



*Scuola Dottorale di Ingegneria
Sezione di Ingegneria dell'Elettronica Biomedica,
dell'Elettromagnetismo e delle Telecomunicazioni*

XXIII Ciclo

Codifica Distribuita di Sorgenti Video 3D

Veronica Palma

Docente-guida:

Prof. Alessandro Neri

Coordinatore:

Prof. Lucio Vegni

A.A. 2010/2011



*Scuola Dottorale di Ingegneria
Sezione di Ingegneria dell'Elettronica Biomedica,
dell'Elettromagnetismo e delle Telecomunicazioni*

XXIII Ciclo

Distributed Coding of 3D Video Sources

Veronica Palma

Advisor:

Prof. Alessandro Neri

Coordinator:

Prof. Lucio Vegni

A.A. 2010/2011

Copyright ©by Veronica Palma. All right reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy, recording, or any information storage and retrieval system, without permission in writing of the author. The University “Roma Tre” of Rome, Italy, has several rights in order to reproduce and distribute electronically this document.

Department: University “Roma Tre” of Rome, Italy

Department of Applied Electronics

Laboratory of Telecommunications (COMLAB)

PhD Thesis: Distributed Coding of 3D Video Sources

Author: **Veronica Palma**

Advisor: Prof. **Alessandro Neri** (University “Roma Tre” of Rome, Italy)

Year: 2011



This thesis describes the research carried out within the COMLAB Laboratory of the University “Roma Tre” of Rome, Italy, from 2008 to 2010.

Premessa

Codifica distribuita di sorgenti video 3D

Negli ultimi anni, diverse tecniche di codifica video hanno raggiunto un grosso successo a livello commerciale ed è ormai chiaro che i sistemi video digitali sostituiranno completamente tutti i sistemi video analogici. Le strategie di codifica video convenzionali sono basate sull'idea che è compito del codificatore calcolare le statistiche della sorgente, creando così un codificatore complesso che interagisce con un semplice decodificatore. Broadcasting, video on demand e video streaming si basano proprio su questo paradigma. La codifica video distribuita (DVC) adotta invece un concetto differente poiché sposta la complessità computazionale della parte del decoder che ha il compito di sfruttare le statistiche delle sorgenti (parzialmente o totalmente) per ottenere una compressione efficiente. Questo nuovo approccio è particolarmente ideale per tutte quelle nuove applicazioni per cui è richiesto un basso consumo di energia e di potenza come video-camere wireless, reti di sensori, acquisizione multi-view dell'immagine, etc..

Come introdotto pocanzi, la codifica video distribuita è un nuovo approccio basato su due importanti risultati della teoria dell'Informazione: il teorema di Slepian-Wolf e quello di Wyner-Ziv.

Il teorema di Slepian-Wolf fa riferimento al caso in cui due sequenze random discrete e statisticamente dipendenti, X e Y , sono codificate in maniera indipendente, a differenza delle tecniche predittive tradizionali, MPEG e ITU-T, dove le due sorgenti sono codificate insieme. Il teorema afferma che il rate minimo per codificare le due sorgenti, tra loro dipendenti, è lo stesso che si avrebbe se le due sorgenti fossero codificate unitamente, con una probabilità di errore piccola. La codifica di Slepian-Wolf viene riferita, in letteratura, come codifica di sorgente distribuita senza perdita poiché le due sorgenti, statisticamente dipendenti, sono perfettamente ricostruite al decoder unico (trascurando una arbitrariamente piccola probabilità di errore nella decodifica). Inoltre tale teorema è fortemente legato alla codifica di canale: la dipendenza fra le due sequenze, X e Y , può essere vista come un canale virtuale dove X rappresenta l'informazione originale non corrotta, mentre Y , detta *side information*, è disponibile al decoder ed è sfruttata per stimare una versione

rumorosa di X . L'errore di stima tra X e la side information può essere corretto applicando tecniche di codifica di canale alla sequenza X (generando così i bit di parità) con l'idea che al decoder Y assuma il ruolo di informazione sistematica. Tuttavia il teorema di Slepian-Wolf ha un limite molto forte poiché si riferisce ad una codifica senza perdita ed è quindi poco adatto per scenari reali infatti la codifica lossless (senza perdita), raggiunge fattori di compressione piuttosto bassi poiché non elimina l'informazione video che non è percepibile dal sistema visivo umano. Nel 1976, A. Wyner e J. Ziv hanno studiato il caso corrispondente di codifica con perdita (lossy) e ne hanno derivato il teorema di Wyner-Ziv. Tale teorema afferma che in alcune condizioni è possibile effettuare una codifica delle due sorgenti indipendente senza perdita di efficienza rispetto al caso congiunto, anche se la codifica è con perdita. È dunque possibile comprimere, secondo i teoremi di Slepian-Wolf e Wyner-Ziv, due sorgenti statisticamente dipendenti, in maniera distribuita (codificata separata, decodifica congiunta), ottenendo una efficienza di codifica pari a quella di schemi più tradizionali (codifica unita, decodifica separata).

Sebbene le fondamenta teoriche della codifica di sorgente distribuita sono state stabilite negli anni '70, solo ultimamente sono stati proposti schemi pratici di DVC. La ragione maggiore del recente sviluppo si può rintracciare nell'evoluzione ultima che ha subito la codifica di canale e in particolare nell'introduzione dei Turbo-Codici e dei codici Low-Density-Parity-Check (LDPC).

L'analisi degli aspetti basilari del DVC, il suo approccio statistico e le principali strategie pratiche portano a concludere che l'architettura DVC presenta i seguenti vantaggi:

1. Una allocazione flessibile della complessità generale del codec: infatti l'approccio DVC permette di spostare parte della complessità del codificatore al decoder. È quindi applicabile in tutti quei casi in cui il codificatore deve essere semplice, poco costoso e consumare al minimo l'energia.
2. Robustezza all'errore migliorata. È legata al fatto che i codec DVC sfruttano le proprietà statistiche piuttosto che la predizione.
3. Scalabilità indipendente del codec. Mentre negli attuali codec scalabili, c'è tipicamente un approccio predittivo dagli strati più bassi a quelli superiori, richiedendo

al codificatore di conoscere il risultato della decodifica dello strato precedente così da migliorare il successivo, nella codifica DVC non viene richiesta una conoscenza deterministica del livello precedente ma solo un modello di correlazione. Ciò implica che, per i vari strati, i codec possono essere non noti e differenti.

4. Uso della correlazione multivista. La tecnica DVC può essere estesa anche al caso multivista dove è necessario tener conto anche dell'informazione inter-vista (oltre a quella intra-vista). In questo caso, la codifica DVC introduce dei benefici significativi poiché a differenza dell'approccio convenzionale che richiede che le sequenze catturate da viste differenti siano simultaneamente disponibili dalla parte del codificatore con la conseguenza che le varie telecamere comunichino fra loro, un codificatore basato sul paradigma DVC non necessita di elaborare congiuntamente i frame appartenenti alle varie viste e né di una comunicazione tra le varie telecamere.

Di seguito vengono riportati gli argomenti trattati durante il Dottorato di Ricerca:

1. Valutazione degli artefatti video introdotti in un sistema di codifica distribuita stereoscopico, Capitolo 2;
2. Codifica distribuita "joint" sorgente-canale per sorgenti 3D, Capitolo 3;
3. Sistema di codifica distribuita multivista, Capitolo 4;
4. Trasmissione di contenuti multimediali basate sui codifici a fontana in reti MANET, Capitolo 5.
5. Ricerca di immagini in database multimediali basata sui momenti di Zernike e sui polinomi di Laguerre-Gauss, Capitolo 6.

Nel Capitolo 2 saranno valutati gli artefatti introdotti dalla codifica distribuita per sequenze stereoscopiche. L'obiettivo di questo lavoro è valutare la qualità delle immagini stereo attraverso modelli oggettivi e soggettivi e discutere i possibili artefatti introdotti da questo particolare approccio di codifica. Le valutazioni delle prestazioni di un sistema di codifica distribuito saranno confrontate con le prestazioni della codifica H.264/AVC che rappresenta ad oggi un sistema di codifica video altamente sfruttato.

Gli artefatti precepiti in sequenze video 3D non solo creano un risultato visivamente non gradevole ma allo stesso tempo creano malessere al sistema visivo umano. Per questa ragione, la comunità scientifica si sta focalizzando sulla definizione di metrica di qualità percettiva che possa quantificare la distorsione tipica introdotta in sequenze video 3D.

É quindi necessario classificare gli artefatti presenti in un contenuto stereoscopico o più generalmente 3D. Ogni fase che va dalla acquisizione, alla codifica, alla trasmissione fino alla visualizzazione di sequenze video stereo introduce artefatti tipici di quello step:

- **Acquisizione:** la maggior parte dei video 3D sono ottenuti da video-camere doppie o da configurazioni multi-vista dove ogni vista é registrata separatamente. Un altro approccio consiste nel trasformare il contenuto 2D in 3D tramite algoritmi di conversione che sfruttano le mappe di profondità. Una terza possibilità prevede l'aumento della profondità tramite la presenza di un sensore addizionale. Per tutti questi approcci, una non corretta impostazione dei parametri quali, ad esempio, la distanza fra le due video-camere, la lunghezza della lente focale o la distanza di convergenza, può creare una visione non corretta della profondità così come rumore, aliasing, e l'effetto puppet theater.
- **Rappresentazione dei dati acquisiti:** in particolare la conversione del formato da 2D a 3D può causare artefatti come il ghosting e aliasing temporale e spaziale.
- **Codifica:** le sequenze 3D sono generalmente codificate secondo schemi di codifica multi-vista o algoritmi di codifica 2D adattati per lo stereo. Questa fase può alterare importanti dettagli dell'immagine per la percezione della profondità.
- **Trasmissione:** da una parte, la perdita dei pacchetti dati e la presenza di canali rumorosi possono essere cause di una qualità percepita del contenuto 3D degradata; dall'altra parte, gli algoritmi che tentano di correggere questi errori possono a loro volta introdurre nuovi artefatti.
- **Visualizzazione:** la qualità video stereoscopica dipende fortemente dall'approccio adottato per la visualizzazione 3D, cioè dipende dagli artefatti che caratterizzano i display 3D. Effetti di flickering, cross-talk e puppet theater possono essere presenti

in questa fase.

Considerando ciò, lo scopo di questo capitolo é lo studio e l' analisi degli artefatti introdotti da una sistema di codifica stereo distribuita e valutare la qualità video attraverso metriche oggettive e soggettive. In particolare un' analisi sulla sensibilità dei parametri che controllano il bitrate verrà affrontata.

Nel Capitolo 3 viene dato spazio alla codifica congiunta sorgente-canale. Il principio di separazione sorgente-canale di Shannon afferma che é possibile ottenere prestazioni ottime adottando un approccio separato per la codifica di sorgente e per la codifica di canale.

Sulla base di questo principio, i sistemi di comunicazione moderni si sono sviluppati secondo una rigida architettura a strati per cui la codifica di sorgente é effettuata al livello applicativo mentre quella di canale a livello fisico. Se da una parte questo tipo di implementazione permette di sfruttare un design modulare, dall' altra parte ci sono casi in cui la codifica congiunta sorgente-canale può avere maggiori vantaggi.

Secondo lo schema tandem, una sorgente Wyner-Ziv passa attraverso un quantizzatore e un codificatore di Slepian-Wolf (SW). I bit risultanti vengono poi protetti tramite tecniche di codifica di canale. Tuttavia é possibile ottenere una codifica unita sorgente-canale, e a tal fine é necessario combinare due codifiche di canale, una relativa alla codifica SW and l' altra alla codifica di canale, in una singola codifica.

É possibile considerare per la sorgente X , il problema della codifica su due canali.

- Il primo canale é il canale rumoroso attraverso cui passano i bit sorgente-canale e rappresenta la distorsione subita dai bit di parità.
- Il secondo é il canale virtuale di correlazione tra la sorgente e la side information disponibile al decoder e rappresenta la distorsione dei bit sistematici.

In questo capitolo viene proposta un' analisi dello stato dell' arte dei sistemi DVC che introducono una codifica congiunta sorgente-canale e viene presentato un modello di codifica distribuita sorgente-canale per video 3D basato sui turbo-codici che preservi la qualità visiva percepita e allo stesso tempo mantenga una bassa complessità computazionale.

Dopo aver valutato la codifica distribuita per doppia sorgente e quindi analizzare modelli per sequenze stereoscopiche, il passo successivo prevederà l' introduzione di più di due sorgenti e cioè lo studio sistemi di codifica distribuita multi-vista.

Negli ultimi anni, i sistemi video multivista sono diventati sempre più popolari grazie alla presenza di applicazioni multimediali ed interattive, come ad esempio la TV 3D, o scenari con reti di sensori wireless. Inoltre, la grossa diffusione di smart phone corredati di videocamere ad alta definizione e la disponibilità di connessione 3G, come HSPA e LTE, è uno dei fattori chiave per la co-creazione di contenuto multimediale per applicazioni a valore aggiunto.

Tuttavia, l' impiego di camere multivista aumenta la quantità di dati da elaborare. La compressione dei dati diventa quindi, in tali sistemi, un fattore estremamente importante. Rispetto a codec tradizionali, un approccio di codifica distribuita multi-vista ha i seguenti vantaggi:

- Non è necessario che le video-camere comunichino fra loro, a differenza delle codifiche multi-vista convenzionali dove la correlazione inter-vista viene calcolata dalla parte dell' encoder. In scenari reali risulta difficile scambiare una tale quantità di dati e la codifica distribuita si propone come una soluzione molto interessante, soprattutto quando si lavora con un sistema composto da un alto numero di video-camere.
- La bassa complessità computazionale permette di trasmettere i dati video con un basso ritardo. In un sistema di codifica distribuita, la complessità computazionale è spostata dalla parte del decoder permettendo così che il codificatore abbia un design leggero e semplice mentre la complessità del decoder non è una questione fondamentale in uno scenario DVC.
- La selezione delle viste che devono essere codificate è più flessibile. In approcci convenzionali, i frame di riferimento sono predefiniti durante tutta la codifica. Tutti i frame di riferimento devono essere decodificati in anticipo rispetto al frame corrente. Invece, nel nostro caso, questa ridondanza può essere evitata poiché la predizione inter-viste viene fatta al decoder e la decodifica delle viste differenti viene scelta liberamente.

Considerato ciò, il Capitolo 4 è dedicato allo studio di sistemi di codifica video distribuita multivista. In particolare grande attenzione verrà data alla generazione delle side information e della fusione dell' informazione temporale proveniente fra frame della stessa telecamera e dell' informazione spaziale proveniente da frame di telecamere diverse.

Nel Capitolo 5, l' attenzione è rivolta alla realizzazione di un sistema di trasmissione multicast di contenuto multimediale in una rete MANET.

L' idea è di proporre un sistema di codifica unita sorgente-canale basata sui codici LT che permette di fornire un servizio per contenuti multimediali che sia al tempo stesso affidabile e real-time. Lo scenario considerato è una rete MANET (Mobile Ad-hoc NETWORK) dove sono presenti sensori wireless distribuiti sul territorio e che possono muoversi.

Con rete MANET si intende un sistema autonomo e mobile composto da router e da host legati tramite distribuzioni wireless arbitrarie. La posizione dei router e degli host può cambiare continuamente e in modo imprevedibile.

Gli elementi caratterizzanti una rete MANET sono l' assenza di una infrastruttura dedicata, la presenza di nodi mobili in grado di auto-configurarsi e la presenza di link a bassa capacità e fragili.

Da una parte lo sviluppo di reti ad-hoc può essere portato avanti rispetto alle variazioni dei requisiti grazie alla loro proprietà di scalabilità; dall' altra parte, è necessario fare i conti con prestazioni ridotte dovute a tecniche di routing multi-hop e ad un controllo distribuito. Non solo, ma la presenza di link instabili e la scarsa qualità del canale wireless, pone una sfida ai tradizionali schemi di routing. In questo contesto, i codici LT rappresentano una valida soluzione per trasmissione dati su reti a perdita di pacchetti.

In questo capitolo, viene analizzato l' uso dei codici LT per un sistema di codifica joint sorgente-canale in trasmissioni a perdita di pacchetti caratterizzate dalla mobilità. Più specificatamente, consideriamo uno scenario MANET con nodi che si muovono randomicamente e una singola sorgente che trasmette dati multimediali a N nodi riceventi. Per il rilancio delle informazioni viene impiegato PUMA come algoritmo di routing.

PUMA (Protocol for Unified Multicast Announcement) è un algoritmo che fa routing in reti MANET e che trasmette pacchetti in flooding, inondando cioè la rete. Si basa su

un approccio multipath tra il router e un nodo eletto, detto *core*, della rete mesh ed è caratterizzato da una alta robustezza alle perdite e ai guasti dei collegamenti.

Ogni nodo trasmittente invia pacchetti dati attraverso il percorso più breve e quando un pacchetto dati deve raggiungere una rete mesh, l'informazione viene mandata in flooding e ogni nodo mantiene aggiornata una propria cache con l'identificativo del pacchetto con il fine di buttare i duplicati.

Quando un flusso informativo, protetto dai codici LT, viene inviato al nodo destinatario intermedio, quest'ultimo deve collezionare una quantità minima di pacchetti, eventualmente inviati da differenti vicini ad un hop, che permette la decodifica LT e può ricostruire l'intero flusso informativo. A questo punto, il nodo codifica il flusso ricostruito sulla base dei codici LT. L'ordine dei pacchetti viene randomizzato sulla base di un metodo di scrambling e i pacchetti sono ritrasmessi nella sotto-rete successiva.

Il Capitolo 6 è incentrato sullo studio di algoritmi di riconoscimento di oggetti all'interno di immagini presenti in database (image retrieval). Le funzionalità base di un servizio Internet basato sul contenuto sono l'invio e la trasmissione del contenuto insieme alla ricerca di contenuti che può avvenire tramite utente o tramite dispositivo di ricerca. Inizialmente gli algoritmi di ricerca si basavano sull'uso di metadati che descrivevano il contenuto semantico estratto da un dato contenuto multimediale, anche attraverso procedure manuali. Tuttavia i futuri servizi Internet richiederanno sempre di più funzionalità atte all'ispezione, al riconoscimento, alla categorizzazione e indicizzazione del contenuto multimediale che richiedano il minimo intervento umano. Da qui la necessità di implementare algoritmi che siano al tempo stesso veloci ed affidabili, in grado di localizzare ed inseguire oggetti complessi all'interno di una scena indipendentemente dall'orientazione e dalla scala. Diverse tecniche sono state adottate e tra queste le più efficaci risultano quelle basate su invarianti che permettono la rappresentazione di template invariante rispetto alla scala e alla rotazione. Un vettore di feature invariante rispetto alla scala e alla rotazione viene estratto dall'oggetto complesso che si vuole indagare. Per ciascun punto, la somiglianza tra il template e il vettore di feature è calcolata e viene estratto il massimo. Queste tecniche differiscono per la scelta degli invarianti e tra queste quella basata sui momenti di Zernike risulta essere molto interessante per la buona performance totale.

Per la localizzazione di oggetti all' interno di immagini saranno utilizzate due tecniche basate entrambe sulla decomposizione Quadtree e applicando in un primo caso le funzioni di Laguerre-Gauss e nell' altro caso l'uso dei momenti di Zernike. Sia l' immagine template sia l' immagine di riferimento sono decomposti nelle basi delle funzioni di riferimento (funzioni di Laguerre-Gauss o funzioni di Zernike). L' immagine template é analizzata secondo una procedura a blocchi tramite decomposizione quadtree. La ricerca di un pattern complesso in un database multimediale é basata su una procedura sequenziale che verifica se ogni immagine candidata contiene ciascun quadrato della lista quadtree ordinata e migliorando, passo dopo passo, la stima della posizione, dell' orientazione e della scala (caso Laguerre-Gauss).

I momenti di Zernike si ottengono decomponendo il template nel dominio delle funzioni circolari armoniche (CHF), e sono definiti su un disco di raggio unitario. Le proprietà delle CHF permettono di calcolare un pattern semplicemente moltiplicando i coefficienti dell' espansione per un fattore esponenziale complesso la cui fase é proporzionale all' angolo di rotazione. Di conseguenza, gli invarianti alla rotazione possono essere facilmente ottenuti considerando l' ampiezza dei coefficienti dell' espansione.

Nelle metodologie proposte nel dominio di Zernike e nel dominio di Laguerre-Gauss, viene selezionato un cerchio contenente l' oggetto da localizzare e la porzione di pattern che cade all' interno del cerchio approssimata da una versione troncata dei coefficienti del polinomio considerato (sia esso Zernike o Laguerre-Gauss) fino ad un dato ordine. Il matching tra le immagini é calcolato tramite funzionale di verosimiglianza che in questo caso é espresso in termini di coefficienti dei momenti di Zernike o coefficienti della trasformata di Laguerre-Gauss. Molte applicazioni richiedono il riconoscimento e la localizzazione di pattern complicati che devono essere distinti da oggetti simili che differiscono per pochi dettagli. In questa situazione l' uso dei momenti di Zernike per il calcolo del funzionale di verosimiglianza richiede un gran numero di termini. Per localizzare oggetti di forma arbitraria e al contempo ridurre il carico computazionale, il pattern viene partizionato in blocchi usando la decomposizione quadtree. La grandezza di ogni blocco é adattata all'oggetto da analizzare ed é controllata dalla norma quadrata dell' errore corrispondente all' espansione troncata dei momenti di Zernike o dei coefficienti di Laguerre-Gauss. I

blocchi quadtree sono poi ordinati rispetto all' energia del gradiente filtrato con passabasso.

Per il procedimento basato sui momenti di Zernike, la stima a massima verosimiglianza della posizione e dell' orientazione del primo blocco é calcolata attraverso una procedura iterativa di quasi-Newton. Rispetto alle tradizionali tecniche di massima verosimiglianza basate sul matching di una immagine candidata con l' intero set di pattern ruotati, questo procedimento richiede la massimizzazione locale del funzionale derivato dai coefficienti di Zernike. La posizione e l' orientazione stimate sono poi utilizzate per verificare se l' immagine corrente contiene o meno il secondo blocco della lista quadtree ordinata. La procedura viene ripetuta iterativamente e finisce quando tutti i blocchi della lista sono stati processati o in alternativa quando l' energia della differenza eccede una determinata soglia.

Rispetto al caso dei momenti di Zernike, sfruttando la trasformata di Laguerre-Gauss é possibile tramite funzionale di verosimiglianza stimare posizione, rotazione dell' immagine ma anche la scala. Anche in questo caso, il funzionale é applicato ai coefficienti dell' espansione permettendo cosí un costo computazionale ridotto.

