compared to fully fledged HEVC. However, overall for the texture views 20% bit rate can be saved, as shown in Table 4.3. Here, the BDRate and BDPSNR gains for using the proposed hybrid architecture with monoscopic compatibility over an HEVC simulcast scenario are given. Consequently, the overall savings come from the inter-view prediction of the HEVC encoded side views, which gains significantly as can be seen in Figure 4.15(b) for sequence *GT_Fly*. All other sequences show comparable results. Both left and right views are compared between both architectures. A clear gap between both architectures is visible.

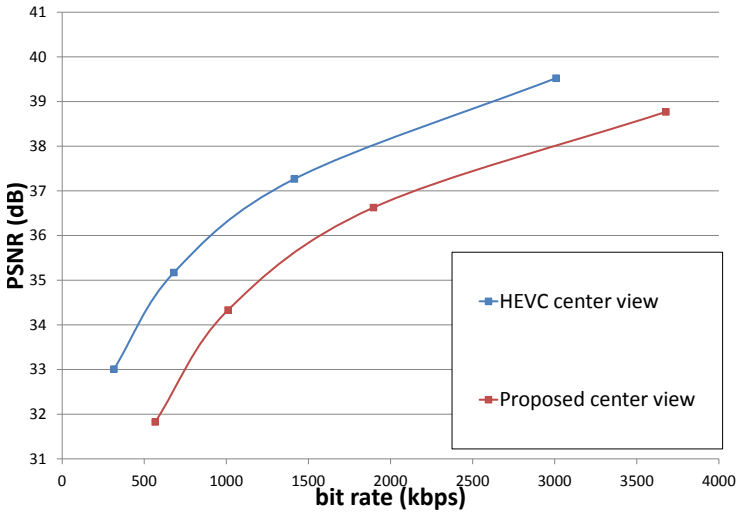
	Texture Coding		Total bitstream	
	BDRate [%]	BDPSNR [dB]	BDRate [%]	BDPSNR [dB]
Poznan_Hall2	-52.69	1.96	-46.61	1.7
Poznan_Street	-73.91	3.45	-69.72	3.16
Undo_Dancer	-78.48	4.58	-77.57	4.59
GT_Fly	-78.23	4.34	-77.14	4.33
Kendo	-51.5	3.23	-38.93	2.16
Balloons	-55.36	3.73	-46.48	2.91
Newspaper	-58.31	3.58	-49.49	2.81
Average	-64.07	3.55	-57.99	3.1

Table 4.4: Average bit rate and PSNR gain for each sequence using the proposed hybrid architecture (mono compatible) compared to HEVC simulcast excluding the center view. A negative value for BDRate indicates less bit rate and thus a gain, a positive BDPSNR indicates an increase in quality and thus a quality gain.

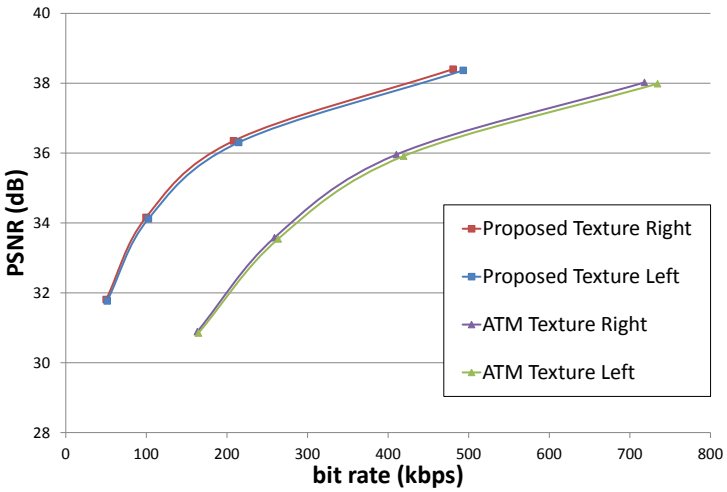
An overview of the gains for the HEVC side views can be found in Table 4.4. In this table the RD is compared by only taking the left and right view for texture and depth into account. In both architectures, all considered views are encoded using HEVC. Due to the inter-view prediction, a 58% gain in bit rate is noticed for the proposed hybrid architecture. Figure 4.16 provides the overall RD curves for all 1920x1088 sequences, here clearly the gain due do the improved side view encoding is illustrated. Note that for one sequence (*Poznan_Hall2*) the gain achieved in the side views is not enough to perform significanlty better than HEVC simulcast. A clear indication other than the content properties is not found for this behaviour.

4.3.2 Comparison with MVC

In order to generate the MVC sequences, 3DV-ATM is used. Texture and depth are generated as two independent MVC stream and transmitted in simulcast. For this configuration to work in real-world systems, syntax elements should be implemented to indicate the function (texture or depth) of each package. However, for the sake of this performance analysis, the overhead for such functionality is limited and will not be considered. Since the base layer of both architectures is H.264/AVC, the center texture views share the same rate and distortion. All gain reported is obtained by applying



(a) Center view of *GT_Fly* comparing the HEVC and H.264/AVC encoding.



(b) Left and right view of *GT_Fly*.

Figure 4.15: RD results for sequence *GT_Fly* show the bit rate loss for the center view, and the gains for the side views due to inter-view prediction.

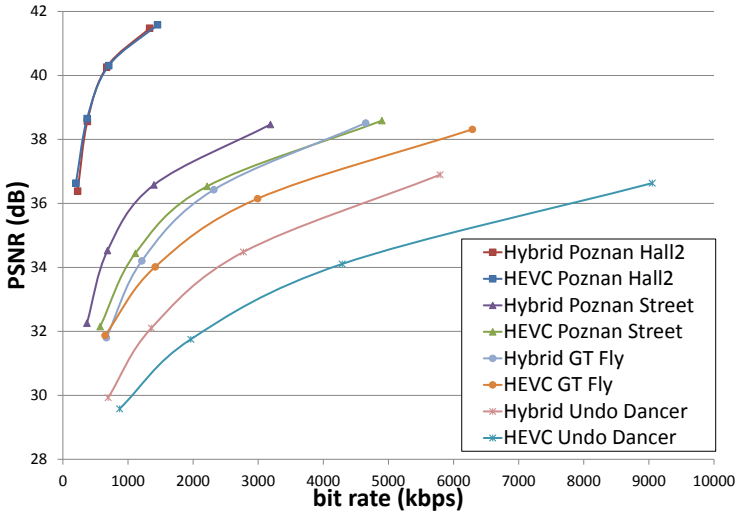


Figure 4.16: Overall RD gain for all 1920x1088 sequences for the proposed architecture compared to monoscopic compatibility.