Entrambe le tecniche permettono la ricerca di immagini all' interno di ampi database con un successo superiore ai metodi già esistenti in letteratura. Permettono altresí la possibilità di individuare esattamente l' immagine cercata a partire da una regione di interesse di riferimento, e di stimarne posizione, orientazione e scala, a differenza dello stato dell' arte dove viene individuata la classe di appartenenza dell' immagine e non tanto l' immagine stessa.

Contents

Premessa	i
Abstract	1
1 Distributed Video Coding	4
1.1 Introduction	4
1.2 Theoretical Background	5
1.2.1 Wyner-Ziv theorem	12
1.3 State of Art	13
1.4 The Considered DVC Architecture	15
1.4.1 Transformation	17
1.4.2 Quantization	18
1.4.3 Slepian-Wolf Encoder	18
1.4.4 Parity bit Request Channel	20
1.4.5 Side Information Creation	21
1.4.6 Slepian-Wolf Decoder	24
1.4.7 Reconstruction	27
1.5 Application Scenarios for DVC	27
1.5.1 Wireless Video Cameras	28
1.5.2 Wireless Low-Power Surveillance	30
1.5.3 Mobile Document Scanner	30
1.5.4 Video Conferencing with Mobile Devices	31
1.5.5 Distributed Video Streaming	32

1.5.6	Multiview Video Entertainment	32
2	Stereo Video Artifacts in a Distributed Coding Approach	34
2.1	Introduction	34
2.2	Artifacts introduced in stereo video coding	37
2.2.1	Artifacts in image structure	39
2.2.1.1	Blocking effect	39
2.2.1.2	Blurring effect	39
2.2.1.3	Ringling effect	40
2.2.1.4	Staircase effect	40
2.2.1.5	Mosaic pattern effect	40
2.2.2	Artifacts in Image Color	41
2.2.2.1	Color Bleeding	41
2.2.3	Artifacts related to motion	41
2.2.3.1	Mosquito Noise	41
2.2.3.2	Judder	41
2.2.4	Binocular Artifacts	42
2.2.4.1	Cross-distortion	42
2.2.4.2	Cardboard effect	42
2.2.4.3	Depth Bleeding	42
2.3	Quality metrics	42
2.3.1	Objective Quality Evaluation	43
2.3.1.1	PSNR	44
2.3.1.2	SSIM	45
2.3.1.3	VQM	45
2.3.2	Subjective Video Quality Measurements	46
2.3.2.1	Mean Opinion Score (MOS)	47
2.4	Experimental Results	47
2.5	Conclusions	52

3	Distributed Joint Source-Channel Coding	53
3.1	Introduction	53
3.2	Theoretical Background	54
3.2.1	Distributed Joint Source-Channel Coding	55
3.3	Related Works	57
3.4	Distributed Joint Source-Channel Coding for 3D Videos	59
3.4.1	Turbo codes	60
3.4.2	Joint Source-Channel Decoding	62
3.5	Experimental Results	63
3.6	Conclusions	65
4	Multiview Distributed Video Coding	67
4.1	Introduction	67
4.2	Related Works	69
4.3	Side Information Techniques	70
4.3.1	Multiview Motion Estimation (MVME)	70
4.3.2	Side Information with encoder driven fusion	72
4.3.3	Side Information with Motion Compensated Temporal Interpolation and Homography Compensated Inter-view Interpolation	72
4.3.4	View Synthesis	73
4.4	The proposed method with only one WZ camera	74
4.5	Multi-view Side Information Creation	75
4.5.1	Temporal information in Zernike domain	76
4.5.2	Spatial Information	78
4.5.3	Fusion scheme	80
4.6	Experimental Results	81
4.7	Proposed Approach with all WZ Cameras	82
4.8	Experimental Results	85
4.9	Conclusions	86

5	Fountain Code based AL-FEC for Multicast Services in MANETs	88
5.1	Introduction	88
5.2	MANET	90
5.3	Puma Protocol	92
5.4	LT code	94
5.5	The proposed approach	96
5.6	Experimental Results	97
5.6.0.1	First Scenario (no PLR)	99
5.6.0.2	Second scenario (with PLR)	102
5.6.0.3	Third Scenario (Increased number of pkts and with PLR)	102
5.6.1	Remarks	106
5.7	Conclusion	106
6	Image Search	108
6.1	Introduction	108
6.2	Zernike polynomial expansion	111
6.3	The Proposed approach in Zernike domain	114
6.3.1	The quadtree decomposition	114
6.3.2	Rotation and Location Estimation Procedure	119
6.4	Experimental Results	122
6.4.1	Experiments and performance evaluation	122
6.4.2	Comparison with other methods	123
6.4.3	Computational complexity	124
6.5	The proposed method in Laguerre-Gauss domain	128
6.5.1	Laguerre-Gauss Transform	128
6.5.2	Maximum Likelihood Localization	130
6.5.3	Quadtree Decomposition	132
6.6	Experimental Results	134
6.7	Conclusions	136
A	Maximum Likelihood Estimation in Zernike domain	139

List of Figures

1.1	Distributed source coding.	6
1.2	Admissible rate "Slepian-Wolf Region"	11
1.3	Lossy compression with side information	12
1.4	Stereo video coder architecture.	15
1.5	Turbo encoder structure	19
1.6	A pseudo-random interleaver with $L = 8$	20
1.7	Side information creation as merging of disparity and temporal motion es- timation for stereoscopic video sequence	22
1.8	Stereo side information generation architecture using a mask-based fusion approach.	24
1.9	Turbo Decoder scheme.	25
1.10	wireless camera and monitor	29
1.11	Traffic management center at Tokyo	29
1.12	Document scanning with mobile phone	31
1.13	Video Streaming solution over Internet	32
1.14	Free viewpoint Television scheme	33
2.1	Data flow of 3D TV	36
2.2	Artifacts affecting various stage of 3D video delivery	38
2.3	MOS scores for perceived stereo video quality.	48
2.4	RD performance by PSNR evaluation. PSNR is averaged on the whole right sequence.	49

2.5	RD performance by VQM evaluation. VQM is averaged on the whole right sequence.	49
2.6	RD performance by SSIM evaluation. SSIM is averaged on the whole right sequence.	50
3.1	Channel model with uncoded side information	54
3.2	The system model for Distributed Joint Source-Channel Coding	56
3.3	Achievable rate region defined by Slepian-Wolf bounds	57
3.4	Turbo encoder structure in DVC approach.	61
3.5	Turbo decoder structure in DVC approach.	61
3.6	parallel channel model for DJSCC scheme	63
3.7	Rate-distortion comparison with different schemes.	64
3.8	SSIM comparison with different schemes	64
3.9	RD comparisons by PSNR evaluations between our proposed method and conventional H.264 coding. PSNR is averaged on the whole right sequence.	65
3.10	RD comparisons by SSIM evaluations between our proposed method and conventional H.264 coding. SSIM is averaged on the whole right sequence.	65
4.1	General scheme of multi-view distributed video acquisition	68
4.2	4 different paths obtained with two H.264 cameras and two reference frames in each H.264 cameras.	71
4.3	Motion estimation and disparity estimation.	71
4.4	fusion scheme at the decoder for [1]	73
4.5	Multi-view distributed video coder architecture	76
4.6	Multi-view distributed video coder architecture	78
4.7	spatial side information based on homography	79
4.8	Fusion scheme of side information	80
4.9	RD performance by PSNR evaluation. PSNR is averaged on the whole central camera sequence.	82
4.10	RD performance by SSIM evaluation. SSIM is averaged on the whole central camera sequence.	83

4.11	RD comparison between the proposed method and the state of art.	84
4.12	Multiview scheme with frame repartition. WZ frame and KF frame are alternated for each camera.	84
4.13	Distributed multiview video coding.	85
4.14	PSNR evaluations of the proposed method respect to the state of art. . . .	86
4.15	RD by SSIM evaluation. The SSIM has been averaged on the all video sequence.	87
5.1	Example of MANET applications	91
5.2	Mobile Ad-hoc NETwork	92
5.3	Example of network composed by three main sub-layers used for simulations.	97
5.4	Comparison of the arrival time of $k(1+\varepsilon)$ packets between network that performs scrambling and the case where scrambling is not performed for the second step.	100
5.5	Comparison of the arrival time of $k(1+\varepsilon)$ packets between network that performs scrambling and the case where scrambling is not performed for 3A sub-network of the third step.	100
5.6	Comparison of the arrival time of $k(1+\varepsilon)$ packets between network that performs scrambling and the case where scrambling is not performed for 3B sub-network of the third step.	101
5.7	Comparison of the arrival time of $k(1+\varepsilon)$ packets between network that performs scrambling and the case where scrambling is not performed for the second step.	103
5.8	Comparison of the arrival time of $k(1+\varepsilon)$ packets between network that performs scrambling and the case where scrambling is not performed for 3A sub-network of the third step.	103
5.9	Comparison of the arrival time of $k(1+\varepsilon)$ packets between network that performs scrambling and the case where scrambling is not performed for 3B sub-network of the third step.	103
5.10	Arrival times comparison at different PLRs in network step 2.	104

5.11	Arrival times comparison at different PLRs in network 3A of the third step.	104
5.12	Arrival times comparison at different PLRs in network 3B of the third step.	104
5.13	Comparison on arrival times for 10000 packets between network that performs scrambling and the case where scrambling is not performed for the second step.	105
5.14	Comparison on arrival times for 10000 packets between network that performs scrambling and the case where scrambling is not performed for 3A sub-network of the third step.	105
5.15	Comparison on arrival times for 10000 packets between network that performs scrambling and the case where scrambling is not performed for 3B sub-network of the third step	106
6.1	Zernike filters with order up to $n = 3$ and $m = 3$, real part and imaginary part, respectively.	113
6.2	The architecture of the proposed image retrieval system. First, salient points are extracted by means of Harris detector and a Zernike moments quadtree decomposition is applied. Then a sequential detection and estimation procedure is performed to retrieve the candidate image inside the database.	114
6.3	Example of how an image can be split in more blocks according to a quadtree decomposition.	115
6.4	Example of quadtree decomposition by means of Zernike moments computation on Lena image.	116
6.5	In this Figure, the original image (a), the obtained neighborhoods with Zernike expansions (b) and the reconstructed neighborhoods (c) of the salient points are shown. The size of the neighborhoods is chosen according to the quadtree decomposition.	118
6.6	Examples of images for each categories present in the database <i>COREL-1000-A</i>	126
6.7	Three database samples with different orientations	136

6.8	Laguerre-Gauss Likelihood map of "Einstein" image	137
-----	---	-----

List of Tables

2.1	Objective/subjective results for a low bit-rate channel.	51
5.1	Decoding time in the first step	100
5.2	Encoding time in the second step	100
5.3	Decoding time in the second step	101
5.4	Encoding time for 3A in the third step	101
5.5	Decoding time for 3A in the third step	102
5.6	Encoding time for 3B in the third step	102
5.7	Decoding time for 3B in the third step	102
6.1	Performance results of the proposed method.	123
6.2	Average retrieval precision of the proposed ZM method compared with conventional (global and regional search, color-texture, and SIMPLicity) and recent methods (Genetic Algorithm GA). The simulations have been tested on <i>COREL-1000-A Database</i>	124
6.3	Number of the additions and multiplications performed during the comparison between the query image and the image in the DB.	127
6.4	Angle and scale estimate error for some images from the database	135

Abstract

Multimedia communication over wireless networks has generated a lot of research interests in the last years. Limited network bandwidth and the requirement of real-time playback on one hand, and severe impairments of wireless links on the other represent the main challenge. The additional issue has to do with the time-varying nature of wireless links and network heterogeneity, which make the channels between the sender and the clients extremely diverse in their available bandwidths and packet loss ratios. These diverse transmission conditions and bandwidth scarcity require an efficient scalable multimedia compression. Therefore, a robust scalable video coder is needed. Although standard video coders (e.g., H.264) can offer high coding efficiency in the scalable mode, they are very sensitive to packet loss, which results in error propagation. Motivated by its potential applications in distributed sensor networks, video coding, and compressing multi-spectral imagery, there has been a flurry of recent research activities on distributed source coding. Distributed video coding (DVC) has been proposed as a promising new technique because it adopts a completely different coding concept respect to conventional codec shifting the complexity to decoder who has the task to exploit - partly or wholly - the source statistics to achieve efficient compression. This change of paradigm also moves the encoder-decoder complexity balance, allowing the provision of efficient compression solutions with simple encoders and complex decoders. This new coding paradigm is particularly suitable for emerging applications such as wireless video cameras and wireless low-power surveillance networks, disposable video cameras, certain medical applications, sensor networks, multi-view image acquisition, networked camcorders, etc., i.e. all those devices that require low-energy or low-power consumption.

As mentioned above, Distributed Video Coding is a new video coding approach based

on two major Information Theory results: the Slepian-Wolf and Wyner-Ziv theorems. The Slepian-Wolf theorem and the Wyner-Ziv theorem state that it is possible to separately encode and jointly decode two different sources obtaining a perfect reconstruction at the decoder. The compression efficiency is comparable to conventional predictive coding systems.

Although the theoretical foundations of distributed video coding have been established in the 1970s, the design of practical DVC schemes has been proposed only in recent years. A major reason behind these latest developments is related to the evolution of channel coding, in particular Turbo and Low-Density Parity-Check (LDPC) coding, which allow to build the efficient channel codes necessary for DVC.

DVC approach can be very interesting when dealing with 3D video source both for stereoscopic video sequence and multi-view video sequence because it allows to design a simple encoder shifting all the computational complexity to the decoder. In this way, multiple cameras do not need to communicate because respect to conventional codec where inter-view and intra-view prediction is accomplished at the encoder, here inter-view and intra-view data are exchanged at the decoder.

When dealing with stereoscopic sequences, it is important to take into account all the possible artifacts that corrupt the coding phase. At this aim, an investigation on stereoscopic artifacts and video quality of a 3D distributed video coding system is carried out in this thesis. DVC video quality is estimated by means of subjective and objective evaluations. Then two different techniques for joint source-channel coding in distributed environments are introduced. The first is strictly related on distributed 3D video coding and it is based on turbo code. The second approach considers ad-hoc network with mobile and distributed nodes that acquire multimedia contents and exploit a joint source-channel coding system based on LT code for channel protection and information relaying.

Then, a multi-view distributed video coding system based on Zernike moments is analyzed. Specifically a new fusion technique between temporal and spatial side information in Zernike Moments domain is proposed. The main goal of our work is to generate at the decoder the side information that optimally blends temporal and interview data. Multi-view distributed coding performance strongly depends on the side information quality built at

the decoder. At this aim for improving its quality a spatial view compensation/prediction in Zernike moments domain is applied. Spatial and temporal motion activity have been fused together to obtain the overall side-information. The proposed method will be evaluated by rate-distortion performances for different inter-view and temporal estimation quality conditions. Finally, image retrieval techniques in multimedia database are reported. Two methods based on Zernike moments and Laguerre-Gauss Transform are proposed and compared with the state of art.

Chapter 1

Distributed Video Coding

1.1 Introduction

Implementations of current video compression standards, e.g. ISO MPEG schemes or ITU-Recommendations H.26X require a more computational cost at the encoder than the decoder; typically the encoder is 5-10 times more complex than decoder. This asymmetry can be exploited in several scenarios as broadcasting or streaming video on demand systems where video is compressed once and decompressed many times. However, many systems require the opposite conditions i.e. low-complexity encoders at the expense of high-complexity decoders due to a power/processing limited systems. It is normally assumed that the receiver can run a more complex decoder but when the receiver is another complexity-constrained device, a more powerful video transcoder somewhere on the network can be used. The research developments in distributed source coding theorem suggest that efficient compression can be achieved by exploiting source statistics partially or wholly at the decoder only. These theorems are referred as Slepian-Wolf (SW) theorem for distributed lossless coding and Wyner-Ziv (WZ) theorem for distributed lossy coding with side information at the decoder. Based upon these two theorems, distributed video coding (DVC) devotes to offer the solutions for above suggested architectures. Particularly, Wyner-Ziv video coding, a practical case of DVC based on applying WZ theorem in real video coding has been extensively studied. The most attractive advantage of WZ video coding algorithm is that it moves the computation burden from the encoder to the

decoder but the compression efficiency still can be achieved by performing joint decoding at the decoder.

The Wyner-Ziv theory suggests that unconventional video coding system, which encodes individual frames independently but decodes them conditionally, is feasible. Such systems may achieve a performance that is closer to conventional interframe coding (MPEG) than to conventional intra-frame coding (Motion-JPEG).

Wyner-Ziv video codec has a great cost advantage because it compresses each video frame by itself, thus requiring only intraframe processing. The corresponding decoder in the fixed part of the network has to exploit the statistical dependence between frames, by much more complex interframe processing. Although numerous research achievements around WZ video coding have been made in last few years, the compression performances of WZ video coding still cannot match predictive video coding so far. DVC is still far from mature to be commercialized. There is still a wide space in the DVC field that needs to be explored in the future.

1.2 Theoretical Background

SW defined the Distributed Source Coding (DSC) problem of coding correlated sources as illustrated in Figure 1.1. Let us consider two correlated information sources that are obtained from a bivariate distribution $p(x, y)$, [2]. Encoders for the two sources, X and Y , operate without knowledge of the other, while the decoders have full information on both encoded message streams. We want to determine the minimum number of bits per source character required for the two encoded message streams that assures accurate reconstruction by the decoder of the two outputs. We know that when we encode a source X , a rate $R \geq H(X)$ is sufficient for accurate reconstruction of X at the decoder. Now, suppose we deal with two sources $(X, Y) \sim p(x, y)$, then a rate $H(X, Y)$ is sufficient if we are encoding them together.

Consider the scenario where X and Y have to be encoded separately. In this case, a rate $R = R_X + R_Y \geq H(X) + H(Y)$ is sufficient. Slepian and Wolf, however, went on to show that a rate $R \geq H(X, Y)$ would be sufficient to accurately reconstruct both X and Y at

the decoder.

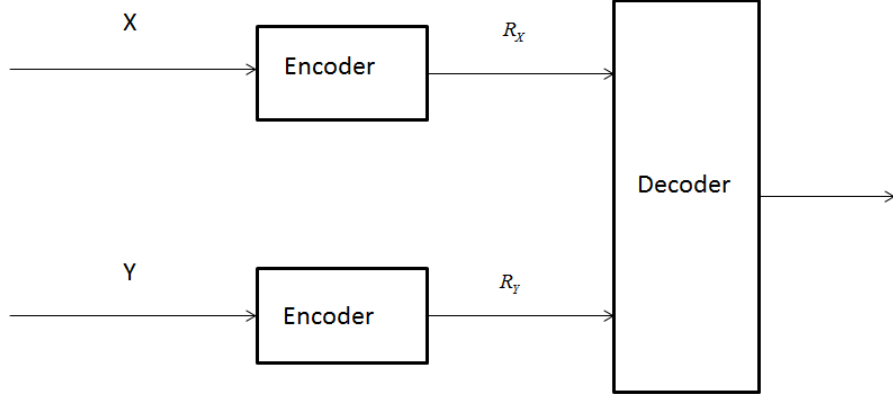


Figure 1.1: Distributed source coding.

First, let us briefly review some results for a single source that have long been known, [3]. Let X be a discrete random variable taking values in the set $\Gamma = \{1, 2, \dots, \Gamma\}$. Denote the probability distribution of X by $p_X(x) = \Pr[X = x], x \in \Gamma$. Now, let $\mathbf{X} = (X_1, X_2, \dots, X_n)$ be a sequence of n independent realizations of X so that the probability distribution for the random n -vector \mathbf{X} is given by:

$$p_X(x) = \Pr[X = x] = \prod_{i=1}^n p_X(x_i) \quad (1.1)$$

$$x = (x_1, x_2, \dots, x_n) \in \Gamma^n, x_i \in \Gamma, i = 1, 2, \dots, n$$

\mathbf{X} can be seen as a block of n successive characters from the output of an information source producing characters independently with letter distribution $p_X(x)$. In a typical long block, we have letter 1 occurring $np_X(1)$ times, letter 2 occurring $np_X(2)$ times etc. The probability of such a long typical sequence is, therefore,

$$\begin{aligned} p_T &= p_X(1)^{np_X(1)} \dots p_X(\Gamma)^{np_X(\Gamma)} \\ &= \exp[np_X(1) \log p_X(1)] \dots \exp[np_X(\Gamma) \log p_X(\Gamma)] \\ &= \exp[-nH(X)] \end{aligned} \quad (1.2)$$

where

$$H(X) = - \sum_{i=1}^{\Gamma} p_X(i) \log p_X(i) \quad (1.3)$$

is referred as the entropy of the source X . We define these $N_T = \exp[nH(X)]$ to be the typical sequences and each of these typical sequences is equally likely and occur with probability $\exp[nH(X)]$. Hence, we can transmit the source information over the channel with a rate $R = H(X)$ that assures an accurate reconstruction at the decoder. This suggests that we can accurately transmit the output of the source information using only $R = (1/n) \log N_T = H(X)$ natural bits (nats) of information per character and that allows accurate recovery of the source output.

A rate R is called *admissible* if for every $\varepsilon > 0$ there exist for some $n = n(\varepsilon)$ an encoder $E(n, \lfloor \exp(nR) \rfloor)$ and a decoder $D(n, \lfloor \exp(nR) \rfloor)$ such that $\Pr[X^* \neq X] < \varepsilon$. Otherwise R is called *inadmissible*.

The Slepian-Wolf theorem can now be analyzed in detail.

Theorem 1: If $R > H(X)$, R is admissible, if $R < H(X)$, R is inadmissible. In the latter case, there exists a $\delta > 0$ independent of n such that for every encoder-decoder pair $E(n, \lfloor \exp(nR) \rfloor)$, $D(n, \lfloor \exp(nR) \rfloor)$, $\Pr[X^* \neq X] > \delta > 0$. The theorem states that for $\eta > 0$, one can achieve arbitrarily small decoding error probability with block codes transmitting at a rate $R = H(X) + \eta$; block codes using a rate $R = H(X) - \eta$ cannot have arbitrarily small error probability. Hence, if the rate of the code is greater than the entropy, the probability of error is arbitrarily small and the information sequence is efficiently decoded at the receiving end.

Theorem 2: For the distributed source coding problem for the source (X, Y) drawn i.i.d $\sim p(x, y)$, an achievable rate point is given by:

$$\begin{aligned} R_X &= H(X|Y) + \varepsilon_x, \varepsilon_x > 0 \\ R_Y &= H(Y) + \varepsilon_y, \varepsilon_y > 0 \end{aligned} \quad (1.4)$$

The feasibility of the rates in the Slepian-Wolf theorem can be proofed introducing a new coding theorem based on random bins. The underlying idea regarding these random bins is very similar to hash functions, i.e., we choose a large random index for each source sequence. If the number of these typical sequences is small enough, then with high probability, different source sequences will have different indices, and we can reconstruct the

source sequence from the index.

The procedure for the random binning follows these rules: for each sequence X^n , an index randomly chosen from $\{1, 2, \dots, 2^{nR}\}$ is given. The set of sequences that have the same index forms a bin. For decoding the source from the bin index, we look for a typical X^n sequence in the bin. If there is one and only one typical sequence in the bin, we declare it to be the estimate of the source sequence; otherwise, there is an error. In practice, if there is more than one typical sequence in this bin, an error is present. If the source sequence is non-typical, then there will always be an error. The probability of error is arbitrarily small for sufficient R .

Consider the encoding and decoding problem for a single source. The proof for the above coding scheme producing an arbitrarily small probability of error for $R > H(X)$ is as follows:

$$\begin{aligned}
P_e^{(n)} &= P[g(\mathbf{X}) = X] \\
&= P\left[(\mathbf{X} \notin A_\varepsilon^{(n)}) \cup (f(\mathbf{X}') = f(\mathbf{X}); (\mathbf{X}', \mathbf{X}) \in A_\varepsilon^{(n)}, \mathbf{X}' \neq \mathbf{X})\right] \\
&\leq P\left[\mathbf{X} \notin A_\varepsilon^{(n)}\right] + \sum_x P\left[\exists x' \neq x : x' \in A_\varepsilon^{(n)}, f(x') = f(x)\right] p(x) \\
&\leq \varepsilon + \sum_x \sum_{x' \in A_\varepsilon^{(n)}, x' \neq x} P(f(x') = f(x)) p(x) \\
&= \varepsilon + \sum_{x' \in A_\varepsilon^{(n)}} 2^{-nR} \sum_x p(x) \\
&\leq \varepsilon + 2^{-nR} 2^{n(H(X) + \varepsilon)} \\
&\leq 2\varepsilon
\end{aligned} \tag{1.5}$$

The basic idea of the proof is to partition the space of \mathbf{X}^n into 2^{nR_X} bins and the space of \mathbf{Y}^n into 2^{nR_Y} bins.

Random code generation: Independently assign every $x \in \mathbf{X}^n$ to one of 2^{nR_X} bins according to a uniform distribution on $\{1, 2, \dots, 2^{nR_X}\}$. Similarly, randomly assign every $y \in \mathbf{Y}^n$ to one of 2^{nR_Y} bins. f_1 and f_2 are assigned to both the encoders and the decoder.

Encoding: Source 1 sends the index of the bin to which \mathbf{X} belongs and source 2 sends the index of the bin to which \mathbf{Y} belongs.

Decoding: Given the index pair (i_0, j_0) , declare $(\hat{x}, \hat{y}) = (x, y)$, if there is one and only

one pair of sequences (xy) such that $f_1(x) = i_0, f_2(y) = j_0$ and $(x, y) \in A_\varepsilon^{(n)}$. Otherwise, declare an error.

The set of X sequences and the set of Y sequences are divided into bins such a way that the pair of indices specifies a product bin. Having done this, the probability of error at the decoder is defined as the union of the following events:

$$\begin{aligned} E_0 &= \{(X, Y) \notin A_\varepsilon^{(n)}\} \\ E_1 &= \{\exists x' \neq X : f_1(x') = f_1(X) \text{ and } (x', Y) \in A_\varepsilon^{(n)}\} \\ E_2 &= \{\exists y' \neq Y : f_2(y') = f_2(Y) \text{ and } (X, y') \in A_\varepsilon^{(n)}\} \\ E_3 &= \{\exists (x', y') : x' \neq X, y' \neq Y, f_1(x') = f_1(X), f_2(y') = f_2(Y) \text{ and } (x', Y) \in A_\varepsilon^{(n)}\} \end{aligned} \quad (1.6)$$

Thus

$$\begin{aligned} P_e^{(n)} &= P(E_0 \cup E_1 \cup E_2 \cup E_3) \\ &\leq P(E_0) + P(E_1) + P(E_2) + P(E_3) \end{aligned} \quad (1.7)$$

Extending the result for a single source to two sources, we can say that the cardinality of the set of jointly atypical sequences (xy) is very small compared to that of the jointly typical sequences. It follows that the probability measure of that set $\rightarrow 0$ for large n . Hence,

$$P(E_0) = \varepsilon$$

Now lets consider $P(E_1)$,

$$P[E_1/(X = x, Y = y)] = \bigcup_{(x', y) \in A_\varepsilon^{(n)}, (x' \neq x)} \{f_1(x') = f_1(x)\}$$

Thus,

$$\begin{aligned} P[E_1] &= \sum_{(x', y')} p(x, y) P[E_1/(X = x, Y = y)] \\ &\leq \sum_{(x, y)} p(x, y) \cdot \sum_{(x', y) \in A_\varepsilon^{(n)}, (x' \neq x)} P[f_1(x') = f_1(x)] \\ &\leq \sum_{(x, y)} p(x, y) 2^{-nR_X} |A_\varepsilon(X/y)| \\ &\leq 2^{-nR_X} 2^{n(H(X/Y) + \varepsilon)} \end{aligned} \quad (1.8)$$

which $\rightarrow 0$ if $R_X > H(X|Y)$ and n is large. The above result follows from the following:

$$P[f_1(x') = f_1(x) / (f_1(x) = i_0)] = 2^{-nR_X}$$

This implies that,

$$\begin{aligned}
P[f_1(x') = f_1(x)] &= \sum_{i_0} P[(f_1(x) = i_0)] \cdot P[f_1(x') = f_1(x) / (f_1(x) = i_0)] \\
&= 2^{-nR_X} \cdot \sum_{i_0} P[f_1(x) = i_0] \\
&= 2^{-nR_X}
\end{aligned} \tag{1.9}$$

$|A_\varepsilon(X/y)|$ is defined to be the set of X sequences that are jointly typical with a particular Y sequence. The proof for the fact that $|A_\varepsilon(X/y)| \leq 2^{n(H(X/Y)+2\varepsilon)}$ is as follows,

$$\begin{aligned}
1 &\geq \sum_{x \in A_\varepsilon^{(n)}(X/Y)} p(X/Y) \\
&\geq \sum_{x \in A_\varepsilon^{(n)}(X/Y)} 2^{-n(H(X/Y)+2\varepsilon)} \\
&= |A_\varepsilon^{(n)}(X/Y)| 2^{-n(H(X/Y)+2\varepsilon)}
\end{aligned}$$

Thus we have,

$$|A_\varepsilon^{(n)}(X/Y)| \leq 2^{n(H(X/Y)+2\varepsilon)}$$

When dealing with large values of n and high rates, the probabilities of the events E_2 and E_3 get arbitrarily small. It follows from the above discussion that the overall probability of error for the joint sequence at the decoder is,

$$P_e^{(n)} \leq 4\varepsilon$$

which is arbitrarily small. It can, therefore, be seen that the condition for achievability of the rate pair has been satisfied by $(R_X, R_Y) = (H(X/Y) + \varepsilon_x, H(Y) + \varepsilon_y)$. Hence, this is the proof for the theorem.

The rate pair that has been suggested above can change roles i.e, we can have $(R_X, R_Y) = (H(X) + \varepsilon_x, H(Y/X) + \varepsilon_y)$ and the theorem would still hold. This is equivalent to saying that the decoder now has complete information about the source and is trying to decode Y based on the joint typical sequence set. Thus, the rate region can be expressed as,

$$\begin{aligned}
R_X &\geq H(X|Y), R_Y \geq H(Y|X) \\
R_X + R_Y &\geq H(X, Y).
\end{aligned} \tag{1.10}$$

Despite the separate encoding of X and Y , SW proves that the total rate, $R_X + R_Y$, for encoding X and Y can achieve the joint entropy $H(X, Y)$ as if they were jointly encoded.

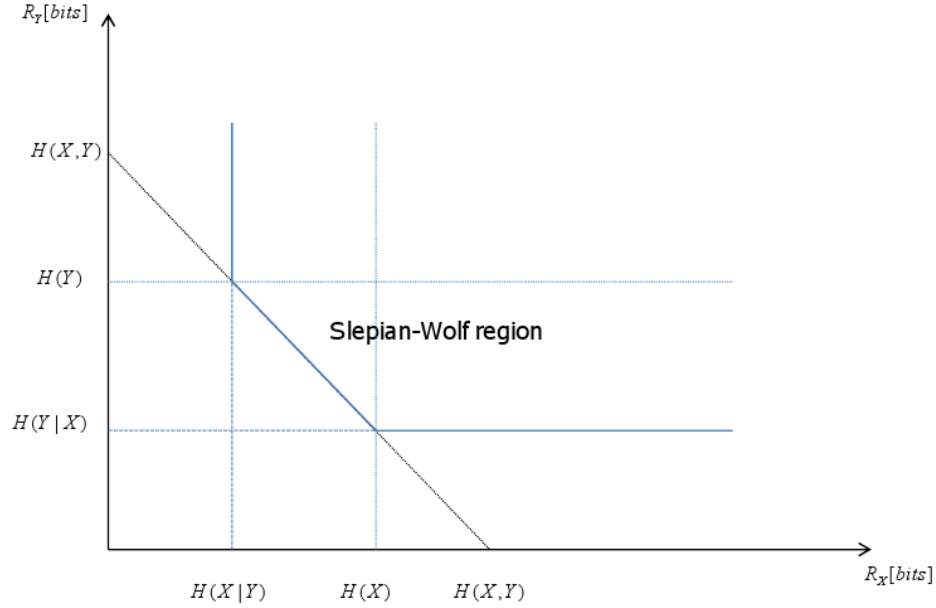


Figure 1.2: Admissible rate "Slepian-Wolf Region"

According to the Slepian-Wolf theorem and equation 1.10, the rate region, called "*Slepian-Wolf region*", for the reconstruction with an arbitrary small error probability of X and Y can be described by Fig. 1.2, where the vertical, horizontal and diagonal lines, corresponding to those three formulas of equation 1.10 respectively, represent the lower bounds for the achievable rate combinations of $R(X)$ and $R(Y)$. Slepian-Wolf coding generally refers to the lossless distributed source coding. Notice that lossless here is not mathematically lossless but allowing a controlled amount of errors which is approaching the lossless case. One interesting feature of Slepian-Wolf coding is that it is a close kin to channel coding which was already studied by Wyner. Considering two i.i.d. binary sequences X and Y and a virtual correlation channel, the source sequence X and side information sequence Y are modeled as the input and output of the virtual channel respectively. Y is therefore a noisy version of X where noise introduced by the channel refers to the correlation between X and Y . Then a systematic channel code can be adopted to encode X and only the resulting parity bits are transmitted. At the decoder, the received parity bits and the side information Y are used to perform error-correcting decoding. In this approach, significant compression is resulted due to the fact that only few parity bits

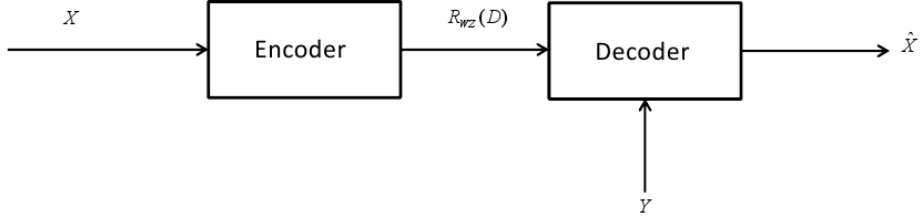


Figure 1.3: Lossy compression with side information

are needed to be sent.

1.2.1 Wyner-Ziv theorem

Later on, Wyner and Ziv studied the counterpart of SW theorem for lossy coding and established the bounds for lossy compression with side information at the decoder,[4], where the decoder produces \hat{X} with a certain distortion D with respect to X as illustrated in Figure 1.3.

When the SI is available at both, encoder and decoder sides, a rate $R_{X|Y}(D)$ is achieved for encoding X with a distortion D . Further, there is an increase of $(R_{X|Y}^{WZ}(D) - R_{X|Y}(D)) = 0$ in rate when the SI is not available at the encoder but only at the decoder side. In other words, the rate in the case where the SI is not available at the encoder is lower bounded by the one when the SI is available at the encoder. However, Wyner and Ziv show that both rates, $R_{X|Y}^{WZ}(D)$ and $R_{X|Y}(D)$, are equal when the sources are memoryless Gaussian and the Mean Square Error (MSE) is used as the distortion metric. With a predefined threshold D , they established the minimum rate necessary to encode X guaranteeing \hat{X} with an average distortion below D . The results indicated that for the same threshold D , the minimum encoding rate (for X) of the case when the statistical dependency between X and Y is only available at the decoder, described by $R_{WZ}(D)$, is bigger than that of the case when the dependency is available both at the encoder and the decoder, described by $R_{X|Y}(D)$. The Wyner and Ziv theorem also can be described by Eq.1.11:

$$R_{WZ}(D) \geq R_{X|Y}(D) \quad (1.11)$$

In the literature, $R_{WZ}(D)$ and $R_{X|Y}(D)$ are called Rate-Distortion (RD) functions. Notice that when $D = 0$, equation 1.11 falls to the Slepian-Wolf result which means that it is possible to reconstruct X with an arbitrarily small error probability even when the correlation between X and Y is only available at the decoder.

Finally, the Slepian-Wolf theorem and the Wyner-Ziv theorem state that is possible compressing two statistically dependent sequences in a distributed way (separate encoding, jointly decoding); the difference between them is that in the Slepian-Wolf theorem the dependency between two sequences is available both at the encoder and the decoder therefore the coding is lossless with allowing an arbitrary small error probability between the source sequence and the reconstructed sequence, while in the Wyner-Ziv theorem the dependency is only available at the decoder and sequences are lossy coded.

1.3 State of Art

Distributed Video Coding (DVC) states that it is theoretically possible to separately encode and joint decode two or more statistically dependent sources at the same rate obtained when the same sources are joint encoded and decoded, [4][2]. This strategy has been adopted by many authors for the design of high compression rate inter-frame video coding schemes. The common goal is to generate at the decoder a side information that optimally blends temporal and interview data. In other terms, while standard video coders exploit the statistical dependencies of the source signal in order to remove spatial and temporal redundancies, in DVC each video frame is encoded independently, knowing that some side information will be available at the decoder to remove transmission errors and improve the video quality. This approach considerably reduces the overall amount of transmission necessary from the cameras to the central decoder and simplifies the complexity of the video encoder by shifting all the complex interframe processing tasks to the decoder. This property can be very interesting for power/processing limited systems such as wireless camera sensors that have to compress and send video to a fixed base station in a power-efficient way. It is normally assumed that the receiver can run a more complex decoder but when the receiver is another complexity-constrained device, a more powerful

video transcoder somewhere on the network can be used.

However, although all these approaches are extremely promising, they are still not as efficient as standard video coders in terms of rate-distortion performance due to the fact that distributed source coding techniques rely on a a-priori knowledge of the correlation structure. These approaches are often not simple in practical applications as asymmetric: in fact some cameras need to transmit their full information to provide side information to the decoder while others only transmit partial information. Finally most of the multi-view DVC approaches do not take advantage of the multi-view geometry to improve the performance of their encoders.

The first attempts to design quantizer for reconstruction with side information were inspired by the information theoretic proofs. Zamir and Shamai , [5], proved that, under certain constraints, linear codes and nested lattices may reach Wyner-Ziv rate-distortion function when source data and side information are jointly Gaussian. This idea has been elaborated and applied by Pradhan et al., [6], who studied both the asymmetric case of source coding with side information at the decoder for Gaussian sources that are statistically dependant and the symmetric case where both sources are encoded at the same rate. Xiong et al., [7] implemented instead a nested lattice quantizer as WZ encoder, followed by a SW coder and proved that Low-Density-Parity-Code (LDPC) can be a powerful solution for DVC.

Yeo and Ramchandran [8] proposed a robust method that exploits inter-view correlation among cameras that have overlapping views in order to deliver error-resilient video in a distributed multiple wireless camera sensors scenario. The system has low encoding complexity, satisfies tight latency constraints, and requires no inter-sensor communication. Each video frame is divided into non-overlapping blocks and the syndrome of each quantized block is transmitted with a cyclic redundancy check (CRC) computed on the quantized block. The encoder at each of the video camera sensors does not need any knowledge about the relative positions of any other cameras. The decoder searches over candidate predictors and attempts to decode using the received syndrome and the candidate predictor as side-information. If the CRC of the decoded sequence checks out, decoding is assumed to be successful. In particular, the decoder first tries to decode a

block using decoder motion search in the temporal dimension; if that fails, then decoder performs the disparity search along the epipolar line in each overlapping camera view.

In [9] the authors propose a practical solution for Wyner-Ziv stereo coding that avoids any communication between the low complexity encoders. The method is based on a pixel-level mask fusion of temporal and interview side information. In particular, the first view is coded in conventional way using H.264/AVC, and DVC principles are applied to the coding of the second, dependent view. The system fuses, pixel-by-pixel, the temporal side information created using a motion-based frame interpolation scheme with the interview side information created using a disparity-based frame extrapolation algorithm. This technique shows the potential of a side information fusion approach performed at the decoding stage. The same approach is followed in [10], where the authors adaptively select either the temporal or the interview side information on a pixel by pixel basis. The system uses also a turbo decoder to detect when decoding is successful and no more parity bits need to be requested via the feedback channel. The proposed algorithm has the advantage to be symmetric with respect to the two cameras.

1.4 The Considered DVC Architecture

The considered practical DVC architecture used in this research follows Pereira approach, [9], and it is shown in Figure 1.4.

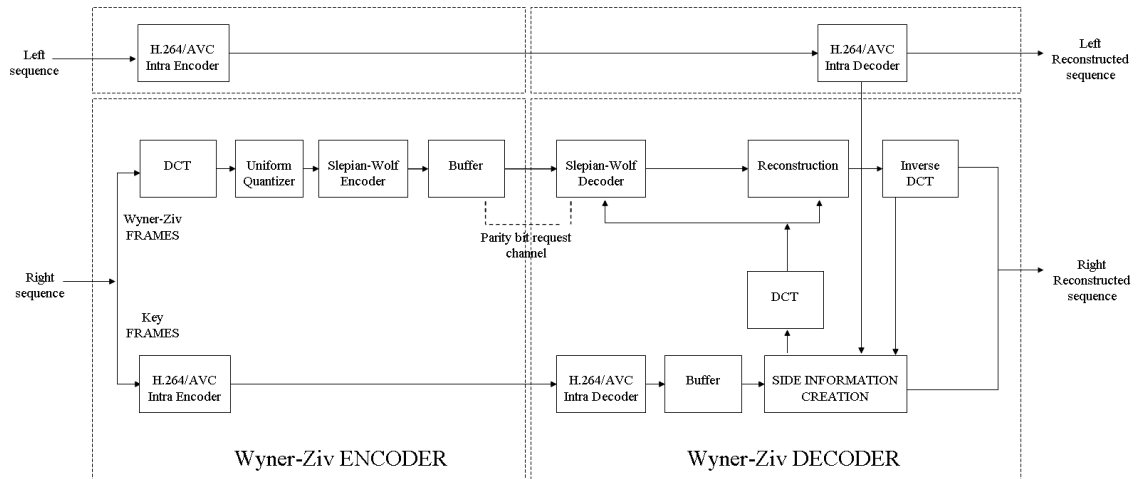


Figure 1.4: Stereo video coder architecture.

This codec is an evolution of the one initially proposed in [11] and uses a feedback channel based turbo coding approach.

The DVC encoding architecture works as follows: a video sequence is divided into Wyner-Ziv (WZ) frames and key frames. The key frames may be inserted periodically with a certain Group of Pictures (GOP) size or an adaptive GOP size selection process may be used depending on the amount of temporal correlation in the video sequence; most results available in the literature use a GOP of 2 which means that odd and even frames are key frames and WZ frames, respectively. While the key frames are conventionally encoded with video codec such as AVC/H.264 Intra; the WZ frames are DCT transformed and then quantized,[12]. Then, the quantized coefficients are split into bit planes, and one by one are turbo encoded. The reason of the this choice lies in near-channel capacity error correcting capability of the turbo code. The parity bits are stored in the buffer and transmitted in small amounts upon decoder request via the feedback channel.

At the decoder, the frame interpolation module is used to generate the side information frame, an estimate of the WZ frame X_i , based on previously decoded frames, X_{i-1} and X_{i+1} . The side information is treated as noisy corrupted version of coded WZ frame and used to decode the coded WZ frames at the decoder. For a Group Of Pictures (GOP) length of 2, X_{i-1} and X_{i+1} are the previous and the next temporally adjacent key frames, Intra coded. The side information (SI) is then fed by an iterative turbo decoder to obtain the decoded quantized symbol stream. The decoder requests for more parity bits from the encoder via the feedback channel whenever the adopted request stopping criteria has not been fulfilled; otherwise, the bitplane turbo decoding task is considered successful. The side information, together with the decoded quantized symbol stream, is also used in the reconstruction module. After all DCT coefficients bands are reconstructed, a block-based 4×4 inverse discrete cosine transform (IDCT) is performed and the result is the reconstructed WZ frame. To finally get the decoded video sequence, decoded key frames and WZ frames are conveniently merged. The statistic dependency between the original WZ frame X_i and the side information Y_i is modeled as Laplacian distribution. When Y_i , received parity bits and derived Laplacian distribution parameters, is obtained then it is possible to turbo decode and then reconstruct the quantized symbols.

When dealing with stereo sequence, there are two dependent views to be coded. In a Wyner-Ziv coding framework, the available statistical dependency has to be exploited not only in time as was done for the monoview case, but also in space, i.e. between the two dependent views.