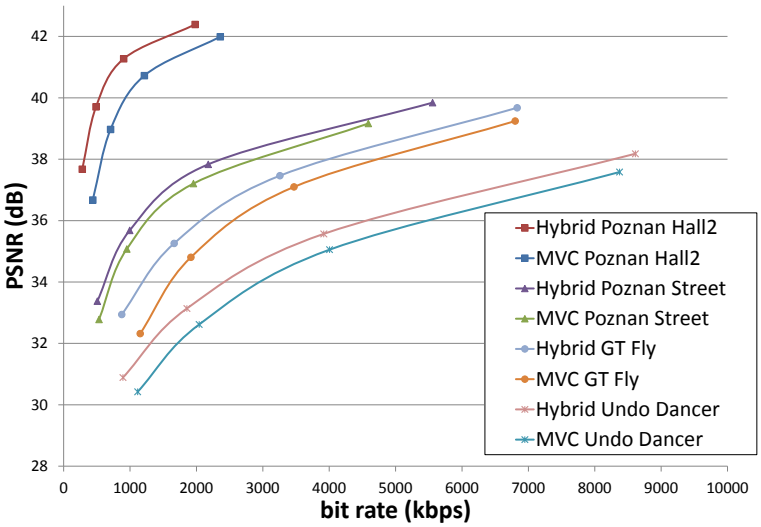


Figure 4.17: Comparison of the RD performance for the total bitstream for the monoscopic compatible architecture compared to the MVC architecture (1920x1088 sequences)

Table 4.5: Detailed RD results for texture and depth of sequence *Kendo* when compared to MVC.

	Texture									
Rate [kbps]	View 1		View 3		View 5		Total		Performance (texture)	
PSNR [dB]	Rate	PSNR	Rate	PSNR	Rate	PSNR	Rate	PSNR	BDRate[%]	BDPSNR[dB]
	716.47	42.33	964.22	42.36	753.53	42.01	2434.22	42.23		
	405.01	40.05	574.85	40.21	425.97	39.81	1405.83	40.02		
MVC	243.61	37.44	349.32	37.64	252.37	37.23	845.30	37.44		
	154.99	34.62	221.25	34.76	159.83	34.43	536.07	34.61		
	550.66	43.42	964.22	42.36	582.25	43.07	2097.13	42.95		
	268.42	40.89	574.85	40.21	283.48	40.72	1126.75	40.60		
Hybrid	144.04	38.27	349.32	37.64	151.25	38.20	644.61	38.04		
	78.32	35.46	221.25	34.76	80.78	35.44	380.35	35.22		
									-31.47	**1.75**

Continued on next page

Table 4.5 – *Continued from previous page*

	Depth									
Rate [kbps]	View 1		View 3		View 5		Total (overall)		Performance (overall)	
	Rate	PSNR	Rate	PSNR	Rate	PSNR	Rate	PSNR	BDRate[%]	BDPSNR[dB]
	288.41	43.68	228.46	44.87	324.98	43.03	3276.07	42.23		
	166.93	40.70	133.39	41.89	189.06	40.10	1895.21	40.02		
MVC	94.02	37.60	76.11	38.62	107.62	37.02	1123.05	37.44		
	49.77	34.13	39.18	34.80	56.14	33.52	681.16	34.61		
	212.47	42.83	164.43	43.97	242.78	42.22	2716.81	42.95		
	109.25	39.66	84.37	40.83	124.69	38.98	1445.05	40.60		
Hybrid	59.68	37.09	46.10	38.27	67.55	36.37	817.94	38.04		
	32.64	34.57	25.70	35.86	36.68	33.89	475.37	35.22		
									-36.09	**2.14**

	Texture Coding		Total bitstream	
	BDRate [%]	BDPSNR [dB]	BDRate [%]	BDPSNR [dB]
Poznan_Hall2	-43.29	1.47	-43.57	1.50
Poznan_Street	-13.77	0.43	-16.01	0.51
Undo_Dancer	-19.46	0.72	-21.11	0.79
GT_Fly	-20.58	0.78	-23.30	0.90
Kendo	-31.47	1.75	-34.62	1.93
Balloons	-25.49	1.40	-27.19	1.52
Newspaper	-19.11	0.90	-20.89	0.98
Average	-24.74	1.06	-26.67	1.16

Table 4.6: Average bit rate and PSNR gain for each sequence using the proposed hybrid architecture (mono compatibility) compared to MVC. A negative value for BDRate indicates less bit rate and thus a gain, a positive BDPSNR indicates an increase in quality and thus a quality gain.

the multiview HEVC to the side views and depth information. However, the total bit rate of all views is taken into account.
To maintain readability and reduce load on the graphs, the RD-curves for the 1920x1088 sequences for the combined texture views (i.e., the total bit rate and the average PSNR of each texture view) are provided. In Figure 4.17, significant gains can be seen independently of the sequence.
Table 4.5 shows a detail of the RD points for each view for MVC and hybrid for the *Kendo* sequence. From this multi-view sequence, three views were used. View 3 represents the center view, while views 1 and 5 represent the left and right view of the *Kendo* sequence. For this sequence, a gain of 36.09% in BDRate or 1.75 dB in BDPSNR is reported. Table 4.6 shows for each sequence the average BDRate and BDPSNR gain for the texture views. On average, for all sequences the proposed hybrid architecture gains 24.74% in BDRate. If also the depth information is taken into account, the BDRate gain increases to 26.67%. The difference in gain is because the center view of the depth is also encoded using HEVC.

4.3.3 Comparison with ATM

Table 4.7 shows the results when compared to the ATM-HP configuration, while in Table 4.8 the BDRate and BDPSNR results are given compared to the EHP-configuration. On average, the proposed hybrid architecture

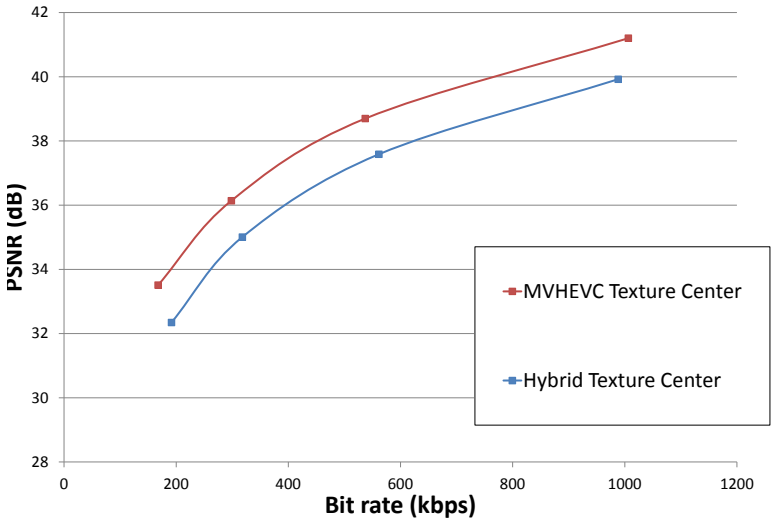
	Texture Coding		Total bitstream	
	BDRate [%]	BDPSNR [dB]	BDRate [%]	BDPSNR [dB]
Poznan_Hall2	-43.23	1.78	-43.85	1.82
Poznan_Street	-19.42	0.64	-21.34	0.71
Undo_Dancer	-25.89	1.03	-27.58	1.13
GT_Fly	-25.24	1.05	-28.07	1.20
Kendo	-32.22	1.93	-36.09	2.14
Balloons	-33.32	2.08	-35.54	2.22
Newspaper	-22.35	1.10	-23.43	1.14
Average	-28.81	1.37	-30.84	1.48

Table 4.7: Average bit rate and PSNR gain for each sequence using the proposed hybrid architecture (mono compatibility) compared to ATM-HP. A negative value for BDRate indicates less bit rate and thus a gain, a positive BDPSNR indicates an increase in quality and thus a quality gain.