1.4.1 Transformation

The aim of the transformation phase is to make the input video more suitable for compression by compacting the signals energy into the lower transform coefficients. The DVC scheme uses the 4×4 separable integer transform in AVC/H.264 with properties similar to the DCT. Given a $N \times M$ frame, the DCT transform is defined as follows, Equation 1.12:

$$X_{DCT}[u, v] = \frac{4}{NM} c[u]c[v] \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} x[n, m] \cos \left[\frac{(2n+1)u\pi}{2N} \right] \cos \left[\frac{(2m+1)v\pi}{2M} \right] \quad (1.12)$$

where $X_{DCT}[u, v]$, for $u = 0, 1, \dots, N-1$ and $m = 0, 1, \dots, M-1$ represents the DCT coefficient at (u, v) , i.e., line (row, vertical axis) u and column (horizontal axis) v , and:

$$c[u], c[v] = \begin{cases} \frac{1}{\sqrt{2}}, & \text{for } u, v = 0 \\ 1, & \text{otherwise} \end{cases}$$

The first cosine term is the vertical basis function generator represented by sampled cosine signal. n sets the sample number and u sets the frequency. For the same reason, the second cosine term is followed as the horizontal basis function generator. Since the DCT is separable, the two-dimensional DCT can be obtained by computing 1-D DCT in each dimension separately. For the implementation view point, most of international standards favor 4×4 block size, considering its complexity and performance. It converts the image block into a form where redundancy or correlation in the image data is reordered in terms of the basis images, so that the redundancy can be easily detected and removed. The detection is possible by the virtue of orthogonal property of the basis images; non-zero coefficients are obtained if an image pattern block coincides with the basis block. Natural image data, of course, may not coincide with the rectangular shaped basis images. Although the DCT shows sufficient performance of compression capability, one major disadvantage of the DCT is the block structure that dominates at very low bit rates, called blocking artifacts transforms. If the difference of quantization errors between two

adjacent blocks is so large, it would be easily detectable by human eye and the block disparity occurs.

1.4.2 Quantization

In general, a Wyner-Ziv coder can be thought as consists of a quantizer followed by a Slepian-Wolf encoder. Quantization of the sampled data is performed with a finite number of levels. It is assumed that the sampling is uniform and sampling rate is above the Nyquist rate so that there is no aliasing in the frequency domain. Some criteria, such as minimization of the quantizer distortion, have been used for quantization of image data. Quantizer design includes input (decision) levels, output (representation) levels and the number of levels. A uniform quantizer is completely defined by the number of levels, step size and if it is a midriser or a midreader. Instead of the type of quantizers, a quantized output (reconstruction) value is determined in a certain interval (quantization step) where any of the input values happens. Since the reconstruction value represents the whole range of input values, quantization inherently is a lossy process and the lost information may not be recovered. Since, usually, the distribution of image data is concentrated on mean value region and image processing, including predictive coding and transform coding, produces more abundant distribution on smaller levels near zero, which means less energy or variance, the region can be quantized with fine step size, while others can be quantized with coarse step size. A nearly uniform quantizer is designed using these properties, enlarging the step size only in the mean value region, called a deadzone. Except for the deadzone (input range for which the output is zero), the stepsize is constant.

1.4.3 Slepian-Wolf Encoder

The Slepian-Wolf codec can be implemented by a systematic channel code as turbo code. This coding technique consists essentially of a parallel concatenation of two binary convolutional codes, decoded by an iterative decoding algorithm. These codes obtain an excellent bit error rate (BER) performance by making use of three main components, [13]. A turbo encoder is constructed using two RSC (Recursive Systematic Convolutional) encoders arranged in parallel and combined with a random interleaver, together with a

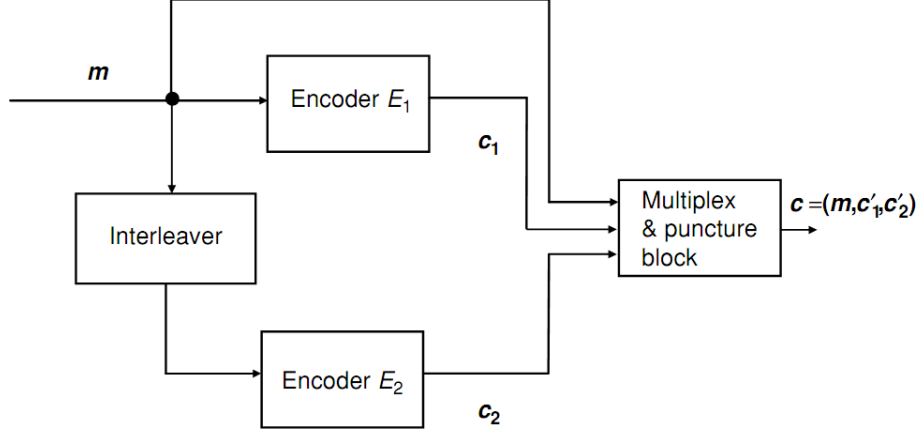
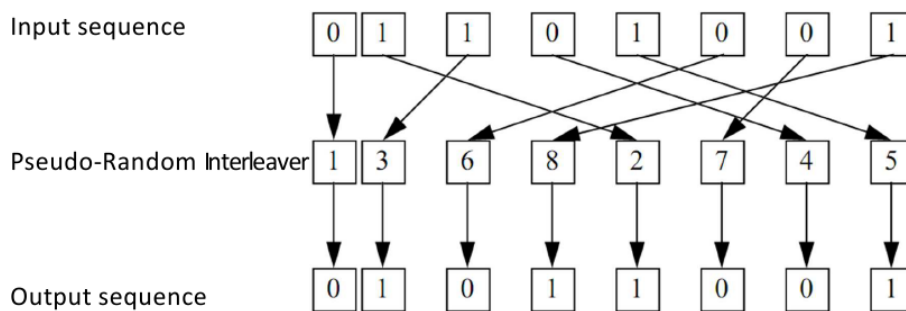


Figure 1.5: Turbo encoder structure

multiplexing and a puncturing block. Typically, the two encoders E_1 and E_2 are RSC encoders of rate $R_c = 1/2$, such that $c'_1 = c_1, c'_2 = c_2$ and the lengths of the incoming sequences m, c_1 and c_2 , and c'_1 and c'_2 are all the same. Then the overall turbo code rate is $R_c = 1/3$. Puncturing is a technique very commonly used to improve the overall rate of the code. It consists in periodically eliminating one or more of the outputs generated by the constituent RSC encoders. Thus, for instance, the parity bits generated by these two encoders can be alternately eliminated so that the redundant bit of the first encoder is first transmitted, eliminating that of the second decoder, and in the following time instant the redundant bit of the second encoder is transmitted, eliminating that of the first. Puncturing is not usually applied to the message (systematic) bits, because this causes a BER performance loss. In this way, the lengths of c'_1 and c'_2 are half the lengths of c_1 and c_2 , respectively, and the resulting overall rate becomes $R_c = 1/2$. Puncturing is not usually applied to the message (systematic) bits, because this causes a BER performance loss. There are two important components of a turbo encoder whose parameters have a major influence on the BER performance of a turbo code: the first is the interleaver, especially its length and structure, and the second is the use of RSC as constituent encoders. The interleaver reads the bits in a pseudo-random order. The choice of the interleaver is a crucial part in the turbo code design in fact the task of the interleaver is to "scramble" bits in a (pseudo-)random. Two are the purposes. Firstly, if the input to the second encoder is interleaved, its output is usually quite different from the output of the first encoder. This

Figure 1.6: A pseudo-random interleaver with $L = 8$

means that even if one of the output code words has low weight, the other usually does not, and there is a smaller chance of producing an output with very low weight. Higher weight, as we said before, is beneficial for the performance of the decoder. Secondly, since the code is a parallel concatenation of two codes, the divide-and-conquer strategy can be employed for decoding. If the input to the second decoder is scrambled, also its output will be different, or uncorrelated from the output of the first encoder. This means that the corresponding two decoders will gain more from information exchange. The excellent BER performance of these codes is enhanced when the length of the interleaver is significantly large, but also important is its pseudo-random nature. The interleaving block, and its corresponding de-interleaver in the decoder, does not much increase the complexity of a turbo scheme, but it introduces a significant delay in the system, which in some cases can be a strong drawback, depending on the application. The RSC-generated convolutional codes are comparatively simple, but offer excellent performance when iteratively decoded using Soft-InputSoft-Output (SISO) algorithms. The interleaver used in this work, is the pseudo-random interleaver: it uses a fixed random permutation and maps the input sequence according to the permutation order. The length of the input sequence is assumed to be L . Figure 1.6 shows a random interleaver with $L = 8$.

1.4.4 Parity bit Request Channel

For each Wyner-Ziv frame, the decoder generates the side information \hat{S} by interpolation or extrapolation of previously decoded key-frames and, if possible, previously decoded Wyner-Ziv frames. To exploit the side information, the decoder assumes a statistical model

of "parity bit request channel". Specifically, a Laplacian distribution of the difference between the individual pixel values S and \hat{S} is assumed. The latter has the following Probability Density Function, Eq. 1.13:

$$p_X(x) = \frac{\alpha_x}{2} e^{-\alpha_x |x|} \quad (1.13)$$

where α_x is related to the subband coefficient variance σ_x^2 of the distribution, it controls how much the side information resembles the original frame. It is estimated by computing the variance of the residual between WZ frame and side information frames offline, [14]. The turbo decoder combines the side information \hat{S} and the received parity bits to recover the Wyner-Ziv frame. If the decoder can't reliably decode the original data, it requests additional parity bits from the encoder buffer through feedback. The "request-and-decode" process is repeated until an acceptable probability of symbol error is reached. Moreover, the bit rate for a Wyner-Ziv frame is determined by the statistical dependence between the frame and the side information. While the encoding algorithm itself does not change, the required bit rate does as the parity bit request channel statistics change. The decision on how many bits to send for each frame is tricky, since the side information is exploited only at the decoder but not at the encoder. One approach to solve the rate control problem relies entirely on decoder and feedback information. The decoder attempts decoding by using the bits received so far. If turbo decoding fails, the decoder requests additional bits from the encoder. Feedback also allows the decoder to have a great flexibility in generating the side information.

1.4.5 Side Information Creation

The side information plays a key issue in the WZ coding architecture. Side information can be seen as a corrupted version of the WZ frame passing through the "virtual correlation channel". In WZ coding, the correlation between the side information and the WZ frame is exploited at the decoder. Since the encoder has no knowledge of the side information during encoding process, the accuracy of the side information is consequently extremely important for the compression performance of WZ coding. In fact it can improve the efficiency of compression, requiring few bits to be sent during decoding if the side information

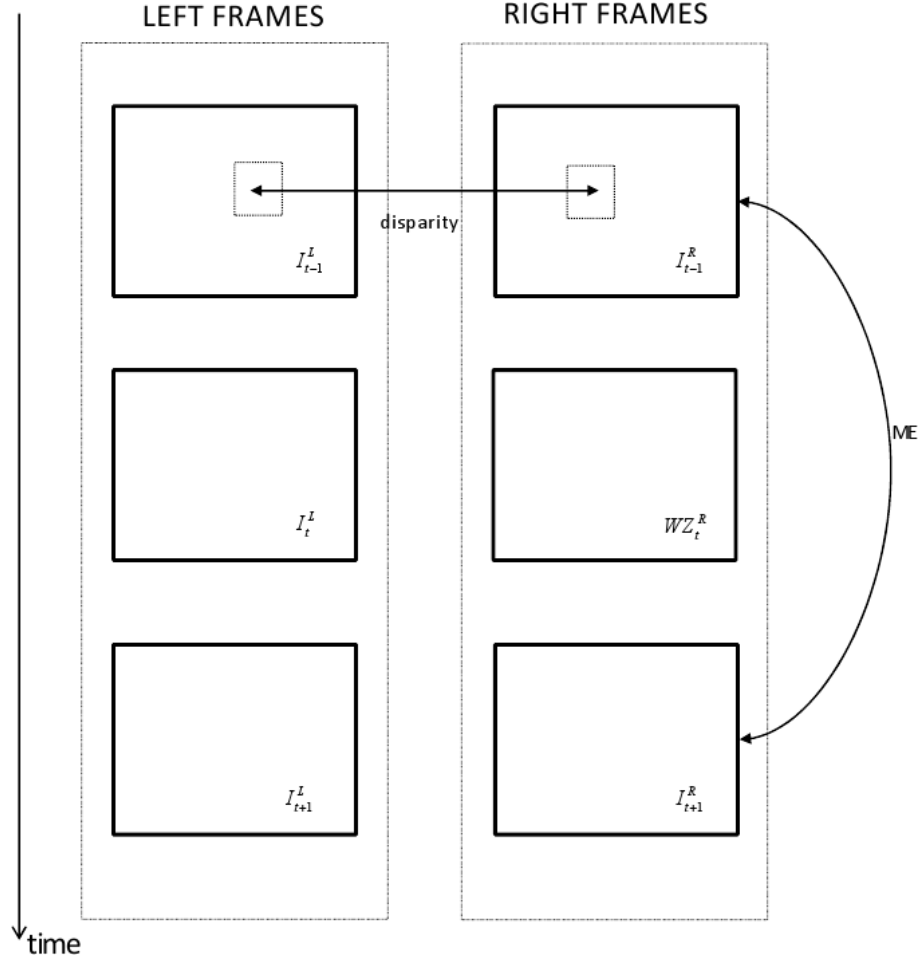


Figure 1.7: Side information creation as merging of disparity and temporal motion estimation for stereoscopic video sequence

is accurate enough, namely if it is very similar as the WZ frame. Otherwise, the encoder has to send more parity bits to correct the "errors" between the side information and the WZ frame and the compression effect is no more so efficient. It is computed fusing two information: the first coming from temporal motion estimation (ME) and the second coming from disparity between left and right frame. It is also very important to design an ad-hoc fusion scheme between these two data. When we are not dealing with stereoscopic video sequence, the side information is obtained exploiting only temporal information between the previous key-frames. In Fig. 1.7, the two motion estimation are shown.

The temporal side information with time index t is generated by performing motion compensated interpolation between the Intra H.264 decoded key frames at time $t - 1$ and

$t + 1$. This interpolation technique involves symmetrical bidirectional block matching, smoothness constraints for the estimated motion and overlapped block motion compensation. Since the next key frame is needed for interpolation, the frames have to be decoded out-of-order, similar to the decoding of B frames in predictive video coding. The essence of side information generation is frame interpolation between two frames. The frame interpolation is based on the assumption that smooth motion lies between key frames and objects motion obeys the linear translation model. This assumption is not always true, especially when dealing with high motion sequences and the interpolation quality can be degraded seriously. Once temporal motion estimation between the previous and next Intra frame and disparity between left and right frame are performed, two side information (temporal and spatial) are created and it is necessary to find an efficient way to fuse the temporal and inter-view correlations so that the decoder may at least take benefit of the most powerful of them for each decoded frame. Several techniques have been proposed in literature, [10], [14]. The fusion-based approach used in this work is based on the following ideas, [9]:

1. For each WZ frame under decoding, the (fused) side information is created based on the temporal and interview side information; this side information which is expected to be better than each individual side information should provide better rate-distortion (RD) performance than temporal or inter-view alone.
2. The process to fuse the two individual side information is based on a binary fusion mask created after decoding the most recent frame; pixel by pixel, this mask is set to 1 if the temporal side information is the most similar to the decoded frame and set to 0 if inter-view side information is the most similar.

In Fig.1.8, the fusion side information scheme is presented. The binary decision mask used for each WZ frame being decoded indicates which is the best side information to use for each pixel: the temporal or inter-view side information.

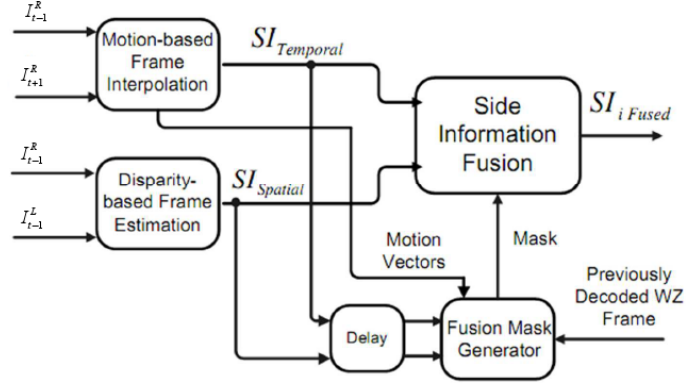


Figure 1.8: Stereo side information generation architecture using a mask-based fusion approach.

1.4.6 Slepian-Wolf Decoder

In the considered DVC system, Slepian-Wolf decoder is essentially constituted by a turbo decoder. Turbo decoding involves iterative exchange between the constituent decoders for progressively better estimates of the message bits, in a decoding procedure that is helped by the statistical independence of the two code sequences generated by each input bit. In the decoding procedure, each decoder considers the information provided by the samples of the channel, the systematic (message) and parity bits, together with the *a priori* information that was provided by the other decoder, that was previously calculated as its extrinsic information. However, instead of making a hard decision on the estimated message bits, as done for instance in the traditional decoding of convolutional codes using the Viterbi algorithm, the decoder produces a soft-decision estimate of each message bit. This soft-decision information is an estimate of the corresponding bit being a 1 or a 0; that is, it is a measure of the probability that the decoded bit is a 1 or a 0. This information is more conveniently evaluated in logarithmic form, by using a log likelihood ratio (LLR). This measure is very suitable because it is a signed number, and its sign directly indicates whether the bit being estimated is a 1 (positive sign) or a 0 (negative sign), whereas its magnitude gives a quantitative measure of the probability that the decoded bit is a 1 or a 0, [13]. The turbo decoder is composed of two Soft Input Soft Output (SISO) decoders: P1 and P2 are the punctured versions of the parity bits produced by the turbo encoder. The

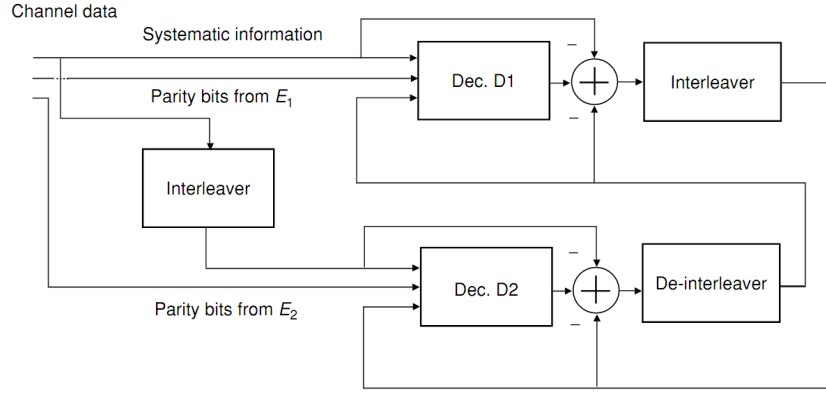


Figure 1.9: Turbo Decoder scheme.

systematic bits, S , are extracted directly from the SI, which can be seen as a corrupted version of the original data after passing through a virtual channel. The parity bits request channel model is used to try to predict the errors present in the SI.

The problem of the decoding of a turbo code is essentially to determine Maximum A Posteriori (MAP) estimates or soft decisions of states and transitions of a trellis encoder. The MAP algorithm is related to many other algorithms, such as Hidden Markov Model, HMM which is used in voice recognition, genomics and music processing.

In addition to MAP algorithm, another algorithm called SOVA, based on Viterbi decoding is also used. SOVA uses Viterbi decoding method but with soft outputs instead of hard. SOVA maximizes the probability of the sequence, whereas MAP maximizes the bit probabilities at each time, even if that makes the sequence not-legal. MAP produces near optimal decoding. In turbo codes, the MAP algorithm is used iteratively to improve performance.

The info bits are called u_k . The coded bits are referred to by the vector c . Then the coded bits are transformed to an analog symbol X and transmitted. On the receive side, a noisy version of X is received. A metric of confidence, that represents how far the received symbol is from the decision regions, is added to each of the three bits. Often Gray coding is used, which means that not all bits in the symbol have same level of confidence for decoding purposes. There are special algorithms for mapping the symbols (one received voltage value, to M soft-decisions, with M being the M in M -PSK.) Let's assume that

after the mapping and creating of soft-metrics, the vector Y is received. One pair of these decoded soft-bits are sent to the first decoder and another set, using a de-interleaved version of the systematic bit and the second parity bit are sent to the second decoder. Each decoder works only on these bits of information and pass their confidence scores to each other until both agree within a certain threshold. Then the process is iterated with next symbol in a sequence or block consisting of N symbols (bits). The state of the source in time instant i is denoted as S_i , and its output is X_i . A sequence of states from time instant i to time instant j will be denoted as $S_i^j = S_i, S_{i+1}, \dots, S_j$, and will be described by the corresponding output sequence $X_i^j = X_i, X_{i+1}, \dots, X_j$. X_i is the i -th output symbol taken from a discrete alphabet. The state transitions are determined by the transition probabilities:

$$p_i(u/u') = P(S_i = u/S_{i-1} = u') \quad (1.14)$$

and the corresponding outputs by the probabilities:

$$q_i(X/u', u) = P(X_i = x/S_{i-1} = u', S_i = u) \quad (1.15)$$

where x is taken from the discrete output alphabet. The discrete hidden Markov source generates a sequence X_1^n that starts at state $S_0 = 0$ and ends at the same state $S_0 = 0$. The output of the discrete hidden Markov source X_1^n is the input of a noisy discrete memoryless channel that generates the distorted sequence $Y_1^n = Y_1, Y_2, \dots, Y_n$. Transition probabilities of the discrete memoryless channel are defined as $R(Y_j/X_j)$, such that for every time instant $1 \leq i \leq n$,

$$P(Y_1^i/X_1^i) = \prod_{j=1}^i R(Y_j/X_j) \quad (1.16)$$

The term $R(Y_j/X_j)$ determines the probability that at time instant j , the symbol Y_j is the output of the channel if the symbol X_j was input to that channel. This will happen with a transition probability $P(y_j/x_j)$ that the input symbol x_j converts into the output symbol y_j . A decoder for this Markov process has to estimate the MAP probability of states and outputs of the discrete hidden Markov source by observing the output sequence

$Y_1^n = Y_1, Y_2, \dots, Y_n$. This means that it should calculate the probabilities:

$$\begin{aligned} P(S_i = u/Y_1^n) &= \frac{P(S_i=u, Y_1^n)}{P(Y_1^n)} \\ P(\{S_{i-1} = u', S_i = u\}/Y_n) &= \frac{P(S_{i-1}=u', S_i=u, Y_1^n)}{P(Y_1^n)} \end{aligned} \quad (1.17)$$

The notation here is that the state S_i defines a given state i in a trellis, whereas its particular value is obtained from an alphabet U of states of the trellis, with $u = 0, 1, 2, \dots, U - 1$. Therefore, in a trellis, the sequence $Y_1^n = Y_1, Y_2, \dots, Y_n$ is represented by a unique path. The following MAP probability is associated with each node or state of a trellis:

$$P(S_i = u/Y_1^n)$$

and the following MAP probability is associated with each branch or transition of the trellis:

$$P(S_{i-1} = u', S_i = u/Y_1^n)$$

The decoder will calculate these probabilities plus joint probabilities.

1.4.7 Reconstruction

This phase is the opposite of the quantization step at the encoder. To reconstruct the current frame, the side information with decoded WZ DCT bins are used. It consists in either accepting a side information value if it fits into the quantization interval corresponding to the decoded bins or truncating the side information value into this quantization interval. Let Y be the side information value, d the decoded quantization index, Δ the quantization step and \hat{X} the reconstructed value. The reconstructed value \hat{X} is computed as:

$$\hat{X} = \begin{cases} Y, & \text{if } d\Delta \leq Y \leq (d+1)\Delta \\ d\Delta, & \text{if } Y < d\Delta \\ (d+1)\Delta, & \text{if } Y > (d+1)\Delta \end{cases} \quad (1.18)$$

1.5 Application Scenarios for DVC

Several scenarios where distributed video coding can be applied are identified in this section, highlighting benefits and drawbacks. The most relevant DVC applications are studied evaluating DVC strengths such as error resilience, encoder-decoder complexity tradeoff,

low power encoder consumption according to specific requirements of each scenario. The most promising scenarios for DVC applications are, [15]:

- Wireless Video Cameras;
- Wireless Low-Power Surveillance;
- Mobile Document Scanner;
- Video Conferencing with Mobile Devices;
- Distributed Video Streaming;
- Multiview Video Entertainment.

1.5.1 Wireless Video Cameras

With the new emerging technologies for wireless communication, the availability of sending video data in a wireless fashion has now become a reality. In this contest, DVC approach can be efficiently exploited for wireless communications with remote devices. A list of some important applications where wireless cameras are used, is provided, but it is clear that many other situations can be considered as well. The first application deals with the possibility of using wireless portable cameras as home surveillance. A portable device that can be placed everywhere can be very useful in order to monitor if something is going to happen in a room, or when someone comes in a shop...

Very small wireless cameras can also be employed for police investigation purpose or for remote sensing of phenomena that are very hard to be physically reached (e.g. biomedical applications). Another application example can be the monitoring of traffic control. The advantage of having wireless cameras with respect to wired camera is the possibility to reconfigure the network and the positions of nodes inside the network.

In this contest, the following requirements appear to be relevant: low cost, error resilience, low-power consumption and small size. If on one side, DVC approach can be suitable for these kind of scenarios, on other side some current drawbacks have to be taken into account. In fact, the required decoding complexity seems to be very high for



Figure 1.10: wireless camera and monitor



Figure 1.11: Traffic management center at Tokyo

real-time applications moreover until now DVC did not reach the same level of compression efficiency as state of art predictive coding (e.g. H.264/AVC).

1.5.2 Wireless Low-Power Surveillance

In order to provide surveillance and security, different low-power consumption components are interconnected and the communication among them is guaranteed by wireless communication protocols. The components that provide information to the system are cameras (and/or sensors) and images captured or displayed by one or multiple devices.

As the range of applications that are placed into the wireless low-power surveillance network scenario is so wide and varied, the requirements may depend on the applications focused. However there are general requirements suitable to all the possible scenarios like low power consumption, small weight and small size, error resilience, compression efficiency and delay constraint. DVC technique allows to answer to all these requirements but at the same time it is necessary to consider a network transcoder because in an end-to-end low power surveillance network scenario, a transcoder inside the network must be used in order to keep both the encoder and the decoder as simple as possible.

1.5.3 Mobile Document Scanner

Nowadays, mobile phones can be used as portable fax or scanner simply by sweeping the phone across the page. Document scanning on the go with a mobile phone would give wireless carriers the opportunity to provide a host of new services, ranging from the most basic ones like document transmission to email addresses, to printer or the user's pc, to more advanced services like Optical Character Recognition (OCR) and instantaneous translations for travelers, sending back the translated text via instant messaging.

Scanning an A4 sized page by moving a mobile phone video camera over the document is likely to take about 3 or 5 seconds. Assuming a video frame rate ranging between 5 to 10 frames per second, this will produce between 15 to 50 images which a central server must merge together to extract the text and record any images. The application runs on the central server must then forwards the processed document to the targeted end device e.g., user's pc, email, printer and mobile phone.



Figure 1.12: Document scanning with mobile phone

The main requirements regarding this application are: complexity, video post-processing, central server processing and image processing quality. The most relevant advantages for DVC approach are a lower encoding complexity and an improved error resilience while the drawbacks are lower compression efficiency and an higher decoding complexity since the decoding is performed on the central server, one can afford to have an increased decoder complexity up to a point related to the scalability of the service. However, approaches with a more flexible load balancing between encoder and decoder might be very beneficial for such applications.

1.5.4 Video Conferencing with Mobile Devices

In a video-conference system, two or more users positioned in different locations can interact via two-way video and audio transmissions simultaneously through a set of interactive telecommunication technologies. It has also been called "visual collaboration" and is a type of groupware. With a video-conferencing it is possible to bring people at different sites together for a meeting, using audio and video systems. This can be as simple as a conversation between two people in private offices (point-to-point) or can involve several sites (multi-point) with more than one person in large rooms at different sites. Besides the audio and visual transmission of meeting activities, video-conferencing can be used to share documents, computer-displayed information, and whiteboards, [16].

It requires real-time and low-complexity. DVC answers to these requirements but at the same time it has low compression efficiency and it needs for a transcoder.

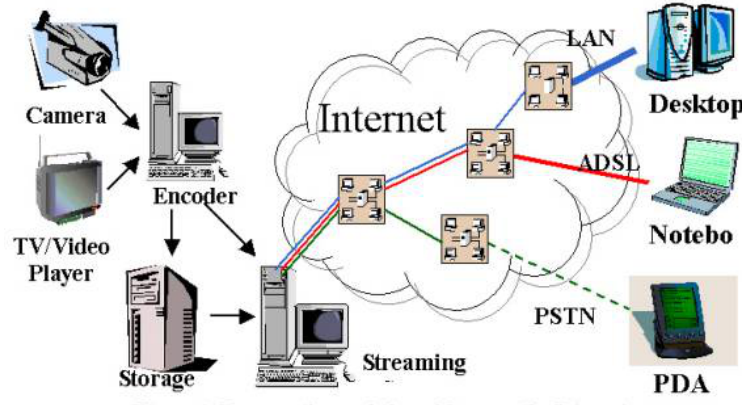


Figure 1. Transporting real-time video over the Internet.

Figure 1.13: Video Streaming solution over Internet

1.5.5 Distributed Video Streaming

The huge developments of Internet have given the chance to realize video streaming systems that allow a user to view a video sequence at its own place while downloading it from a remote server or a disk. In this setting, the user does not want to download first the video to see it at a later time but he wants instead to see the sequence while "streaming". With the same idea that led to the development of peer to peer networks for "distributed" download of files, it possible to consider the possibility of performing "distributed streaming" in order to give to the receiver the maximum data flow. In this way, the video stream is sent to the receiver by different senders in a distributed fashion, in order to reduce the bitrate at the sender sides and increases it at the receiver. The major requirements and functionalities to be considered are: compression efficiency, bitrate allocation, real-time performance, flexibility and error resilience.

1.5.6 Multiview Video Entertainment

Multiview images of a scene can be used for several applications ranging from free viewpoint television (FTV) to surveillance; entertainments applications are currently playing a more and more important role in multiview video systems. In FTV, the user can freely control the viewpoint position of any dynamic real-world scene. This system can cover a limited area, to extend the coverage area it is possible to introduce distributed sensors

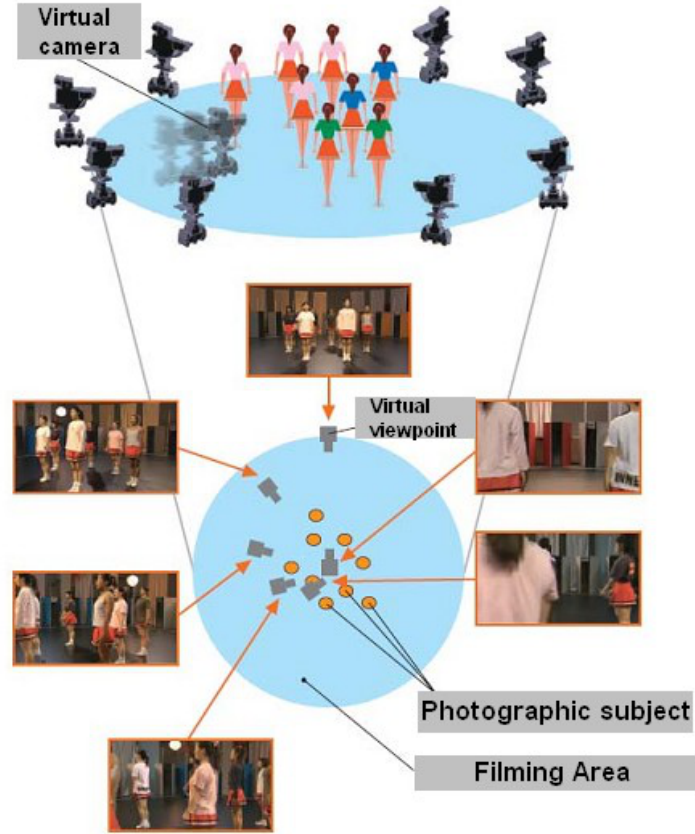


Figure 1.14: Free viewpoint Television scheme

network.

Many tasks can benefit from the availability of multi-view images of the same scene as restoration, interpolation and object recognition. Data reduction becomes a key-issue in multiview images and video processing. Furthermore, due to strong correlation between multiple views, multiview data reduction has its own characteristics that differ significantly from traditional image/video compression. The major requirements are: low cost and low complexity, high number of cameras, camera parameters and a priori knowledge of geometric information and robustness to occlusions. DVC satisfies almost all requirements but at the same time it is clear that visual occlusions present a challenging problem for any distributed video coding technique.

Chapter 2

Stereo Video Artifacts in a Distributed Coding Approach

2.1 Introduction

When dealing with stereoscopic 3D video, the perceived artifacts produce not only visually displeased results, but also general discomfort on the human visual system. Due to these reasons, the scientific community is focusing on the definition of a perceptual quality metric that quantifies the typical distortion that could occur. At this aim it is important to identify classes of artifacts which could arise in several scenarios involving stereoscopic content. It is important to underline that visual artifacts could arise at any processing and delivery phase of a stereo video sequence [17] [18]:

- Acquisition and content creation: there are three common approach to capture 3D video. Most 3D content is obtained by dual camera or multi-camera configurations where each view (left and right for stereoscopic sequence) is separately recorded. Settings parameters such as camera base distance (distance between two cameras), convergence distance and camera lens focal length are used to scale horizontally disparity and the degree of perceived depth, [19]. A second approach transform 2D video into 3D content using a conversion algorithm that derives depth map from 2D still frames or video sequence. Third, video output can be augmented by depth information collected by another sensor. All these approaches can cause unnatural effects

due to an incorrect configuration, calibration or positioning of the camera system. Typical artifacts introduced in this phase are noise, aliasing, blur, barrel distortion, pincushion, keystone distortion, vignetting, and aberration artifacts caused by the camera, as well inter-channel distortion such as vertical disparity, depth plane curvature, cardboard effect and puppet theater effect. In addition, there is a group of temporal artifacts such as motion blur and temporal mismatch between channels, keystone distortion, temporal mismatch and cardboard effect.

- Representation of the acquired data: there are many existing conversion formats but two main groups have mostly evolved: *multi-view video* where more than two video streams of the same scene are represented from different point of views, *video plus depth* already standardized as MPEG-C, Part 3, [20]. In particular, if representation format is different from the one the scene was captured then artifacts as dense depth video, ghosting by occlusion and temporal and spatial aliasing can be introduced.
- Encoding phase: stereoscopic color and depth video are normally encoded with multi-view coding schemes where video sequences are encoded as separate streams and temporal and inter-channel correlations are used to compress data or algorithms for 2D video adapted for stereo where 2D video and depth map are encoded separately. This step can alter image details that are important for depth perception. Typical degrading effects of this step are blocking, mosaic patterns, staircase effect, ringing, color bleeding and mosquito noise, as well as a depth ringing artifacts specific for dense depth video. Also, various asymmetric stereo-video coding schemes are sources of cross-distortion artifacts, where one channel is spatially or temporally downsampled.
- Transmission: on one side, packet data loss and channel noise introduced in a video content delivery can be sources of a degraded perceived quality of the content; on the other side, the algorithms that attempt to correct these errors, can cause additional problems on their own. Impairments are due to packet loss, jitter and color bleeding. Transmission causes propagating and non-propagating packet-loss artifacts, noise and jitter; however the last two are not characteristic for the DVB-H channel.

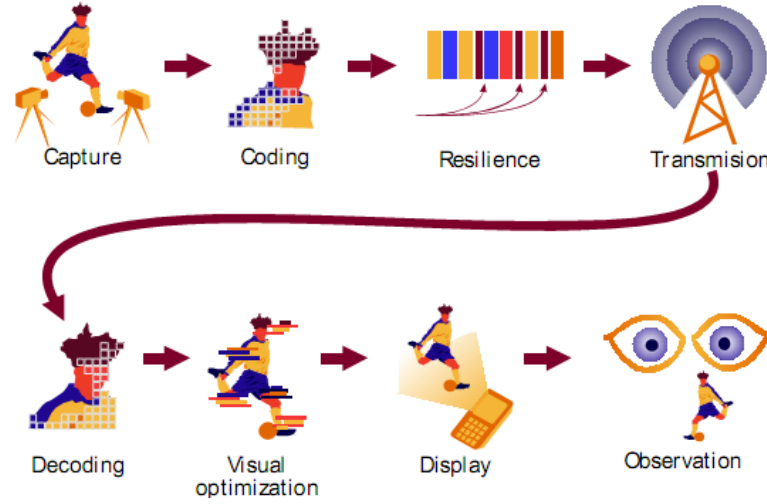


Figure 2.1: Data flow of 3D TV

- Visualization: stereo video quality is strongly dependent on the adopted approach to 3D visualization, i.e. on the artifacts that characterize the 3D displays. Flickering, cross-talk, puppet theater effect and shear distortion, aliasing, view interspersing (also known as ghosting artifacts), and accommodation-convergence rivalry can occur at this step. Additionally, the autostereoscopic displays, which are suitable for mobile 3DTV suffer from image flipping (also known as pseudoscopy), picket fence effect. The last group of artifacts depends greatly on the observation angle.

Considering that, from a quality of experience point of view, the attention has been focused on conventional video coders, the scope of this chapter is to analyze stereo video artifacts introduced by stereo distributed video coders available in the literature by means of subjective experiments. A comparison with more traditional coders is also provided. In particular, a sensitivity analysis with respect to the parameters that control bit-rate will be carried out, [21].

The rest of this chapter is organized as follows: in Section 2.2 an analysis of the artifacts related to stereo video coding is presented, while in Section 2.3 the adopted metrics are described. Finally in Section 2.4 some experimental results are reported and in Section 2.5 conclusions and future work are drawn.

2.2 Artifacts introduced in stereo video coding

In conventional 2-D video coding, the introduced monoscopic artifacts comprise all the typical artifacts of 2D images as blurring, noise, blocking and other structural changes. In the viewing of a stereo video, the final user could recognize "2D artifacts" but still having a perfect perception of the depth; obviously, larger distortion could damage the binocular view.

The stereoscopic artifacts change the relation between the two views and thus forbid the brain to have the proper binocular depth view. Such artifacts can change the disparity information of a scene or cause any other structural changes. Other annoying effects could convey unnatural information to the brain inducing so eye-strain and visual discomfort [19].

When dealing with stereo video content artifacts, we have to consider 4 groups of impairments based on how they are perceived by human brain: *structure*, *color*, *motion* and *binocular* [18], [22].

By structure, we mean those distortions that can impact on structural changes (i.e. contours and texture); by motion and color, those that can affect motion and color vision. Finally, binocular impairments can degrade the binocular depth perception when it is perceived as a stereo-pair (cannot be noted with a single eye). Based on this classification, we focalize our attention on artifacts introduced in the phase of coding.

In traditional and stereoscopic video coding, quantifying the artifacts in terms of the visual impact is a difficult task. In fact, the perceived distortion is not only related to the absolute quantization error but it is depending on local, global spatial inter-view and temporal characteristics of the video sequences [23]. Consequently, it is not possible to provide a specific bit-rate at which anyone artifacts is showed. Hence, the discussion will take into account the descriptions of some possible artifacts as *blocking effect*, *blurring*, *ringing*, *staircase effects* and *mosaic patterns* for the category **structure**; *color bleeding* for the category **color**; *motion compensation artifacts* and *mosquito effect* for the category **motion** and *cross-distortion*, *cardboard effect* for the category **binocular**.

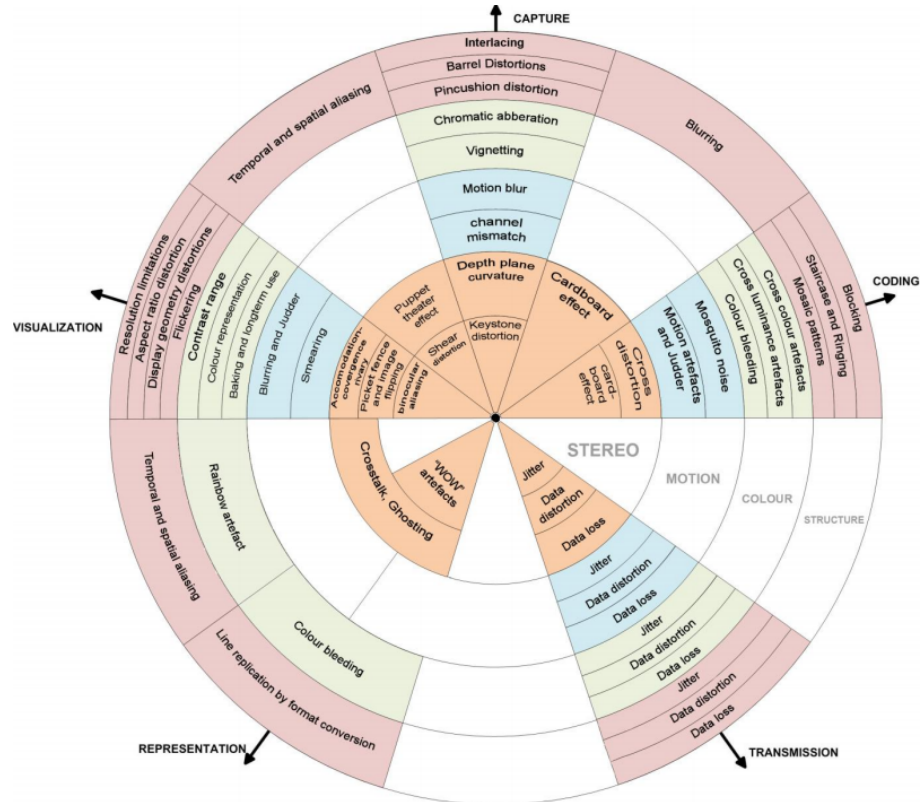


Figure 2.2: Artifacts affecting various stage of 3D video delivery

2.2.1 Artifacts in image structure

2.2.1.1 Blocking effect

The *blocking* effect is a discontinuity between consecutive blocks in the image. It can be seen as a discontinuity at the borders of each block in a reconstructed frame coming from individual treatment of each block in the coding stage according to its content,[22]. The severity of blocking effect is subject to the coarseness of the quantization of DCT coefficients of either one or both adjacent. One of the aims for the coding of pixels as block units is to exploit the high local inter-pixel correlation in a picture. Unfortunately, coding a block as an independent unit does not take into account the possibility that the correlation of the pixels may extend beyond the borders of a block into adjacent blocks, thereby leading to the border discontinuities. Since the blocking effect is more visible in the smoothly textured sections of a picture, the lower order DCT coefficients and in particular DC coefficients, play the most significant role in determining the visibility of the blocking effect. However, the blocking effect may occur in spatially active areas as a result of very coarse quantization.

Higher quantization suppress more DCT coefficients and this lost information causes the blocking artifacts. Due to the coarse quantization, a loss of spatial details is a consequence, visible as blurring after the reconstruction of an image. Blocking artifacts in color channels can also cause color bleeding.

2.2.1.2 Blurring effect

The *blurring* effect is a lack of spatial details in moderate to high spatial activity regions of pictures, such as in roughly textured areas or around scene object edges. For intra-frame coded macro blocks, blurring is directly related to the suppression of the higher order AC DCT coefficients through coarse quantization, representing the content of a block only through lower order coefficients.

2.2.1.3 Ringing effect

The *ringing* effect is most evident along high contrast edges in areas of generally smooth texture in the reconstruction, and appears as a shimmering or rippling outwards from the edge up to the encompassing blocks boundary. The higher is the contrast of the edge, the greater is the level of the peaks and troughs of the rippling.

2.2.1.4 Staircase effect

The *staircase* effect is linked to both the blocking and mosaic pattern effects because it can be seen as a discontinuity between adjacent blocks. When a diagonal edge is represented within a string of consecutive blocks, the consequence of coarse quantization is the reconstruction of the diagonal edge as a number of horizontal or vertical steps. It produces staircase edges.

In addition, the alignment of patterns or objects within a scene cannot always be well approximated by the separable DCT. The staircase effect and the ringing effect often occur together. High-contrast areas are the source of ripples and shimmering near the borders due to coarse quantization of high frequency components during the quantization.

2.2.1.5 Mosaic pattern effect

The *mosaic pattern* effect is the apparent mismatch between all, or part, of the contents of adjacent blocks; the overall effect remembers square tiles visually ill-fitted in a mosaic. It can be a block with a certain contour or texture dissimilar to the neighboring blocks, or a block used in the representation of an object which does not blend satisfactorily with the other constituent blocks. The reduction of the high frequency components in horizontal and vertical direction can lead to this annoying artifact. This is another situation where the basic DCT functions become visible.

2.2.2 Artifacts in Image Color

2.2.2.1 Color Bleeding

The *color bleeding* can be described as a smearing of chrominance information along high-contrast chrominance areas and it is equivalent as distortion to blurring in the luminance channel. The color bleeding is due to a coarse quantization of high frequency chrominance coefficients. It results in the representation of the chrominance components with only the lower frequency coefficients. Color bleeding results from coarse or even zero quantization of higher order AC transform coefficients for the color channels. Due to the colour sub-sampling schemes, this kind of distortion has an annoying influence on the color information in the whole macro-block, named chrominance ringing.