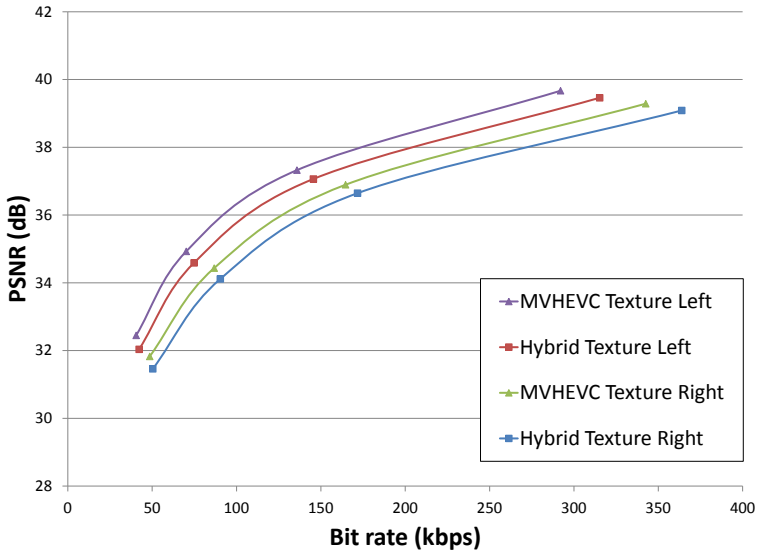
with mono compatibility yields a reduction of 23.44% in bit rate or a gain of 1.07 dB for the total bitstream; 19.97% or 0.90 dB respectively for the texture views. Since the optimizations of ATM compared to MVC affect the side views, the gain by using the hybrid architecture for the texture views slightly reduces.
Note that for the overall RD, according to the common test conditions, the depth maps by using ATM are encoded using a half resolution, while for the hybrid architecture, the full resolution depth maps are encoded. Nevertheless the use of HEVC gains more than 20% compared to ATM based scenarios.

4.3.4 Comparison with MVHEVC

The difference between the hybrid architecture and the MVHEVC architecture is caused by the difference in coding performance of the center texture view. In the hybrid case, the center view is encoded using H.264/AVC, while HEVC encoding is applied for MVHEVC. Therefore, the coding efficiency of the center view will determine the total benefit of using MVHEVC over the hybrid architecture. The gain of HEVC over H.264/AVC for the center view is illustrated in Figure 4.15(a) and Figure 4.18(a). Furthermore, since the resulting decoded center view will be slightly different between both scenarios, the left and right view of the hybrid and MVHEVC bit-



(a) Center view for the *Newspaper* sequence.



(b) Left and right view for the *Newspaper* sequence.

Figure 4.18: RD performance for MVHEVC and the monoscopic compatible hybrid architecture for the *Newspaper* sequence.

	Texture Coding		Total bitstream	
	BDRate [%]	BDPSNR [dB]	BDRate [%]	BDPSNR [dB]
Poznan_Hall2	-33.68	1.26	-35.6	1.35
Poznan_Street	-12.18	0.38	-14.88	0.48
Undo_Dancer	-15.45	0.56	-17.97	0.68
GT_Fly	-8.78	0.32	-13.82	0.53
Kendo	-25.74	1.45	-32.01	1.81
Balloons	-26.38	1.52	-30.07	1.76
Newspaper	-17.58	0.83	-19.7	0.92
Average	-19.97	0.90	-23.44	1.07

Table 4.8: Average bit rate and PSNR gain for each sequence using the proposed hybrid architecture (mono compatibility) compared to ATM-EHP. A negative value for BDRate indicates less bit rate and thus a gain, a positive BDPSNR indicates an increase in quality and thus a quality gain.

streams will not be identical. In general, it is noticed that the HEVC encoded center view results in a better prediction for the side views. This can be seen in Figure 4.18(b), where for sequence *Kendo* the left and right views of both architectures are shown.
Since both systems use the same depth encoding, there is no additional gain for the depth encoding for MVHEVC. The average bit rate reduction of MVHEVC is 26.71% with a BDPSNR gain of 1.09 dB. If the depth is taken into account, the average bit rate for the total 3D information gains 24.68% or the average PSNR gains 0.98 dB.
Table 4.9 shows the coding gain for using MVHEVC over the proposed hybrid architecture. On the other hand, MVHEVC does not benefit from multi-codec chip implementations and consequently does not allow for easy market adoption since no forward compatibility for H.264/AVC is maintained.

4.3.5 Comparison with HTM

From all architectures currently considered within JCT-3V, HTM has the best performance. Due to the additional low-level optimizations, the performance of HTM will improve compared to MVHEVC. Note that also the depth map encoding for HTM gains because of the sub-coding unit adaptations.

Table 4.10 shows the results of HTM compared to the hybrid architecture for both the total bit rate and the texture views. Compared to the hybrid architecture, HTM gains 34.81% in bit rate on average for all sequencess, or the PSNR increases with 1.48 dB for the texture views. The gain slightly increases if also the depth is taken into account. For the overall system, a gain of 38.77% in bit rate and 1.75 dB in PSNR is reported for HTM.
The objective evaluation shows that the hybrid architecture gains significantly in rate-distortion performance compared to fully H.264/AVC based systems. Using inter-view prediction for the HEVC side views approximately halves the bit rate compared to standard HEVC. Therefore, the proposed hybrid solutions outperforms HEVC simulcast. Multiview HEVC and HEVC with sub-coding unit adaptations even further reduce the bandwidth of 3D video.

4.3.6 Overview of the results

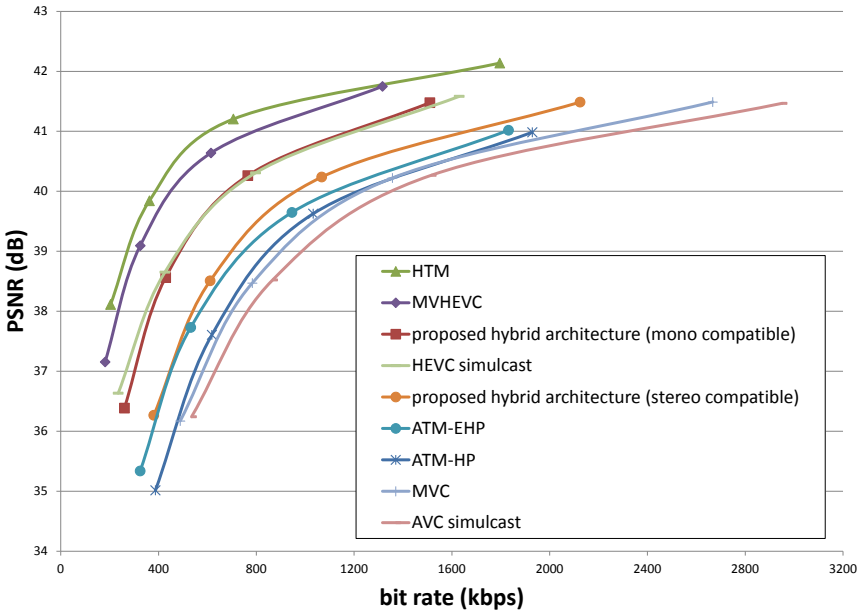
Figure 4.19 shows the RD-curves for each sequence encoded using the different described architectures. As expected, the proposed hybrid architecture performs in between fully H.264/AVC architectures and HEVC based architectures. The coding efficiency of all architectures is dependent on the content which is used. Since different content or base view separation yields different results, some prediction mechanisms might not be fully exploited. Finally, for both H.264/AVC and HEVC systems, it can be seen that for the high bit rate range, sub-macroblock or sub-coding unit adaptations of the architecture only yield a small gain.
Figure 4.20 shows the overhead required for adding 3D functionality compared to single-view H.264/AVC for the texture data, i.e., without taking the depth into account. This is shown for each H.264/AVC based architecture, since the HEVC based architectures do not support compatibility with H.264/AVC. The overhead for adding stereo 3D in a simulcast solution will be the bit rate for one H.264/AVC view, while for introducing three-view 3D video, the bit rate of the H.264/AVC left and right views is required. Compared to this simulcast overhead, it can be seen from the results that only 22% of this overhead is required for adding 3D functionality to monoscopic H.264/AVC using the presented hybrid architecture. If MVC is used on top of monoscopic H.264/AVC, 62% of the additional bit rate of simulcast H.264/AVC is required, while ATM-HP and ATM-EHP need 47% and 36% of the simulcast overhead. Note that these numbers are the same for stereo and multiview 3D functionality (for each additional view, the bit rate increases by an average of 22% of the simulcast overhead for the hybrid solution).