2.2.3 Artifacts related to motion

2.2.3.1 Mosquito Noise

The *mosquito* effect is a temporal artifact seen mainly in smoothly textured regions as a fluctuations of luminance/ chrominance levels around high contrast edges, or moving objects, in a video sequence. This effect is related to the high frequency distortions introduced by both the ringing effect, and the prediction error produced by the motion compensated mismatch artifacts. The mosquito noise also affects stationary areas within a moving scene, characterized by high spatial frequencies. Flickering in the luminance and chrominance channels may be observed.

2.2.3.2 Judder

This kind of artifacts is typical in teleconference systems and phone applications and it can be seen as an image flipping in the direction of movement. The necessary bandwidth for the transmission is a function of the change in video content but in this sort of applications bandwidth is mostly limited and therefore especially in fast moving scenes the image sequence is cut into discrete snapshots to fit the temporal bandwidth of the source.

2.2.4 Binocular Artifacts

2.2.4.1 Cross-distortion

Cross-distortion is caused by asymmetrical video coding in both temporal or spatial domains. In the case of temporal asymmetric video coding, one channel has lower frame-rate than the other; otherwise, in case of spatial asymmetric video coding one channel has lower resolution than the other.

Cross distortions occur when the bit budget of the right and left picture presented to an observer is not equal. It leads to a decreased overall quality of the image or sequence, whereas the human visual system fortunately tries to compensate this mismatch. If the difference in quality between the left and right image becomes too big, a wrong or even distorted depth is perceived.

2.2.4.2 Cardboard effect

The *cardboard* effect is typically caused by image acquisition or compression parameters resulting in a coarse quantization of the disparity or depth maps. Due to it, the objects appear flat as if the scene is divided into discrete depth planes. The flattening of the objects in a scene evokes an unnatural depth percept.

2.2.4.3 Depth Bleeding

More unusual coding artifacts that have impact on depth perception are *depth bleeding* and *depth smoothing*. Depth bleeding affects the depth channel, it is similar to color bleeding. Depth smoothing is due to asymmetric compression or resolution of the depth channel.

2.3 Quality metrics

As we have illustrated in the previous sections, stereo video stream can be subject to several distortions during the capturing, representation, coding, transmission or visualization steps. Any of these phases may result in a degradation of visual quality. With the deployment of Blue-Ray, DVD, personal computer and communications technologies,

video files are disseminated in many different formats from CD to AVI (Audio Video Interleave) and other file types. Video technology has drastically changed over the years from the black and white television to the chance, nowadays, to create recordable media with high resolutions in our own living room. Video content, today, can be transferred from place to place on multiple types of media. The providers and consumers of video material have kept pace with the times by increasing their expectations of quality. The problem that the industry faces now is how we measure video quality, guarantee delivery of high quality video and prove that the quality is delivered at the promised level.

There are two primary ways to measure video quality. The first is Subjective Quality Assessment. This method exploits structured experimental designs and "human" participants to evaluate the quality of the video presented when compared with a given reference. The second is Objective Quality Assessment. This measures physical aspects of a video signal and considers both the physical aspects and psychological issues. Both types of testing are far from an exact science but they have proven to be very useful tools.

In the following we will discuss some objective and subjective quality metrics used for evaluation of video quality. These metrics are generally used for modeling 2D video quality but since there are not specific quality metrics for stereoscopic video, the conventional quality metrics, PSNR, SSIM and VQM, will be adopted for objective evaluation; at the contrary, the mean opinion score, MOS, will be analyzed for subjective assessments [24].

2.3.1 Objective Quality Evaluation

An objective image quality metric can be useful for different applications. First, it can be employed to dynamically monitor and adjust image quality. Second, it can be used for optimizing algorithms and parameters settings of image processing architecture, [25]. Finally, it can be a benchmark for an image quality systems and algorithms.

Objective quality metric can be classified according to the availability of the original image with which the processed image has to be compared.

If the reference image is known, then the metric is said to be *full-reference*, otherwise if the reference image is not available, the quality assessment is *no-reference* or *blind*. A third

case can be represented by partial availability of the reference image (only the extracted features are present), and in this case, the metric is said to be *reduced-reference*.

2.3.1.1 PSNR

A useful and often used metric is peak-to-peak signal-to-noise ratio, PSNR.

This image quality index is defined as the ratio between the maximum signal power and noise power that can interfere with the fidelity of the representation,[26]. The PSNR is most commonly used in the field of image processing as a measure of quality of reconstruction of lossy compression codecs. The signal in this case is the the reference image, and the noise is the error introduced by compression. It is strongly used for comparison of compression codes, because it gives an approximation to human perception of reconstruction quality. However it can happen that in some cases, one reconstruction may appear to be closer to the original than another, even though it has a lower PSNR (a higher PSNR means higher quality reconstruction). This metric is not always so reliable and lot of attention must be taken with the range of validity of this metric; it is only conclusively valid when it is used to compare results from the same codec (or codec type) and same content. It is most easily defined via the mean squared error (MSE) which for two $m \times n$ images I and K (only luminance is considered) where one of the images is considered a noisy approximation of the other is defined as:

$$MSE = \frac{1}{mn} \sum_i^m \sum_j^n ||I(i, j) - K(i, j)||^2 \quad (2.1)$$

The PSNR is defined as:

$$PSNR = 10 \cdot \log \left(\frac{MAX_I^2}{MSE} \right) = 20 \cdot \log \left(\frac{MAX_I}{\sqrt{MSE}} \right) \quad (2.2)$$

Here, MAX_I is the maximum possible pixel value of the image. When the pixels are represented using 8 bits per sample, this is 255. For color images with three RGB values per pixel, the definition of PSNR is the same except that the MSE is the sum over all squared value differences divided by image size and by three. Typical values for the PSNR in lossy image and video compression are between 30 and 50 dB, where higher is better. Acceptable values for wireless transmission quality loss are considered to be about 20 dB to 25 dB.

This metric is appealing because it is simple to calculate and has clear physical meaning. However, it does not match very well the perceived visual quality.

2.3.1.2 SSIM

The Structural Similar Measure (SSIM) [25] compares local patterns of pixel intensities that have been normalized for luminance and contrast; in particular, the SSIM index is a combination of three different attributes: luminance, contrast and structure, [27].

The three components are combined to yield an overall similarity measure. Considering two windows x and y of a common size $N \times N$, this metric holds:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\mu_x^2 + \mu_y^2 + C_2)} \quad (2.3)$$

where μ_x is the average of x , μ_y is the average of y , σ_x is the variance of x , σ_y is the variance of y and σ_{xy} is the covariance; all these parameters are computed within a local 8x8 square window. $C_1 = (K_1, L)^2$ and $C_2 = (K_2, L)^2$ are two variables to stabilize the division with weak denominator; L is the dynamic range of pixel-values and $K_1, K_2 \ll 1$ are small constants.

An important point is that the three components are relatively independent. For example, the change of luminance and/or contrast will not affect the structures of the image.

For image quality assessment, it is useful to apply the SSIM metric locally rather than globally. In fact, localized quality measurement can provide a spatially varying quality map of the image, which delivers more information about the quality degradation of the image, resulting more useful in some applications.

2.3.1.3 VQM

The National Telecommunication and Information Administration (NTIA) has developed a General Model for estimating video quality and its associated calibration techniques (e.g. estimation and correction of spatial alignment, temporal alignment, and gain/offset errors), [28]. The NTIA's research has focused on developing a technology independent parameters that model how people perceive video quality. These parameters have been

combined using linear models to produce estimates of video quality that closely approximate subjective test results.

The General Model uses a reduced-reference system that provides an estimate of the overall impressions of video quality, [29]. Reduced-reference metrics systems uses low-bandwidth features that are extracted from the source and destination video streams. Real-time in-service quality can be measured by reduced-reference system (provided an ancillary data channel available to transmit the extracted features) and this is necessary for tracking dynamic changes in complex scene and/or transmission systems.

The General Model and its associated calibration techniques comprise a complete automated objective video quality measurements systems. The calibration of the original and the processed video streams includes spatial alignment, valid region estimation, gain and level offset calculation, and temporal alignment. VQM calculation involves extracting perception-based features, computing video quality parameters to construct the General Model.

VQM can be computed using various models based on certain optimization criteria. These models include:

1. Television
2. Video-conferencing
3. General
4. Developer
5. PSNR

2.3.2 Subjective Video Quality Measurements

It is clear that it may not be possible to fully characterize system performance by objective means; consequently, it is necessary to supplement objective measurements with subjective measurements. Subjective assessment uses human subjects (real end users) to evaluate, compare or assess the quality of images under test. Subjective assessment is the most reliable way to determine actual image quality, and cannot be replaced with objective

testing.

In this case, the mean opinion score (MOS) provides a numerical indication of the perceived quality of received stereo video content after compression and/or transmission.

2.3.2.1 Mean Opinion Score (MOS)

The MOS is expressed as a single number that can range from 1 to 5 or from 1 to 100. The lowest number represents the worst case while the biggest number corresponds to the highest perceived video quality. MOS tests for video are specified by ITU-R BT.500-11 [30].

The MOS is generated by averaging the results of a set of standard, subjective tests where a number of viewers rate the viewed video quality of test sequences.

Compressor/decompressor (codec) systems and digital signal processing are commonly used in voice communications, and can be configured to conserve bandwidth, but there is a trade-off between voice quality and bandwidth conservation. The best codecs provide the most bandwidth conservation while producing the least degradation of video quality. A drawback of obtaining MOS estimations is that it may be more time-consuming and expensive as it requires hiring experts to make estimations.

At the aim of evaluating the quality of a stereo video content, objective and subjective quality assessments can be performed. The goal of objective video quality assessments is to develop a quantitative measure that can automatically predict perceived video quality.

2.4 Experimental Results

As illustrated in Section 2.3, subjective evaluation testing is used to measure the effect of distributed video coding artifacts on the perceived quality of the reconstructed stereoscopic sequence,[21]. The obtained results have also been compared with the quality evaluated by using three 2-D video objective quality models namely PSNR, VQM and SSIM.

The Video Quality Metric adopted here, is derived by Watson's DCT-based metric (DVQ) [31] [32] video quality evaluation and considers only the luminance of the video sequence

and takes into account a human Spatial Contrast Sensitivity Function (SCSF).

In the following we show the results for the DIPLODOC 3D "road stereo sequence" [33], that is 240x320 pixels and 15 frames per second; 201 frames were used for the sequence.

In the DVC architecture, a GOP equal to 2 has been analyzed, where the right view has been Wyner-Ziv coded while the left view is coded with a conventional H.264/AVC. As usual for WZ coding, only luminance data has been coded; the total bit-rate includes the luminance rate for the WZ frames and key frames for the right view to be coded since the left view is always the same.

16 non-expert observers (8 males and 8 females) participated in the experiments and were asked to rate the video sequence perceived quality according to the method proposed in [30]: the subjective ratings for the coded stereoscopic sequences have been scaled into a linear opinion score scale, which ranges from 0 (bad quality) to 100 (excellent quality). The observers were also asked to evaluate the kind of annoying visual artifacts for each shown sequence.

The bit-rate of the 3-D coded sequence have been systematically varied and 5 reference bit-rates were considered: 10, 25, 100, 500, 2100 and 5000 Kbit/sec, i.e. ranging from a low bit-rate transmission case where the DVC approach results to be more suitable to a high bit-rate channel case, where a conventional H.264/AVC coder usually results to be more appropriate. For each reference bit rate, the H.264/AVC 3-D coded sequence and the DVC 3-D coded sequence have been considered. The stimulus set contains 15

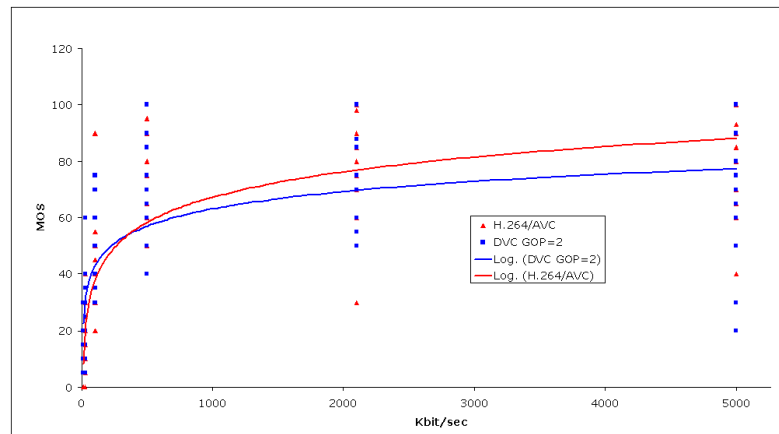


Figure 2.3: MOS scores for perceived stereo video quality.

coded sequences and the original, uncompressed sequence is used as the reference in the evaluation test. The set is randomized and presented sequentially.

Figure 2.3 shows MOS scores for the overall perceived quality, while Figures 2.4, Figure 2.5 and Figure 2.6 show the rate distortion (RD) performance for the analyzed WZ stereo coding architecture, by respectively considering the PSNR, VQM and SSIM quality models. These metrics have been adopted to evaluate the quality of the decoded 2-D right sequence [24].

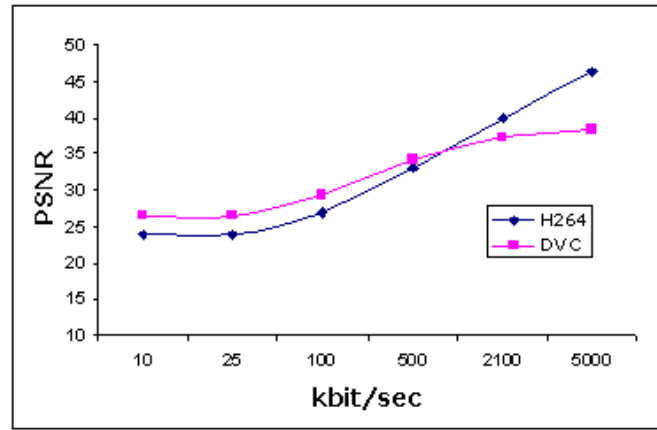


Figure 2.4: RD performance by PSNR evaluation. PSNR is averaged on the whole right sequence.

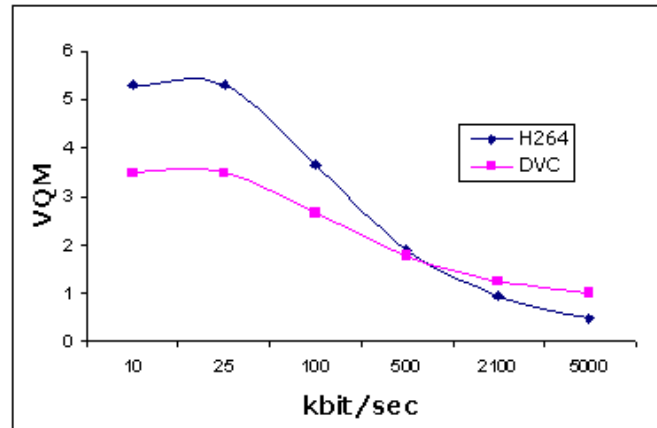


Figure 2.5: RD performance by VQM evaluation. VQM is averaged on the whole right sequence.

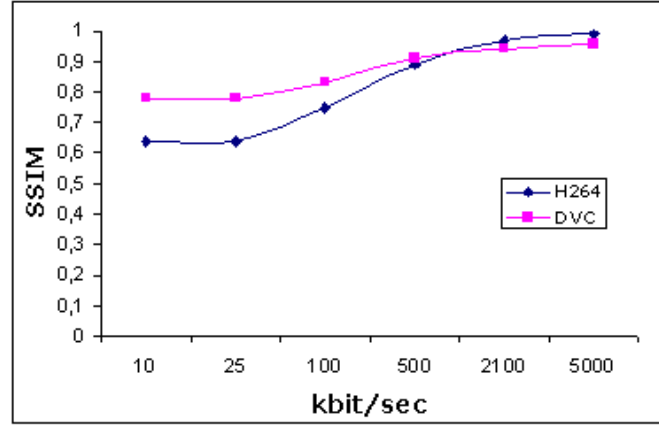


Figure 2.6: RD performance by SSIM evaluation. SSIM is averaged on the whole right sequence.

A number of interesting conclusion can be drawn. Given that DVC schemes are more suitable for low bit-rate channels because less amount of data need to be transmitted, Figures 2.3 supports the above statement from a visual perspective point of view. On the contrary, the reversal of the trend at about 400 Kbit/sec shows that a conventional H.264/AVC coder results to be more appropriate at high bit-rate even if DVC approach would be still preferred in some cases due to the advantage of low-complexity encoders. Figures 2.4, 2.5 and 2.6 report the same trend for the WZ coded right sequence.

A *flickering* effect has been noticed on the DVC coded sequence only at low bit-rate due to the alternate of low-quality decoded H.264/AVC key frames and higher quality decoded WZ frames. Obviously this effect is not anymore noticeable at high bit-rate when the quality of the decoded key frames and the quality of the decoded WZ frames becomes similar. Due to this considerations, we have increased the GOP's length in the DVC method and included these coded sequences in the set evaluated by the observers. In order to show that, from a perceived quality point of view, when the bit-rate becomes lower, it is more convenient to use a longer GOP, GOP of length 9 was considered for reference bit-rate 25 Kbit/sec. Average MOS values reported in Table 2.1 shows that the human eye perceives less annoyance when, at low bit-rate, it is presented a sequence coded with a longer GOP. Average values of PSNR, VQM and SSIM, computed on the whole right sequence, confirm the above statement from an objective point of view.

For the reasons described above, the *blocking* effect typical of a conventional H.264/AVC

Table 2.1: Objective/subjective results for a low bit-rate channel.

	PSNR [dB]	VQM	SSIM	MOS
25 H.264/AVC	23,76	5,31	0,64	16
25 DVC GOP=2	26,5	3,48	0,78	27
25 DVC GOP=9	29,8	2,36	0,87	29

at low bit-rate was less noticeable in the DVC decoded sequence. In fact this effect only affects the key frames of the right sequence and the left sequence. We believe that a symmetric approach where both the views have a limited amount of key frames would not let the observer perceive a blocking effect.

A *blurring* effect has been also noticed on the DVC coded sequence due to the particular WZ coder that is used. In the analyzed scheme, a turbo decoder with puncturing rate equal to 1/3 has been used. Note that the turbo decoder performance is strictly dependent on the amount of parity bit planes used to reconstruct the WZ frames. The adopted turbo decoder was always able to reconstruct the WZ frames even if in very uniform areas to be decoded a blurring effect was noticed by few careful observers. This effect is to be related not only with the used WZ decoder but also with the accuracy of the side-information: this is the actual challenge in the most recent stereo DVC approaches that have to exploit the temporal and inter-view correlation.

The combination of the above effects let few participants note a *jerkiness*-like effect for low bit-rate cases: this corresponds to the perception of originally continuous motion as a sequence of distinct "snapshots". In fact, when a sequence of still frames is perceived by the human brain at a continuous rate, intermediate images are interpolated and the observer subjectively appears to see continuous motion that in reality does not exist.

Finally, in the lowest bit-rate case, some observers perceived a loss of stereo vision either in H.264/AVC coded sequences and in WZ ones.

2.5 Conclusions

In this chapter, a stereoscopic video quality assessment has been conducted for the evaluation of a distributed video coding scheme. The considered coding approach has been compared with conventional H.264/AVC. The objective evaluations showed that DVC has a better quality than H.264/AVC for lower bit-rate; at the contrary, for higher bit-rate conventional stereo video coders result more powerful. These results have been validated by subjective tests.

Objective and subjective video quality experiments have been carried out to evaluate video artifacts introduced in a stereo distributed video coding system. The most relevant artifacts that have been noticed are flickering, blocking and blurring.

The obtained results can be considered a starting point for an extensive analysis of the DVC stereo perceived quality that takes into account different DVC schemes.

Chapter 3

Distributed Joint Source-Channel Coding

3.1 Introduction

This chapter presents a distributed joint source-channel 3D video coding system. Here, instead of using the classical two steps scheme, we adopt a single source-channel encoder for both compression and protection resulting in a distributed 3D video coding scheme.

Shannon's source-channel separation principle [34] states that, in the limit of large block-length and for a large class of communication setups, the optimal performance can be approached by adopting a tandem source and channel coding schemes.

Driven by the separation principle, modern communication systems have been developed according to a rather rigid layered architecture [35]: the source coding is implemented at the application layer, while the channel coding is designed and optimized at the physical layer, [36], [37]. While a separated (layered) architecture has the advantage of modular system design, allowing that a great variety of services can exploit a common data network infrastructure, there are cases where a Joint Source-Channel Coding (JSCC) approach can be very appealing. On one hand, there exist several relevant multiterminal settings where the separated approach is known to be suboptimal [34]. On the other hand, even in standard point to point channels where Separated Source-Channel Coding (SSCC) is asymptotically optimal, the use of independently designed source and channel codes may

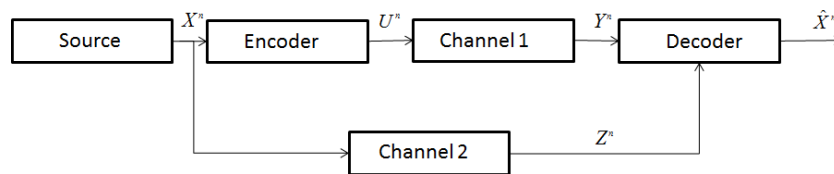


Figure 3.1: Channel model with uncoded side information

result in poor performance in the practical non-asymptotic regime of finite block length and low complexity encoding/decoding, [38].

The purpose of the work presented in this chapter is to outline a new pragmatic approach to the of DVC-based Joint Source-Channel 3D video coding scheme for noisy channel that preserves the perceived visual quality while guaranteeing a low computational complexity. The mathematical framework will be fully detailed and the tradeoff among redundancy and perceived quality and quality of experience will be analyzed with the aid of numerical experiments.

3.2 Theoretical Background

Let's consider a general model where two independent channels operate in parallel as depicted in Figure 3.1. If the inputs to both channels are encoded, the Shannon coding theory states that it is possible to transmit reliably the source information if source entropy rate is below the sum of each channel capacities, $C_1 + C_2$; otherwise if source entropy rate is above this values, then no source information can be reliably transmitted. In the model we consider the information passing through channel 2 is not encoded and this can represent a practical scenario such as Slepian-Wolf case.