	Texture Coding		Total bitstream	
	BDRate [%]	BDPSNR [dB]	BDRate [%]	BDPSNR [dB]
Poznan_Hall2	-37.87	1.15	-34.93	1.06
Poznan_Street	-16.87	0.50	-16.59	0.50
Undo_Dancer	-19.51	0.69	-19.03	0.68
GT_Fly	-39.20	1.49	-37.56	1.43
Kendo	-28.48	1.50	-23.25	1.15
Balloons	-27.44	1.50	-24.62	1.30
Newspaper	-17.63	0.80	-16.79	0.75
Average	-26.71	1.09	-24.68	0.98

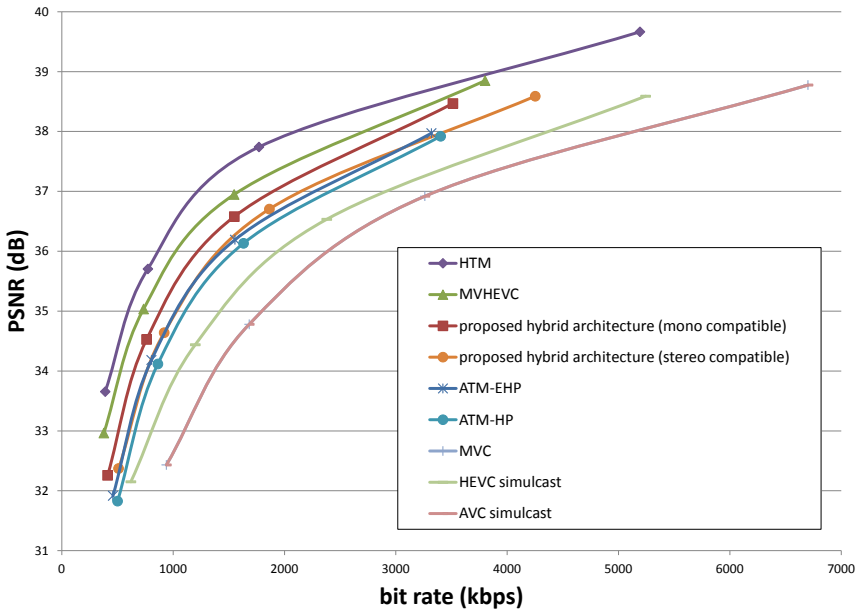
Table 4.9: Average bit rate and PSNR gain for each sequence using MVHEVC compared to the proposed hybrid architecture (mono compatibility). A negative value for BDRate indicates less bit rate and thus a gain, a positive BDPSNR indicates an increase in quality and thus a quality gain.

	Texture Coding		Total bitstream	
	BDRate [%]	BDPSNR [dB]	BDRate [%]	BDPSNR [dB]
Poznan_Hall2	-43.14	1.29	-43.72	1.35
Poznan_Street	-28.1	0.88	-30.86	0.99
Undo_Dancer	-26.66	1.00	-25.23	0.95
GT_Fly	-43.41	1.73	-43.46	1.76
Kendo	-36.03	1.93	-47.90	2.75
Balloons	-39.35	2.24	-45.98	2.76
Newspaper	-26.95	1.27	-34.24	1.70
Average	-34.81	1.48	-38.77	1.75

Table 4.10: Average bit rate and PSNR gain for each sequence using HTM compared to the proposed hybrid architecture (mono compatibility). A negative value for BDRate indicates less bit rate and thus a gain, a positive BDPSNR indicates an increase in quality and thus a quality gain.

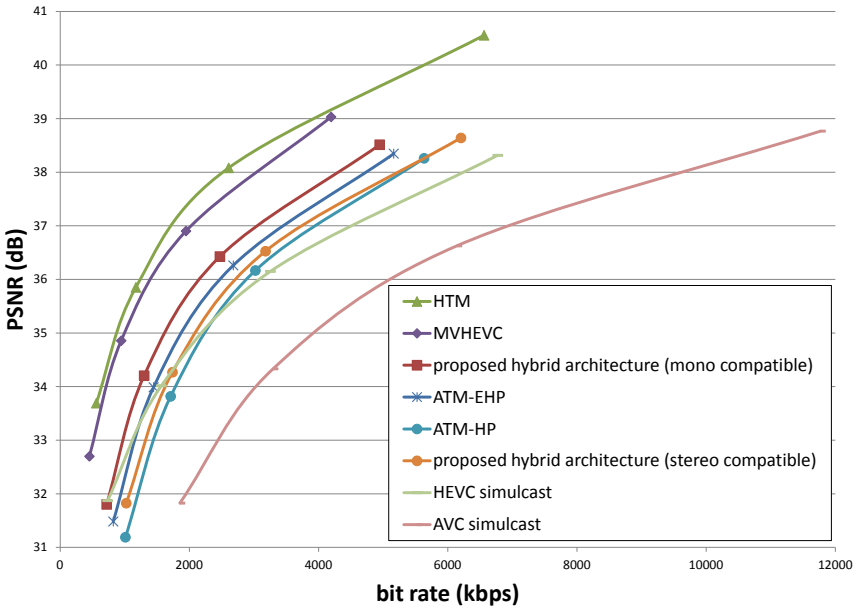


(a) *Poznan_Hall2*

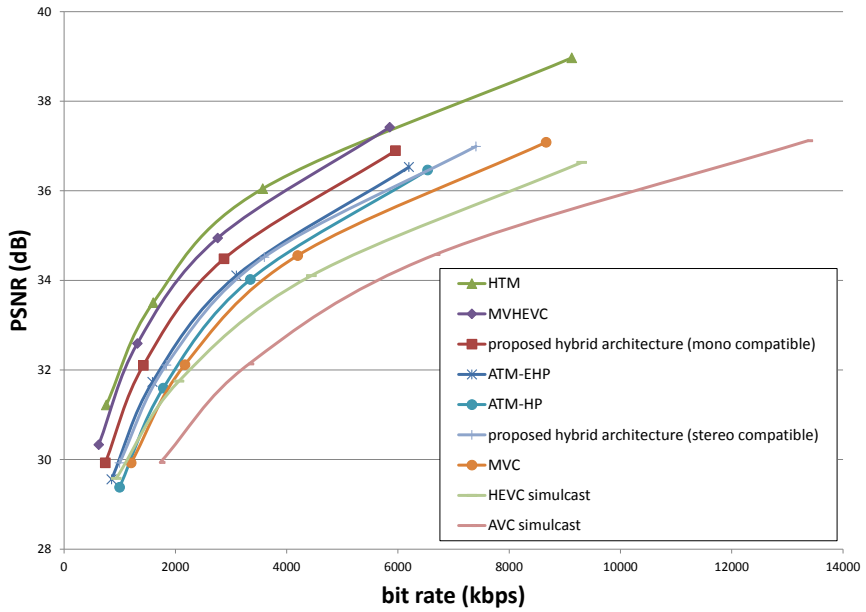


(b) *Poznan_Street*

Figure 4.19: Overview of the coding performance of all described architectures for all sequences (1920x1088).