As demonstrated in [39], the source can be reliably transmitted if its conditional entropy rate (given the output of channel 2) is below the capacity C_1 and otherwise if the conditional entropy exceeds C_1 , no reliable transmission is guaranteed. This condition for a reliable transmission is equivalent to the source entropy rate being below the sum of channel 1 capacity and the mutual information rate of channel 2. So the source information is related to channel 2 because if input/output mutual information is maximum then source information can be reliably transmitted when entropy rate is below $C_1 + C_2$.

Be X^n , a word of length n , the output of the source, and suppose that the decoder selects a word \hat{X}^n upon observations of the outputs of both channels. As demonstrated in [39], if the source and the channel 2 are such that (\mathbf{X}, \mathbf{Z}) are jointly ergodic and stationary, and assume that $H(X/Z)$ is the conditional entropy rate of X given Z , then it has been proofed that:

1. the source is transmissible if

$$H(X/Z) < C_1 \tag{3.1}$$

2. the source is not transmissible if

$$H(X/Z) > C_1 \tag{3.2}$$

This result suggests that the information rate of the source can be seen as the sum of two components:

$$H(X) = H(X/Z) + I(X; Z) \tag{3.3}$$

such that the first component is encoded and then pass through channel 1 and the second component is transmitted through channel 2 without encoding first. Even though, the second part is not encoded, the source can be reconstructed with a arbitrary small probability of error at the decoder.

Moreover, if source is matched to channel 2 in the sense that it maximizes its input/output mutual information, then it turns out that it is possible to transmit information at the rate $C_1 + C_2$ without encoding information that passes through channel 2, (side information). Equivalently, the Slepian-Wolf limit in this noisy channel case is $H(X/Z)/C_1$, with $C_1 \leq 1$. A separation theorem for lossy source-channel coding with decoder side information, i.e., the noisy channel Wyner-Ziv case, is given in [40]. In the separation theorem given in [39], the conditional entropy $H(X/Z)$ is replaced by the Wyner-Ziv rate-distortion function $R_{WZ}^*(D)$.

3.2.1 Distributed Joint Source-Channel Coding

When the channel is noisy in the SW problem, source-channel coding with side information is needed. According to Shannon theorem, a reliable transmission can be accomplished by

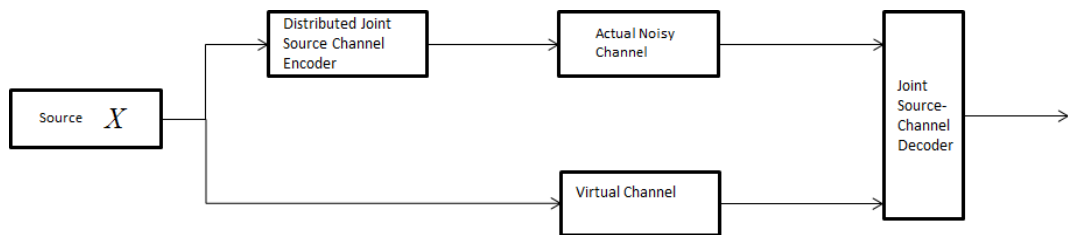


Figure 3.2: The system model for Distributed Joint Source-Channel Coding

separate source and channel coding. However although tandem designs are asymptotically optimal, practical design are expected to perform better when employing joint source channel coding. Following the separate design, first the sources are Wyner-Ziv coded, where a WZ code can be considered a concatenation of a quantizer and a SW code, where the quantizer design with side information is an essential part. In practice, the source and its side information are assumed to be connected by a virtual error-prone channel. Then, the Slepian-Wolf coded bits are protected with a channel code against the distortion they are going to encounter when transmitted through a noisy channel. Since SW coding is essentially channel coding, it is more meaningful to combine two channel codes, the ones used for SW coding and the channel coding into a single channel code and use just this channel code for joint source-channel coding with side information. The main idea of underlying such a joint scheme is to view the source-channel bits as the parity bits of a systematic channel code and to consider an equivalent channel coding problem over two channels. The first channel is the actual noisy channel through which the source-channel coded bits (the parity bits of the systematic channel code) are sent to the decoder, and it describes the distortion experienced by the parity bits of the code. The second channel is the "virtual" correlation channel between the source (the systematic bits of the channel code) and the side information available at the decoder [41] [42]. In the following a brief overview of the joint source channel Slepian-Wolf encoder is presented.

Let us consider $X = [X_1, X_2, \dots, X_n]$ and $Y = [Y_1, Y_2, \dots, Y_n]$ where X_i 's and Y_i 's are i.i.d. equiprobable binary random variables. In addition, X_i and Y_i are correlated so that $\Pr[X_i \neq Y_i] = p < 0.5$. Y is available lossless at the joint decoder and we try to compress X as efficiently as possible [43],[44]. Since the rate used for Y is its entropy

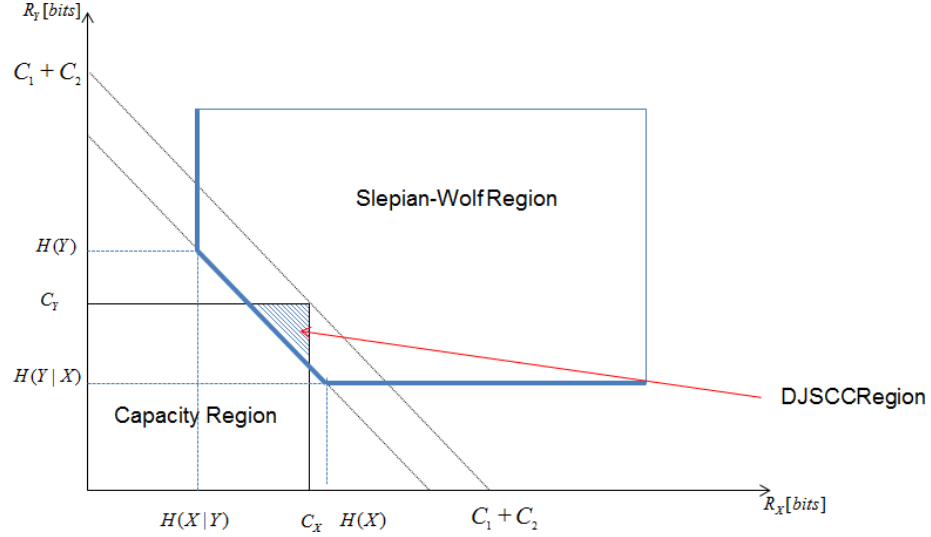


Figure 3.3: Achievable rate region defined by Slepian-Wolf bounds

$nR_y = nH(Y_i) = n$ bits, theoretically the minimum rate required to make X decodable given reference Y at the decoder is the conditional entropy $H(X_i|Y_i)$. The conditions for two correlated sources X and Y can be expressed as:

$$\begin{aligned} H(X|Y) &< C_X \\ H(Y|X) &< C_Y \\ H(X, Y) &< C_X + C_Y \end{aligned} \tag{3.4}$$

Assuming that $C_X = C_Y = C$, it can be obtained that:

$$\begin{aligned} R_X &\geq H(X|Y)/C \\ R_Y &\geq H(Y|X)/C \\ R_X + R_Y &\geq H(X, Y)/C \end{aligned} \tag{3.5}$$

where R_X and R_Y are respectively the encoding rate of X and Y . In figure 3.3 the achievable rate defined by Slepian-Wolf bounds and the DJSCC rate region are shown.

3.3 Related Works

The most direct method for implementing Distributed Joint Source-Channel coding (DJSCC) is sending additional bits for general Distributed Source Coding (DSC) scheme to approach the theoretical bound. Based on this idea, efficient designs representing the state of art

are reported in this section. Reference [45] presents the case where decompression must be done from compressed data corrupted by Additive White Gaussian Noise (AWGN). Turbo codes are used by finely designing the matrices of the two constituent encoders. The design of [43] exploits systematic Irregular Repeat-Accumulate (IRA) codes [46] for DJSCC. The main idea is to view the system as transmitting the source over two channels. The first one is the actual channel through which the source-channel coded bits are sent to the decoder and describes the distortion that affects the parity bits of the systematic IRA codes. The second channel is the enhanced actual channel and it could be either a combination of the actual channel and the correlation channel. This channel describes the distortion of the information bits.

The systematic part goes through the binary symmetric correlation channel and the parity bits through the actual channel. The feasibility of designing different channel conditions for the systematic and the parity part separately is the main advantage in joint source-channel coding with side information. The simulation results confirm the superior performance to the turbo codes scheme.

Using the IRA codes for pre-coding, Raptor codes [47] were designed for DJSCC [42],[48], over packet erasure channels. The rateless property can guarantee the success in decoding regardless of the packet loss ratio. The IRA precoder is followed by an LT code which guarantees the rateless property of the overall Raptor code, meaning that a limitless stream of packets can be generated by the encoder. The use of Raptor encoder leads to a minimization of the number of packets that the decoder has to receive in order to correctly decode beyond the Slepian-Wolf compression limit. They varied the rate of IRA precode and introduced a bias towards selecting IRA parity symbols versus systematic symbols in forming the bipartite graph of the LT code. This bias is motivated by the fact that a correlated version of IRA systematic bits is already available as side information at the decoder, and its optimization is embedded in the overall Raptor encoder design. For the decoder side, an iterative soft-decision that combines the received packets and the side information to perform joint decoding is performed. However, they didn't give a calculable method to determine the optimal design.

In [49], the authors propose a scheme based on distributed arithmetic coding (DAC) over

noisy channels. The proposed encoder is a combination of DAC and arithmetic coding with forbidden symbol. The decoding side is based on a sequential decoder and it introduces an additive MAP metric to rank the various decoding attempts. At each step, the value of the coded sequence is used to select the interval corresponding to a decoded symbol. The selected interval is subdivided according to the known probabilities. In [50], the case of compression of single and correlated binary source using punctured turbo codes is proposed. When dealing with correlated sources, both sources are compressed independently of each other. The encoder structure can be seen as "super" turbo encoder, with each constituent encoder being a turbo code. The decoder is based on message passing algorithm over the graph associated to each constituent turbo encoders. The decoding phase proceeds in a traditional fashion except that no noise is considered here. However, additional extrinsic information is also exchanged between both turbo decoders. To achieve a good performance, the puncturing scheme is carefully selected: half of the systematic bits for each constituent turbo encoder are used. The bit stream from the first source is punctured in a uniform way while the second bit stream is punctured following the interleaver. The design of an efficient joint source-channel coding scheme based on LDPC code is reported in [44]. The architecture they proposed is inspired by [51] for the parallel channel model. The basic idea is dividing the encoded codeword into several fractions in order to process them separately. Since it is unnecessary to transmit all the codeword to the decoder for the existence of correlation, so for one source only a fraction of information bits and a fraction of parity bits is sent through the noisy channel. At the joint decoder, the received different parts of the information bits from different encoders will re-produce an integrated codeword for decoding, and they act as side information for each other. The simulate results verify the limit-approaching performance of the proposed scheme and show better results than [52].

3.4 Distributed Joint Source-Channel Coding for 3D Videos

In this section we present a distributed 3D video coding with joint source-channel coding based on Turbo-code, [53]. The reason of employing this particular codes lies in the

fact that turbo codes are powerful channel codes that allow to get close to Slepian-Wolf bound. Here distributed source-channel video coding is achieved by puncturing parity bits. Puncturing is the process of deleting some parity bits from the codeword according to a puncturing matrix. It represents a tradeoff between rate and performance. For achieving a joint source-channel scheme, the main idea is to consider an equivalent channel coding problem over two channel. The first channel is the actual noisy channel through which the source-channel coded bits (the parity bits of the systematic channel code) are sent to the decoder, and it describes the distortion experienced by the parity bits of the code. The second channel is the "virtual" correlation channel between the source (the systematic bits of the channel code) and the side information available at the decoder.

At the decoder side, an iterative soft-decoding system that combines received packets and the side information to perform joint decoding is considered. It consists of two constituent decoders: the Soft-In/Soft-Out (SISO) Channel decoder and the Soft-In/Soft-Out (SISO) Source Decoder[54]. The information that is passed between the constituent decoders is log-likelihood ratios of the databits.

3.4.1 Turbo codes

It is important to find the optimal trade off between the amount of data transmitted and the quality of the decoded stream.

To this aim, the use of turbo codes [55] is often preferred as they allow to send the minimum amount of data while guaranteeing near channel capacity error correcting performance [56].

In the DVC scheme, depicted in Figure 3.4, after a Wyner-Ziv frame is transformed and quantized, it is separated into bit-planes, which are fed one-by-one to a turbo encoder. The turbo encoder consists in a Parallel Concatenation of Recursive Systematic Convolutional Codes (PC-RSC) in addition to a pseudo-random interleaver to spread burst errors. Each RSC encoder produces two output, the systematic bits S_i and the parity bits P_i , where $i = 1, 2$.

The systematic bits of the encoded data are discarded while all generated parity bits

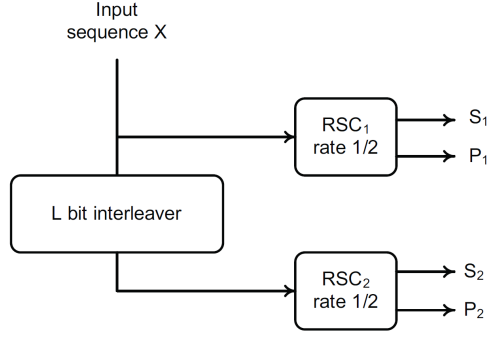


Figure 3.4: Turbo encoder structure in DVC approach.

are stored at the encoder side in a buffer and transmitted in the decoding phase upon the decoder's request via a feedback channel. As for Figure 3.5, in order to reconstruct the data, the iterative Maximum A Posteriori (MAP) turbo decoder uses the parity bits requested to the encoder and the systematic bits, directly extracted from the side information which can be seen as a corrupted version of the original data. A Laplacian distribution is assumed for the difference between the original data and the side information [11] [57]. The parity bits are requested until they are exhausted or an acceptable probability of symbol error is reached: hence, depending on the accuracy of the systematic bits, additional parity bits are requested, thus leading to an efficient use of the band.

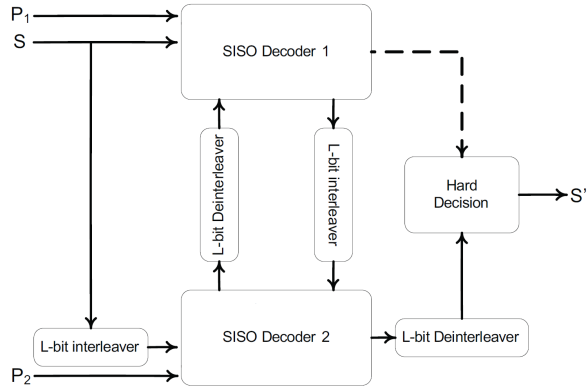


Figure 3.5: Turbo decoder structure in DVC approach.

The aim is to design an algorithm where each encoder has high compression perfor-

mance to minimize transmission costs, low computational complexity to preserve battery life and robustness to avoid effects of channel loss. Traditional video coder as MPEG and H.26x achieve high compression but have high complexity and are sensible to prediction mismatch in case of packet loss. On the other hand MJPEG is robust but has poor compression performance.

3.4.2 Joint Source-Channel Decoding

The proposed scheme, [53], is inspired from [44] for the separate channels model. The basic idea is to divide the encoded codeword into two fractions and process them separately. Let us consider two memoryless sources X and Y which are statistically dependant to each other with cross-over probability $P[Y \neq X/X] = p_{XY}$ and $P[X \neq Y/Y] = p_{YX}$. The k -bit sequence of the sources X is encoded independently using turbo-code (k, n) ; thus, the codeword of X will be represented of systematic bits plus parity bits P_i ($i = X, Y$) where $P_i = k - n$. However it not necessary to transmit the entire codeword to the decoder because of correlation between the two sources. So for source X only k systematic bits and $b_i P_i$ parity bits are transmitted trough the noisy channel with constraints that $0 \leq b_i \leq 1$, $\sum_{i=1}^2 b_i = 1$. The DJSCC encoding rate R_X for the source X is:

$$R_i = \frac{k + b_i(k - n)}{k} \quad (3.6)$$

In Figure 3.6, the separate channel model is drawn for the source X . As explained before, the source Y is available at the decoder for reconstructing the Wyner-Ziv frame.

At the joint decoder, the received different parts of the information bits from different encoders will re-produce the entire codeword for decoding, as they act as side information for each other. For source X , at the aim of obtaining a complete codeword for decoding, the corresponding systematic part is replaced by the side information which is received from Y . This method can be similarly applied in decoding for Y . At the decoder side, received packets and the side information are combined by an iterative soft-decoding system to perform joint decoding. It consists of two constituent decoders: the Soft-In/Soft-Out (SISO) Channel decoder and the Soft-In/Soft-Out (SISO) Source Decoder. The information that is passed between the constituent decoders is log-likelihood ratios of the databits.

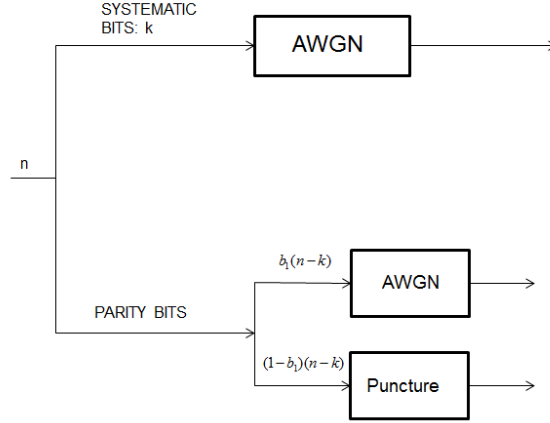


Figure 3.6: parallel channel model for DJSCC scheme

3.5 Experimental Results

In order to evaluate the performance of the proposed DJSCC approach, we tested it on DIPLODOC 3-D "road stereo sequence" [33]. We encode 201 frames, with a capture rate of 15 frame per pixels, and the resolution of each frame is 240x320.

In the DVC architecture, a GOP equal to 2 has been analyzed, where the right view has been Wyner-Ziv coded while the left view is coded with a conventional H.264/AVC. As usual for WZ coding, only luminance data has been coded; the total bit-rate includes the luminance rate for the WZ frames and key frames for the right view to be coded since the left view is always the same.

The coded video stream is transmitted over AWGN channel with $SNR = 8dB$. The turbo code rate is $R^c = k/(k+P) = 1/3$ and the total encoding rate is $R = R_X + R_Y = 1/R^c = 3$. We considered a symmetric rate case where $b_1 = b_2 = 0.5$ (JOINT1/2) and another case where $b_1 = 0.7$ and $b_2 = 0.3$ (JOINT) and we compared rate distortion performances of our joint source channel coding method against tandem method where source coding

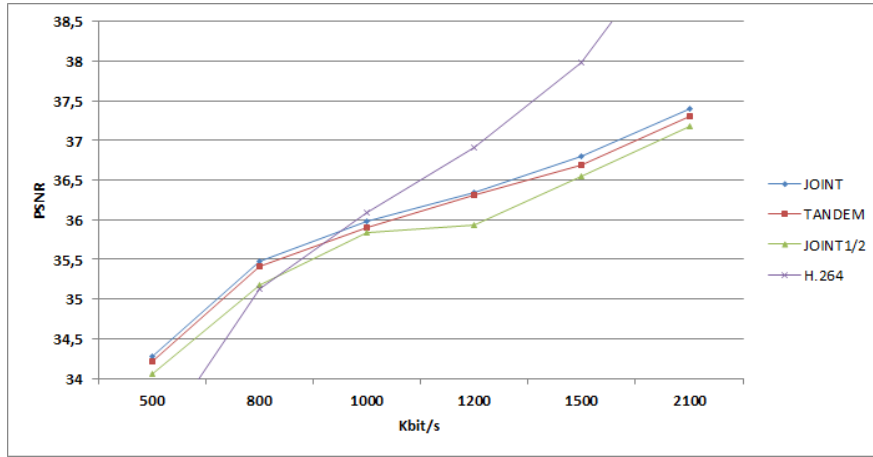


Figure 3.7: Rate-distortion comparison with different schemes.

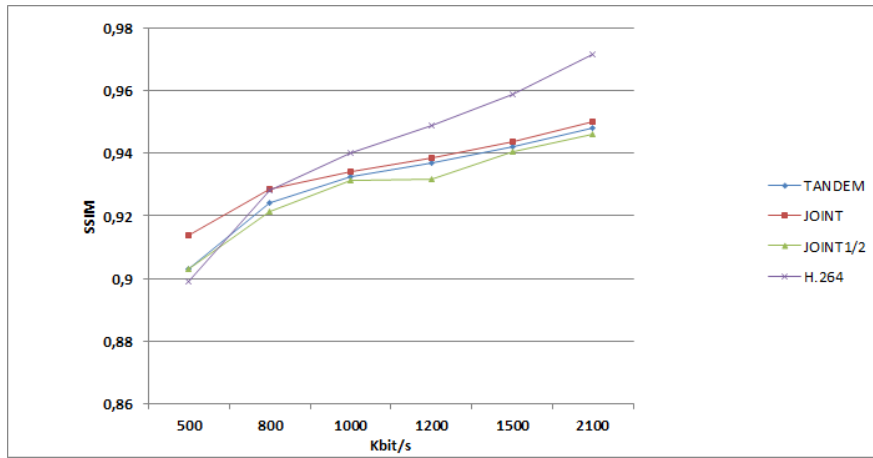


Figure 3.8: SSIM comparison with different schemes

and channel coding are separated. We have also compared Joint DVC approach with H.264 coding. The bit-rate of the 3-D coded sequence have been systematically varied and 6 reference bit-rates were considered: 500, 800, 1000, 1200, 1500 and 2100 Kbit/sec. PSNR and SSIM performances show that joint source-channel coding performs better than tandem approach when the asymmetric case (JOINT) is considered, i.e. $b_1 = 0.7$ and $b_2 = 0.3$, as shown in figures 3.7 and 3.8; on the contrary when the symmetric case is evaluated performance assessments show that tandem approach is better.

In figure 3.10, rate distortion of our approach has been compared to H.264 coding. The bit-rate of the 3-D coded sequence have been systematically varied from 10 to 20000 Kbit/s ranging from a low bit-rate transmission case where the DVC approach results to

be more suitable to a high bit-rate channel case, where a conventional H.264/AVC coder usually results to be more appropriate. For each reference bit rate, the H.264/AVC 3-D coded sequence and the DVC 3-D coded sequence have been considered.