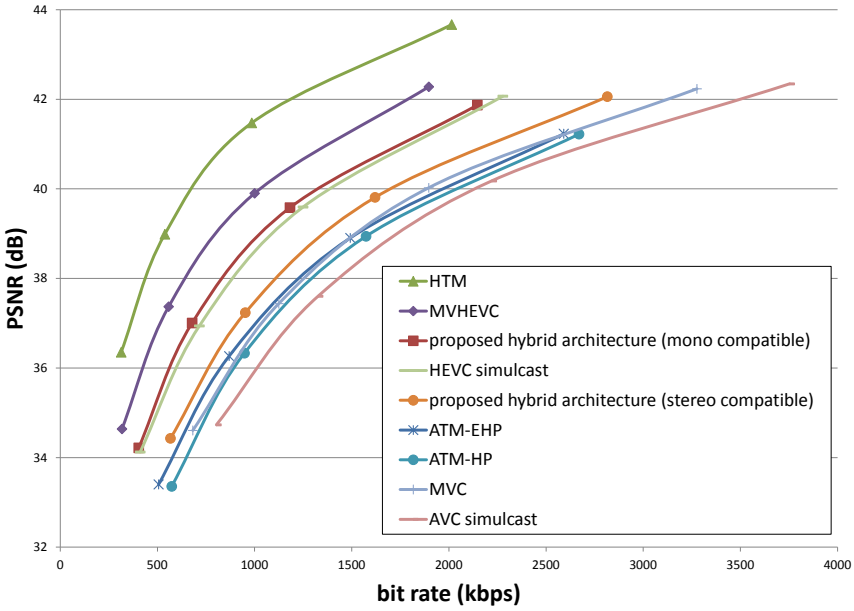


(c) *GT_Fly*

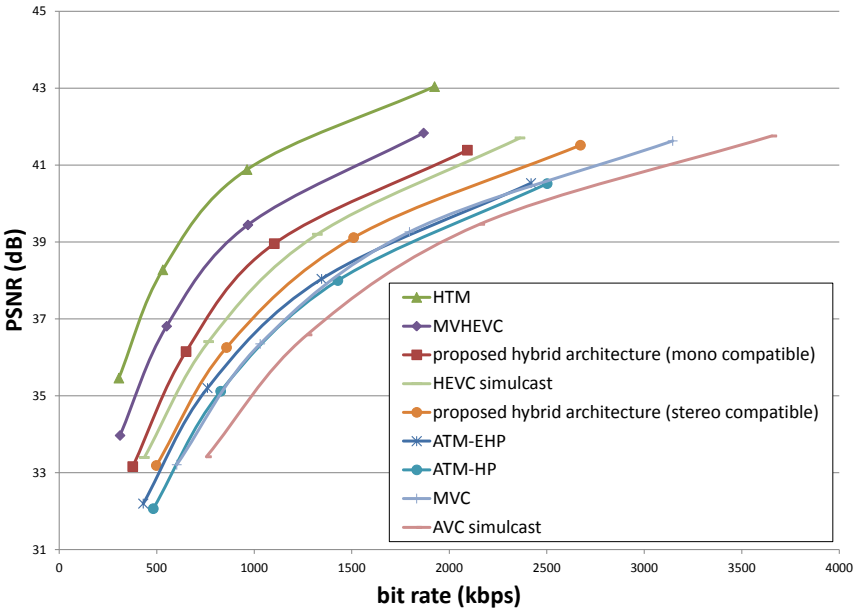


(d) *Undo_Dancer*

Figure 4.19: Overview of the coding performance of all described architectures for all sequences (1920x1088) Cont.

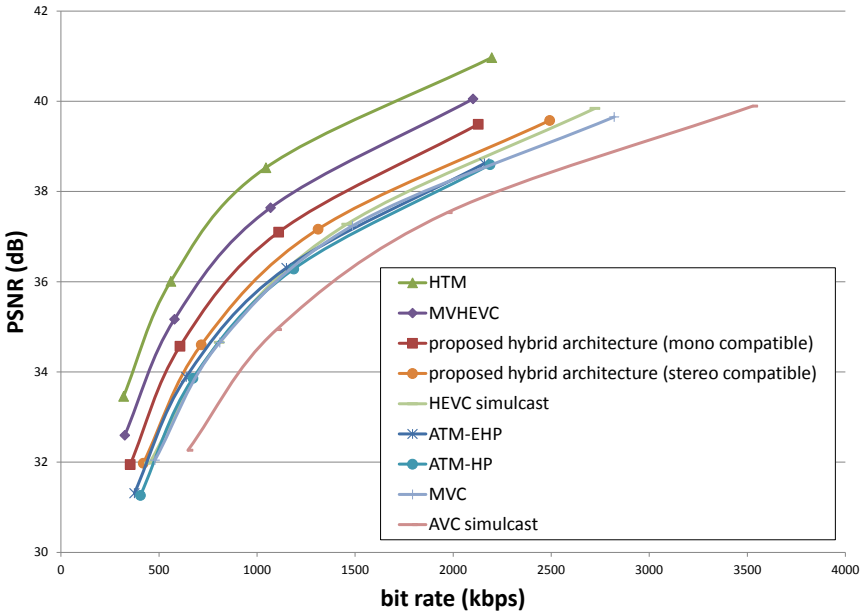


(e) *Kendo*



(f) *Balloons*

Figure 4.19: Overview of the coding performance of all described architectures for all sequences (1024x768)



(g) *Newspaper*

Figure 4.19: Overview of the coding performance of all described architectures for all sequences (1024x768) (cont.)

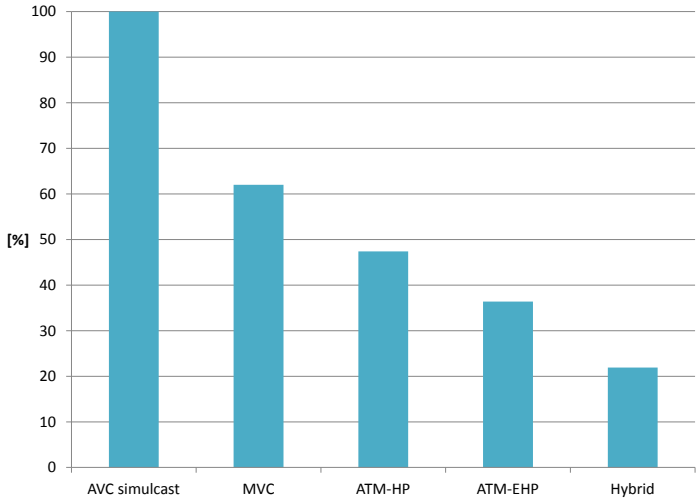


Figure 4.20: Average incremental overhead for additional views on top of monoscopic H.264/AVC (relative to simulcast).

Architecture			BDRate(%)
Hybrid	compared to	H.264/AVC simulcast	-48.21
		MVC	-26.67
		ATM-HP	-30.84
		ATM-EHP	-23.44
		HEVC simulcast	-18.93
MVHEVC	compared to	Hybrid	-24.68
HTM			-38.77

Table 4.11: Coding efficiency performance overview for the H.264/AVC and HEVC based architectures compared to the proposed hybrid architecture (lowest coding efficiency on top).

Finally, Table 4.11 compares the relative overhead of the different architectures. The BDRate difference is given for the proposed hybrid architecture compared to each architecture. For MVHEVC and HTM the gains are expressed in favor of these architectures, while for the other architectures the BDRate is expressed as gain for the hybrid architectures.

4.4 HEVC Extensions

The proposed hybrid coding solutions were designed in the context of 3D applications and standardization. It is clear, however, that hybrid coding can also be considered for other applications. Recently, MPEG has launched a Call for Proposals for a scalable extensions of HEVC [40]. The requirements for this scalable extensions of HEVC [41] specifies that a hybrid combination of coding standards is considered. This hybrid combination is not only targeted for view scalability, but also for spatial scalability. For spatial scalability, one or more modes (profiles) should be defined where an encoder and decoder are able to interact with a base layer that is compliant with AVC Standard's Constrained Baseline Profile, Main Profile and High Profile. The enhancement layer is then HEVC encoded with additional coding tools to allow prediction from the H.264/AVC base layer. The mechanism used to achieve compatibility with the AVC standard shall not cause an unacceptable increase in complexity and reduction in the performance of the HEVC extension when it is operating in a mode (Profile) with an HEVC-compliant base layer.
Considering the expected increase in popularity of tablets with 3D viewing capabilities, combining spatial scalability with view scalability becomes of

interest. Such a system is able to cover both 3D TV with large screens and 3D viewing on tablet devices. Obviously systems combining H.264/AVC and HEVC are similar to the presented architecture in Section 4.2.1. However, the applicability of such systems reaches further than only 3D applications.

4.5 Conclusions on Hybrid 3D video coding

Currently, different architectures for 3D video are being investigated for standardization within MPEG. A hybrid architecture is presented which provides forward compatibility with mono (H.264/AVC) or stereo (MVC or frame compatible) coding. The additional views (or full-resolution view refinements) are encoded based on a multi-view extension of HEVC. Here, the HEVC encoder has been adapted such that reconstructed pictures of the center view can be used as a reference for inter-view prediction.
Besides the advantage of offering forward compatibility, the overhead for offering 3D video is limited in the presented hybrid architectures. Firstly, the 2D bitstream is fully incorporated in the 3D video stream. Secondly, the efficiency of HEVC is exploited for encoding the side views, yielding an additional coding gain.
An extensive overview of the performance of the presented hybrid architectures has been given, in comparison with the non-hybrid ATM and HTM tracks which are currently investigated within MPEG. The results show that the performance of the hybrid architecture is in between the performance of a fully HEVC system and ATM.
The presented architecture can be used on a short term as an intermediate standard to allow 3D video to be delivered at a relatively low cost. Meanwhile, the upcoming HEVC standard can gain more popularity and the base for upgrading all types of video delivery to HEVC will grow. Therefore, the presented architecture will ameliorate the HEVC market introduction. After a roll-out of HEVC for 2D, the presented architecture easily allows to upgrade towards a fully HEVC system (either MVHEVC or HTM).

The research described in this chapter resulted to the following publications.

- Glenn Van Wallendael, Sebastiaan Van Leuven, Jan De Cock, Fons Bruls, and Rik Van de Walle, *"3D Video Compression based on High Efficiency Video Coding"*, in IEEE Transactions on Consumer Electronics, 58(1), pp 137-145, Feb. 2012.

- Glenn Van Wallendael, Sebastiaan Van Leuven, Jan De Cock, Fons Bruls, and Rik Van de Walle, *"Multiview and Depth Map Compression based on HEVC"*, in Proc. of the 2012 IEEE International Conference on Consumer Electronics (ICCE), pp. 168-169, Jan. 2012, USA.

- Sebastiaan Van Leuven, Hari Kalva, Glenn Van Wallendael, Jan De Cock, and Rik Van de Walle, *Joint Complexity and Rate Optimization for 3DTV Depth Map Encoding*, in Proc. of the 2013 IEEE International Conference on Consumer Electronics (ICCE), pp. 191-192, Jan. 2013, USA.

- Fons Bruls, Glenn Van Wallendael, Jan De Cock, Bart Sonneveldt, and Sebastiaan Van Leuven, "Description of 3DV CfP submission from Philips & 'Ghent University - IBBT'", ISO/IEC MPEG Doc. m22603, Dec. 2011, Switzerland.

- Sebastiaan Van Leuven, Fons Bruls, Glenn Van Wallendael, Jan De Cock, and Rik Van de Walle, *"Hybrid 3D Video Coding"*, ISO/IEC MPEG Doc. m23669, Feb. 2012, USA.

- Sebastiaan Van Leuven, Glenn Van Wallendael, Jan De Cock, Fons Bruls, Ajay Luthra, and Rik Van de Walle, *"Overview of the coding performance of 3D video architectures"*, ISO/IEC MPEG Doc. m24968, Apr. 2012, Switzerland.

References

[1] A. Vetro, S. Yea, and A. Smolic. Towards a 3D video format for auto-stereoscopic displays. In *Proceedings of SPIE Conference on Applications of Digital Image Processing XXXI*, Aug. 2008.

[2] M. Tanimoto. Overview of free viewpoint television. *Signal Processing: Image Communication*, 21(6):454 – 461, 2006.

[3] E. Martinian, A. Behrens, J. Xin, and A. Vetro. View Synthesis for Multiview Video Compression. In *Proceedings of Picture Coding Symposium (PCS)*, Apr. 2006.

[4] K. Yamamoto, M. Kitahara, H. Kimata, T. Yendo, T. Fujii, M. Tanimoto, S. Shimizu, K. Kamikura, and Y. Yashima. Multiview video coding using view interpolation and color corrections. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(11):1436–1449, Nov. 2007.

[5] View Synthesis reference software 3.0. `http://wg11.sc29.org/svn/repos/MPEG-4/test/trunk/3D/view_synthesis/VSRS`. Technical report, MPEG, May 2009.

[6] ISO/IEC JTC1/SC29/WG11 (MPEG). Doc. MPEG-W12035: Applications and Requirements on 3D Video Coding. Technical report, MPEG, Geneva, Switzerland, Mar. 2011.

[7] A. Vetro, T. Wiegand, and G.J. Sullivan. Overview of the stereo and multiview video coding extensions of the H.264/MPEG-4 AVC standard. *Proceedings of the IEEE*, 99(4):626 –642, Apr. 2011.

[8] System Description: Blu-ray Disc Read-Only Format Part3: Audio Visual Basic Specifications. Technical report, Blu-ray Disc Association, 2009.

[9] B. Bross, W.-J. Han, J.-R. Ohm, G.J. Sullivan, and T. Wiegand. Doc. JCTVC-H1003: High Efficiency Video Coding (HEVC) text specifi-

cation draft 6. Technical report, ITU-T and ISO/IEC, San Jose, USA, Jan. 2012.

[10] B. Li, G. J. Sullivan, and J. Xu. Doc. JCTVC-G399: Comparison of compression performance of HEVC working draft 4 with AVC High profile. Technical report, ITU-T and ISO/IEC, Geneva, Switzerland, Nov. 2011.

[11] A. Vetro. Frame compatible formats for 3D video distribution. In *Proceedings of IEEE International Conference on Image Processing (ICIP)*, pages 2405–2408, Nov. 2010.

[12] ISO/IEC JTC1/SC29/WG11 (MPEG). Doc. MPEG-N1070: Text of ISO/IEC 14496-10:2009/FDAM 1 Constrained baseline profile, stereo high profile, and frame packing arrangement SEI message. Technical report, MPEG, Lausanne, Switzerland, July 2007.

[13] A. Tourapis, P. Pahalawatta, A. Leontaris, Y. He, Y. Ye, K. Stec, and W. Husak. Doc. MPEG-M17925: A frame compatible system for 3D delivery. Technical report, MPEG, Geneva, Switzerland, July 2010.

[14] P. Merkle, A. Smolic, K. Muller, and T. Wiegand. Efficient prediction structures for multiview video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(11):1461–1473, Nov. 2007.

[15] ISO/IEC JTC1/SC29/WG11 (MPEG). Doc. MPEG-W12036: Call for Proposals on 3D Video Coding Technology. Technical report, MPEG, Geneva, Switzerland, Mar. 2011.

[16] M. Hannuksela D. Rusanovskyy. Doc. MPEG-M22552: Description of Nokia's response to MPEG 3DV Call for Proposals on 3DV Video Coding Technologies. Technical report, MPEG, Geneva, Switzerland, Nov. 2011.

[17] MPEG Video Subgroup. w12558 - Test Model for AVC based 3D video coding. Technical report, MPEG, San Jose, USA, Feb. 2012.

[18] M. Hannuksela, Y. Chen, T. Suzuki. Doc. JCT-3V D1002: 3D-AVC Draft Text 6. Technical report, ITU-T and ISO/IEC, Incheon, Korea, Apr. 2013.

[19] G. Van Wallendael, S. Van Leuven, J. De Cock, F. Bruls, and R. Van de Walle. 3D video compression based on High Efficiency Video Coding. *IEEE Transactions on Consumer Electronics*, 58(1):137 –145, Feb. 2012.

[20] R. Sjoberg, Ying Chen, A. Fujibayashi, M.M. Hannuksela, J. Samuelsson, Thiow Keng Tan, Ye-Kui Wang, and S. Wenger. Overview of HEVC high-level syntax and reference picture management. *IEEE Transactions on Circuits and Systems for Video Technology*, 22(12):1858–1870, 2012.

[21] H. Schwarz, C. Bartnik, S. Bosse, H. Brust, T. Hinz, H. Lakshman, D. Marpe, P. Merkle, K. Mller, H. Rhee, G. Tech, M. Winken, and T. Wiegand. Doc. MPEG-M22570: Description of 3D Video Coding Technology Proposal by Fraunhofer HHI (HEVC compatible, configuration A). Technical report, MPEG, Geneva, Switzerland, Nov. 2011.

[22] H. Schwarz, C. Bartnik, S. Bosse, H. Brust, T. Hinz, H. Lakshman, D. Marpe, P. Merkle, K. Mller, H. Rhee, G. Tech, M. Winken, and T. Wiegand. Doc. MPEG-M22571: Description of 3D Video Coding Technology Proposal by Fraunhofer HHI (HEVC compatible, configuration B). Technical report, MPEG, Geneva, Switzerland, Nov. 2011.

[23] MPEG Video Subgroup. w12559 - Test Model under Consideration for HEVC based 3D video coding. Technical report, MPEG, San Jose, USA, Feb. 2012.

[24] S. Van Leuven, H. Kalva, G. Van Wallendael, J. De Cock, and R. Van de Walle. Joint complexity and rate optimization for 3DTV depth map encoding. In *Proceedings of IEEE International Conference on Consumer Electronics (ICCE)*, pages 191–192, Jan. 2013.

[25] F. Bruls, G. Van Wallendael, J. De Cock, B. Sonneveldt, and S. Van Leuven. Doc. MPEG-M22603: Description of 3DV CfP submission from Philips & Ghent University IBBT. Technical report, MPEG, Geneva, Switzerland, Nov. 2011.

[26] S. Van Leuven, F. Bruls, G. Van Wallendael, J. De Cock, and R. Van de Walle. Doc. MPEG-M23669: Hybrid 3D Video Coding. Technical report, MPEG, San Jose, USA, Feb. 2012.

[27] L. B. Stelmach and W. J. Tam. Stereoscopic image coding: Effect of disparate image-quality in left- and right-eye views. *Signal Processing: Image Communication*, 14(12):111 – 117, 1998.

[28] D.V. Meegan, L.B. Stelmach, and W.J. Tam. Unequal weighting of monocular inputs in binocular combination: Implications for

the compression of stereoscopic imagery. *Journal of experimental psychology-applied*, 7(2):143–153, June 2001.

[29] L.B. Stelmach, W.J. Tam, D.V. Meegan, and A. Vincent. Stereo image quality: effects of mixed spatio-temporal resolution. *IEEE Transactions on Circuits and Systems for Video Technology*, 10(2):188–193, Mar. 2000.

[30] MPEG Video Subgroup. Doc. MPEG-W12352: Common Test Conditions for HEVC- and AVC-based 3DV. Technical report, MPEG, Geneva, Switzerland, Dec. 2011.

[31] S. Liu, F. Liu, J. Fan, and H. Xia. Asymmetric stereoscopic video encoding algorithm based on subjective visual characteristic. In *Proceedings of International Conference on Wireless Communications Signal Processing (WCSP)*, pages 1 –5, Nov. 2009.

[32] W.J. Tam, L.B. Stelmach, F. Speranza, and R. Renaud. Cross-switching in asymmetrical coding for stereoscopic video. *Stereoscopic Displays and Virtual Reality Systems IX*, 4660(2):95 –104, Jan. 2002.

[33] W.J. Tam, L.B. Stelmach, F. Speranza, and R. Renaud. Stereoscopic video: asymmetrical coding with temporal interleaving. *Stereoscopic Displays and Virtual Reality Systems VIII*, 4297(2):299 –306, Jan. 2001.

[34] K. McCann, S. Sekiguci, B. Bross, and W.-J. Han. Doc. JCTVC-E602: HEVC Test Model 3 (HM 3) Encoder Description. Technical report, MPEG and ITU-T, Geneva, Switzerland, Mar., 2011.

[35] Video Subgroup. Doc. w12561: Description of Core Experiments in 3D Video Coding. Technical report, MPEG, San Jose, USA, Feb. 2012.

[36] A. Benoit, P. Le Callet, P. Campisi, and R. Cousseau. Quality assessment of stereoscopic images. *EURASIP Journal on Image and Video Processing*, 2008(1):659024, 2008.

[37] P. Hanhart, F. De Simone, M. Rerabek, and T. Ebrahimi. Doc. m23908: 3DV: Objective quality measurement for the 2-view case scenario. Technical report, MPEG, San Jose, USA, Feb. 2012.

[38] P. Hanhart, F. De Simone, and T. Ebrahimi. Doc. m24807: 3DV: Alternative metrics to PSNR. Technical report, MPEG, Geneva, Switzerland, May 2012.

[39] P. Hanhart and T. Ebrahimi. Doc. JCTVC-A0150: 3DV: Quality assessment of stereo pairs formed from two synthesized views. Technical report, MPEG and ITU-T, Stockholm, Sweden, July 2012.

[40] ISO/IEC JTC1/SC29/WG11 (MPEG). Doc. MPEG-W12957: Call for Proposals on Scalable Video Coding Extensions of High Efficiency Video Coding (HEVC). Technical report, MPEG, Stockholm, Sweden, July 2012.

[41] ISO/IEC MPEG. w12622 - Draft requirements for the scalable enhancement of HEVC. Technical report, MPEG, San Jose, USA, Feb. 2012.

5
Conclusions

I regret only one thing, which is that the days are so short and that they pass so quickly. One never notices what has been done; one can only see what remains to be done, and if one didn't like the work it would be very discouraging.

–Marie Curie

Forecasts for the future Internet traffic show that the amount of video data will steeply increase the next couple of years. This will increase both the cost for the bandwidth as well as the energy cost for the encoding process. Therefore, intelligent systems need to be developed. This will reduce the required bandwidth for video but also the encoding complexity will be limited. Not only an improved general purpose video compression standard, High Efficiency Video Coding (HEVC), has been developed within video standard organizations, also dedicated compression architectures for specific purposes are being investigated. Two examples of such specific purposed architectures are scalable video coding and multiview video coding, which are probably the two most known extensions of H.264/AVC.

Currently, efforts are ongoing to investigate a scalable video coding extension and multiview extension for HEVC. However, the reduction of the complexity of such extensions will always be necessary to reduce both the production and the operational costs of the system. In this work, optimizations to reduce the encoding complexity for scalable video coding (SVC) as well as a multiview extension for HEVC are proposed.

In the first Chapter, an analysis was performed to identify a fast mode decision model for spatial scalable enhancement layers. Furthermore, this analysis led to generic techniques, which can be implemented in other (existing) optimization schemes. The proposed fast mode decision reduces the enhancement layer mode decision complexity. This is achieved by only evaluating modes with a high probability, given the already encoded base layer macroblock mode. Comparable compression efficiency (in terms of rate distortion performance) is achieved compared to state-of-the-art fast mode decision models. However, a significant reduction in complexity of 74.58% is achieved, while the state-of-the-art achieves only 52% in complexity. Furthermore, the proposed model operates independently of quantization, the content in the video stream, and spatial upscaling ratio of the base and enhancement layer. The independence of the model to content, quantization and resolution implies for hardware design that the complexity can be well estimated in advance.

When the complete optimized mode decision process can not be implemented, or when en existing fast mode decision model is in place, smaller optimizations on a (sub-)macroblock mode level can be applied. Therefore, three generic techniques have been proposed: *not encoding orthogonal modes*, *only evaluating sub_8×8 blocks if such sub-macroblocks are present in base layer*, and *only evaluating base layer list predictions*. These techniques are generic in a sense that they can be implemented independently of other fast mode decision models and that they can be mutually combined to improve the encoding complexity. For each technique independently, the compression efficiency and complexity reduction are reported. Almost 54% complexity reduction is achieved by not evaluating *sub_8×8* modes in the enhancement layer. If the techniques are combined, a reduction up to 77.15% in complexity is achieved for a 1.06% increase in bit rate. Furthermore, these generic techniques can be combined with existing models. If combined with Li's model, 87.27% complexity reduction is achieved, for a 2.04% BDRate reduction. Since different levels of complexity can be achieved, a scalable SVC encoder can be designed. However, the following rule of thumb is applicable: the higher the complexity reduction, the lesser the compression efficiency. Moreover, in order to further reduce the complexity, Li's model can be replaced by any complexity reduction model.

The proposed schemes achieve a low complexity for the SVC enhancement layer encoding. This opens the path to a low energy cost in SVC encoders. Furthermore it allows for a broad audience to capture content at low cost, and allows a broad range of different device types to play video content.

A considerable amount of data is encoded in H.264/AVC. However, a current trend (which according to industry reports will be extended in the future) is to play back video on different types of devices. Those devices differ in screen resolution, bandwidth requirements, battery power, processing power, memory availability and storage space. In order to allow as many as possible users to receive video content, transcoding H.264/AVC video to SVC can be performed. This transcoding should best be performed in the network. To reduce the processing power and energy cost, low complexity techniques are proposed. In Chapter 3, an optimized closed-loop transcoder is presented. This architecture reduces transcoding complexity with 91.69%. To further reduce the complexity, the closed-loop transcoder is combined with an open-loop transcoder, resulting in a hybrid transcoder. To reduce the drift effects of the open-loop transcoder, only non-referenced frames have been open-loop transcoded. By adjusting the number of open-loop transcoded frames, the transcoder becomes complexity scalable. The complexity reduction can be scaled between 91.69% and 99.28%. However, by increasing the number of open-loop transcoded frames, the base layer will have more drift and higher bit rate (which results in a lower degree of scalability).

The proposed hybrid transcoder allows a huge number of end users to access video information on different types of devices in different kinds of environments. Compared to a cascaded decoder-encoder, the complexity is reduced significantly. Moreover, the hybrid transcoder allows for a trade-off between the quality and the available resources. The transcoder can be applied in a network with a variable number of video streams. This yields a reduction of the hardware cost, as well as the energy consumption in the network. In the future, this system can be further improved and extended towards HEVC. Moreover, the complexity scalability can be refined to a macroblock level, by deciding on a macroblock basis whether or not a macroblock should be open- or closed-loop transcoded.

A hybrid 3D video compression architecture has been presented in Chapter 4. This architecture applies H.264/AVC encoding which is used as a predictor for HEVC improvement information. Either monoscopic H.264/AVC or stereoscopic MVC compatibility are supported. The former encodes the center view as an H.264/AVC single layer bitstream, while the latter encodes the center and left view as MVC bitstreams. The decoded output of these views are used as a predictor for the HEVC side views. Additionally,

an HEVC refinement might be used based on the MVC encoded left view. To indicate the value of this architecture, the hybrid architecture has been compared with currently investigated architectures within JCT-3V standardization. The hybrid architecture achieves 30% BDRate reduction over other H.264/AVC compatible systems. Compared to HEVC compatible architectures, the presented architecture shows a loss because of the center view is still H.264/AVC compatible. This results in a gain of 26.71% in BDRate for MVHEVC. By applying coding tools, other than pixel domain predictions, in the HEVC based designs, only 8% additional BDRate reduction is achieved, while the complexity increases significantly. Next to the improved compression efficiency, the benefits of this design reach further: end user equipment uses the available resources more efficiently and less HEVC silicon surface is required. The former reduces the decoding energy cost, while the latter reduces the production cost.

The proposed hybrid architecture significantly reduces the BDRate over fully H.264/AVC based architectures. Additionally, compatibility with H.264/AVC is maintained. Therefore, the hybrid 3D architecture can accommodate the adoption of 3D video, and ensure a fast market adoption of 3D video.

For the three presented domains (SVC encoding, H.264/AVC-to-SVC transcoding and hybrid 3D) efficient architectures are proposed. Given the increasing amount of video traffic over the network, the proposed architectures allow to reduce the cost for both maintenance, production and energy consumption. This allows the current system to continue the operations in the near-future. Moreover, these architectures also bridge the gap towards low-powered and low-complexity end user devices. Consequently, production and consumption of video data will become more ubiquitous.