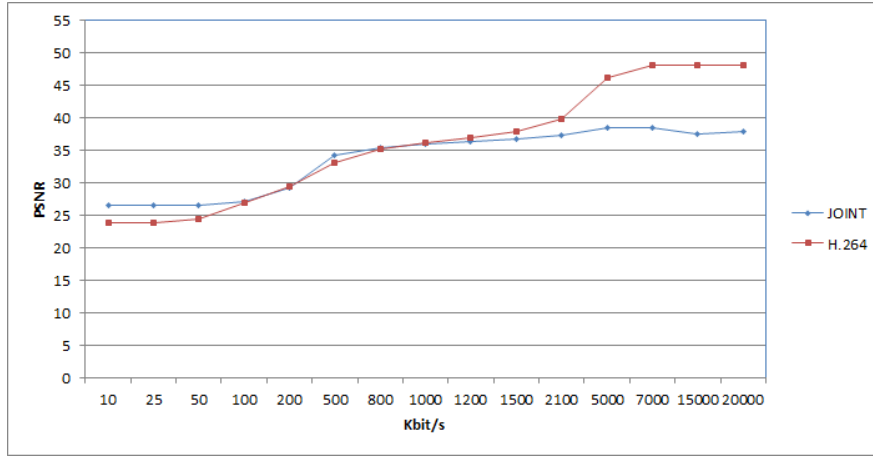


Figure 3.9: RD comparisons by PSNR evaluations between our proposed method and conventional H.264 coding. PSNR is averaged on the whole right sequence.

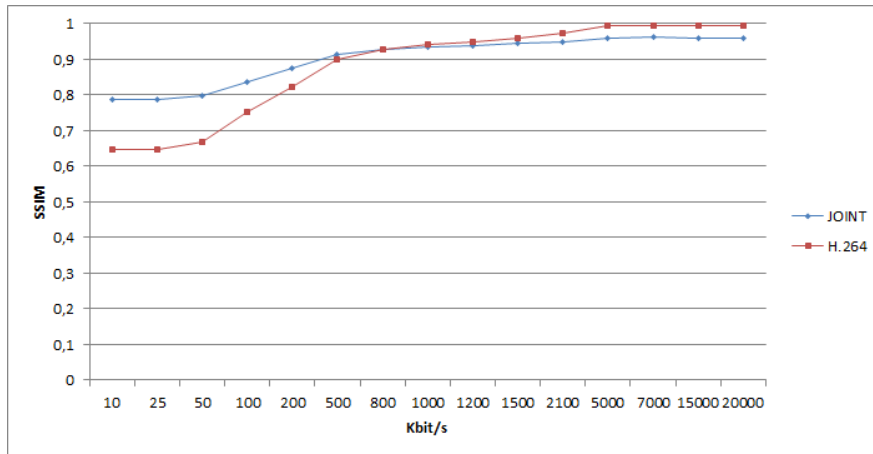


Figure 3.10: RD comparisons by SSIM evaluations between our proposed method and conventional H.264 coding. SSIM is averaged on the whole right sequence.

3.6 Conclusions

In this chapter, the joint source-channel coding problem for the DVC approach has been addressed. It has been also presented a novel approach that performs distributed joint

source channel coding for 3D videos. At this aim, we have applied turbo code in joint source-channel coding. The reason lies in the fact that they allow to send the minimum amount of data while guaranteeing near channel capacity error correcting performance. Performance results show a better trend of DJSCC compared to conventional tandem method when the asymmetric case is considered for joint coding.

The considered coding approach has been also compared with conventional H.264/AVC. The performance evaluations showed that JDSCC has a better quality than H.264/AVC for lower bit-rate; at the contrary, for higher bit-rate conventional stereo video coders result more powerful.

Chapter 4

Multiview Distributed Video Coding

4.1 Introduction

In recent years, multiview video system have become more and more popular due to the adoption of interactive multimedia applications such as 3D television, surveillance and wireless sensors network. In addition, the wide spread of smart phones equipped with high definition cameras and the availability of powerful uplink 3G connections like HSPA (High Speed Packet Access), HSPA+, and LTE (Long Term Evolution) is one of the key enabling factors of co-creation of multimedia contents for several value added applications, like interactive news and distributed environmental surveys.

In surveillance applications, the employment of multiple views can be used to improve the performance of event detection and recognition algorithms.

However, the amount of data generated by multiview systems increases with the number of cameras. For this reason, data compression is a key-factor in such systems. A distributed video coding approach in multiview applications has the following advantages:

- The communication between the different cameras can be removed. In conventional multiview video coding, inter-view correlation is exploited at the encoder. In practical scenario it is very difficult to exchange such amount of data among cameras. In multiview distributed video coding, no communication is needed among cameras.

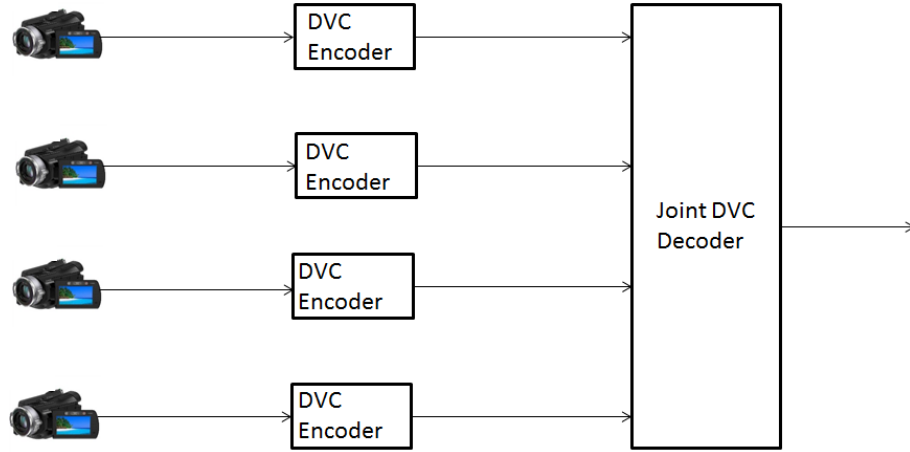


Figure 4.1: General scheme of multi-view distributed video acquisition

This can be very interesting when dealing with dense multicamera systems.

- Low computing complexity makes the multiview video data be transmitted with low delay. Although the complexity of the decoder increases by temporal and inter-view computation, fast algorithms can be used in on-line decoding case. However, a general DVC scenario is offline decoding case where the complexity of decoder is not a major concerns.
- The selection of the views that need to be decoded is more flexible. In conventional multi-view video coding scheme, the reference frames are predefined during encoding. All the reference frames have to be decoded before the current frame despite which view they are from. Instead, in our case, this redundancy can be avoided because the inter-view prediction is done at the decoder and the decoding of different views is freely chosen.

Multi-view Distributed Video Coding (MDVC) differs from mono-view and from stereo DVC in the decoder because the SI is obtained not only using the frames within the same camera but considering information coming from frames of the other cameras, [12]. This chapter is addressed to the design of multi-view distributed video coding schemes. First a summary of the related works is depicted in Section 4.2 while a review of different side information techniques is presented in 4.3. In Section 4.4, a new fusion technique between temporal and spatial side information in Zernike Moments (ZM) space is proposed

where only one camera is DVC coded while the other are conventionally coded. Then the proposed method has been generalized in order to code all the cameras. The aim of these contributions is the proposal of a coding scheme that, thanks to the statistical correlation among the different views, allows reconstructing a 3-D highly defined scene based on the DVC theoretical framework. The creation of side information is described in Section 4.5, the performance assessments are reported in Section 4.6 and finally in Section 4.9 conclusions are drawn.

4.2 Related Works

Artigas et al. [58] proposed two novel fusion techniques between temporal and inter-camera side information. In the first technique, temporal motion interpolation is performed between the previous and the forward frames from the side cameras. The result is subtracted from the current frame and then thresholded to obtain a binary mask. The second algorithm uses the previous and the forward frames as predictors for the current frame on the side cameras to compute a reliability mask. The obtained results show that the fusions improve the average PSNR of the side information using high resolution video. However, the rate-distortion (RD) performance of DVC is not investigated and the simulations are run using the original frames, which is in practice not feasible. Moreover, depth maps are required to perform the inter-camera prediction, which is a hard problem for complex real world scenes.

In [57], a scenario of low-cost camera arrays employing a low-complexity encoders and high-complexity decoder is considered. Captured multi-view video frames are first encoded by WZ or Intra-codec and then transmitted to decoder. To achieve a good efficiency coding performance, a wavelet-based WZ video coding scheme is proposed as the core coding module. WZ frames are DWT transformed and SPIHT method is used to reorder the transformed coefficients before turbo-coding. Once WZ frames are created, a Log-MAP algorithm is used to successively decode them with the side information until an acceptable BER is achieved. The side information is a merge between temporal information and inter-view information and it is calculated with flexible prediction scheme similar to H.264

mode decision scheme. Evaluations of performances show that the proposed method significantly outperforms conventional H.263+Intra Coding. In [59], the wavelet transform is combined with turbo codes to encode a multi-view camera array in a distributed way. At the decoder, a fusion technique is introduced to combine temporal and homography-based side information.

In [60] new fusion techniques are designed, based on idea that the global coding performance are strongly dependant on quality of side information. The better the side information is, the fewer bits are required to encode Wyner-Ziv frames. The fusion methods use temporal and inter-view residuals and a linear combination between the available values. However, although all these approaches are extremely promising, they are still not as efficient as standard video coders in terms of rate-distortion performance due to the fact that distributed source coding techniques rely on a a-priori knowledge of the correlation structure [61]. These approaches are often not simple in practical applications as asymmetric: in fact some cameras need to transmit their full information to provide side information to the decoder while others only transmit partial information. Finally most of the multi-view DVC approaches do not take advantage of the multi-view geometry to improve the performance of their encoders.

4.3 Side Information Techniques

4.3.1 Multiview Motion Estimation (MVME)

The motion vectors are first computed on Intra camera and then used on Wyner-Ziv cameras to estimate WZ frames, [62]. The relationship between two cameras is evaluated by finding disparity vectors. As for motion vectors, they relate each block in the WZ frame with the one more similar in the Intra camera. Then, each matched block in the Intra camera is again searched for in a temporally adjacent frame and so the motion vectors are found. The obtained motion vectors are applied to WZ camera to generate the estimation. The used frames are called *path*, and with this technique there are 4 possible paths.

It is possible to increase the number of paths by finding the disparity vectors in the

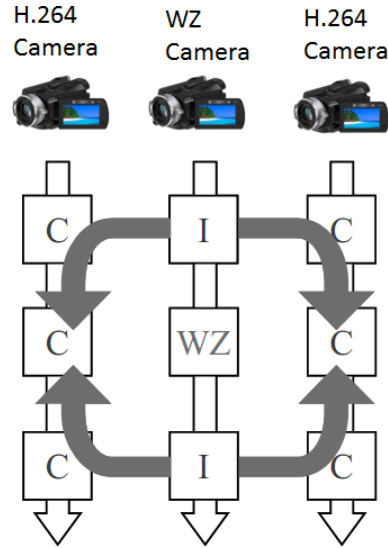


Figure 4.2: 4 different paths obtained with two H.264 cameras and two reference frames in each H.264 cameras.

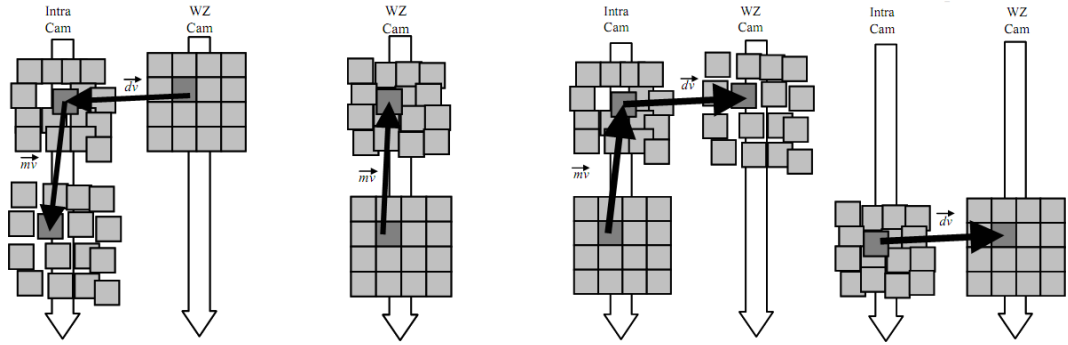


Figure 4.3: Motion estimation and disparity estimation.

previous instant and applying them to the current time instant to estimate the WZ frame. Increasing the amount of motion and disparity data makes the results more reliable but at the same time it increases the amount of data that the decoder must carry out. The reliability measure used in this work to weight different paths is based on the local variance of the motion field around each block. In fact, real fields are usually uniform, except at the edge while incorrectly calculated fields are usually very noisy.

4.3.2 Side Information with encoder driven fusion

In [1], the authors proposed a fusion technique for merging temporal, homography and Disparity Compensation View Prediction (DCVP) side information. At the encoder, a binary mask based on the knowledge of the original video is computed. Then, it is compressed using JBIG [63] and transmitted to the decoder.

The temporal side information is computed with block-based motion estimation algorithm. Then, the motion vectors are interpolated at mid-point, considering the intersection point of each motion vector with a virtual frame at mid-distance from both key-frames.

The inter-view side information is performed calculating the homography from left, right or both cameras. For the spatial information, also disparity compensation view prediction is considered. DCVP uses the left and the right frames from the side cameras. To perform the optimal motion vectors, each vector is weighted with weights 0.1, 0.2 until 0.9. For each weight, PSNR with respect to the central camera is computed. The weight with maximum PSNR is kept and used for the rest of the video. To fuse all the side information (temporal, homography, DCVP), at the encoder a binary mask is created comparing the Wyner-Ziv frame with the previous and the forward frame. The binary mask is, then, encoded using JBIG. At the decoder, the binary mask is used to define for each pixel which reference to use.

The rate-distortion performances overcome monoview DVC by a maximum gap of around 1dB. PSNR for the proposed method is compared only with temporal and spatial side information.

4.3.3 Side Information with Motion Compensated Temporal Interpolation and Homography Compensated Inter-view Interpolation

Dufuax et al. [64] have proposed a multiview distributed video coding with three cameras. Only the central camera is DVC coded while the side camera are coded with conventional video codec as H.264/AVC.

Temporal side information is generated computing block-based motion vectors by block-matching algorithm. The WZ frame is at mid-distance between the previous and the forward frame, so the computed motion vectors are halved.

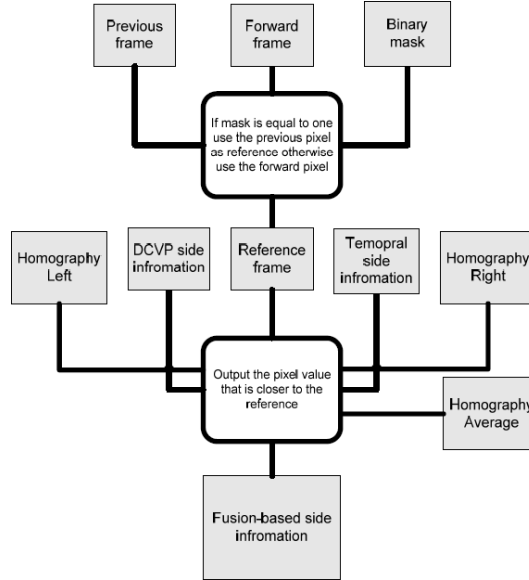


Figure 4.4: fusion scheme at the decoder for [1]

The disparity between central and the side view is modeled by homography. It can be applied in three different ways: by taking the transformed pixel in the left view, by taking the transformed pixel in the right view and by taking the average of the two.

Hence, different modes to create side information are possible and it is possible to switch modes on a pixel by pixel basis. The proposed fusion technique uses the original frame to determine the optimal prediction.

4.3.4 View Synthesis

In [65], the authors proposed a view synthesis prediction technique for multiview video compression. A virtual version of each view is synthesized using previously encoded views and using the virtual view as a reference for predictive coding. The knowledge of camera and scene geometry can improve prediction and compression of a given camera from its neighbors.

Here, a disparity compensation view prediction is used and compared with the view synthesis prediction. With DCVP the value of the intensity of the frame of a given frame can be predicted from the previous camera. It provides improvements over temporal prediction but it doesn't take advantage of important multiview video features. In fact,

while temporal prediction only exploits translation, disparity can be related also rotation, zoom or other intrinsic camera properties that are often difficult to model using only the translational motion compensation. To take advantage of these features, a virtual view from previously encoded views is synthesized and then using this virtual view, a predictive coding is performed.

To synthesize $\hat{I}[c, t, x, y]$, a depth map $D[c, t, x, y]$ is required that describes how far the object corresponding to pixel (x, y) is from camera c at time t , as well as an intrinsic matrix $A(c)$, rotation matrix $R(c)$ and a translation vector $T(c)$ describing the location of the camera relative to some global coordinate system. Using the pinhole camera model to project the pixel location (x, y) into world coordinates $[u, v, w]$, we have:

$$[u, v, w] = R(c) \cdot A^{-1}(c) \cdot [x, y, 1] \cdot D[t, c, x, y] + T(c) \quad (4.1)$$

The drawback of this technique is the difficulty to estimate depth for real world complex scene. In addition, the quality of the side information depends on the precision of camera calibration and depth estimation.

4.4 The proposed method with only one WZ camera

The proposed MDVC scheme, reported in [66], is now described and shown in fig.4.5. The main goal of our work is to generate at the decoder the side information that optimally blends temporal and interview data. Multi-view DVC performance strongly depends on the side information quality built at the decoder. At this aim to improve its quality a temporal view compensation/prediction in Zernike moments' domain is applied. More in detail, we first apply state of the art key point extraction and matching algorithms to estimate the parameters characterizing the effects of the geometrical transformations among different views in the image planes. Then, to handle rotations, we partition each view in blocks and for each of them we compute the Zernike moments as a projection of the function defining the Region Of Interest onto a set of orthonormal functions within circles whose radii are selected according to the previously estimated zoom factors. ZM are generally used in several computer vision applications due to the low sensitivity to image noise and to good feature representation capabilities. The disparity estimation can

be obtained by comparing ZMs rather than by comparing intensity values. We estimate disparity in the moment space by minimizing the difference between reconstructed intensity values. Spatial and temporal motion activity will be fused together at the encoder side to obtain the overall side-information. The adopted fusion scheme compares the temporal and the inter-view residuals (the difference between the two compensated reference frames) and uses the estimation to have the smallest one. The proposed method will be evaluated by rate-distortion performances for different bit-rates.

The overall architecture is composed of three cameras which are assumed to be static. The two side views are coded with a conventional AVC Intra encoder (H.264/AVC) and the central view is coded using Wyner-Ziv coding, in particular the odd frames are considered key-frames (KF) and coded H.264/AVC and the even frames are Wyner-Ziv coded. The Wyner-Ziv frames are transformed (using DCT), quantized and the resulting bit planes are turbo-encoded. However systematic bits are not transmitted and are discarded, instead they are replaced at the decoder side by the side information, created at the decoder side with key-frames. The side information is the merge of temporal motion estimation between key-frames belonging to the same camera and disparity estimation between side views camera frames and central camera frames. Then, SI is turbo-decoded with the necessary parity bits to obtain the decoded WZ frame.

4.5 Multi-view Side Information Creation

Multi-view video coding differs from mono-view DVC in the decoder. More precisely the side information (SI) is constructed not only using the frames within the same camera but using frames from the other cameras as well. Multi-view side information exploits intra-view information between the previous and the next decoded key-frames and the inter-view information between the central Wyner-Ziv frames and side views frames. In this section, a new multi-view side information creation method based on Zernike moments temporal view prediction is presented.

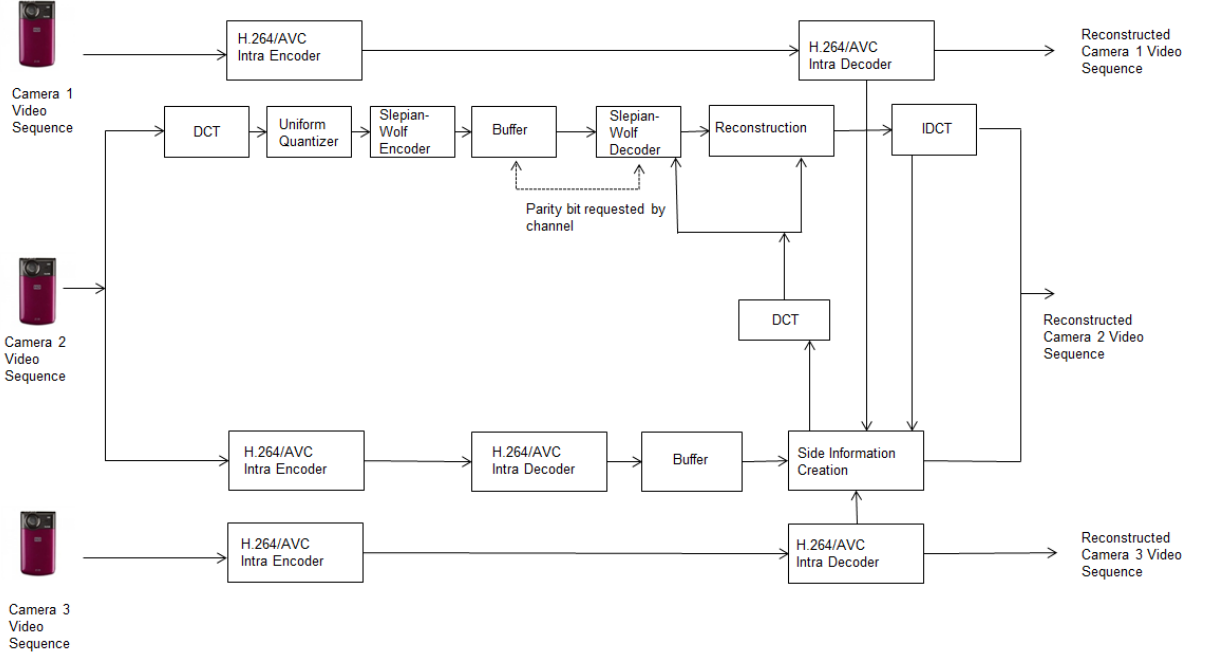


Figure 4.5: Multi-view distributed video coder architecture

4.5.1 Temporal information in Zernike domain

To improve side information quality a temporal view compensation/prediction in Zernike moments' domain is applied. More in detail, we partition each view in blocks and for each of them we compute the Zernike moments for each blocks. The motion estimation can be obtained by comparing ZMs rather than by comparing intensity values. We estimate temporal motion in the moment space by minimizing the difference between reconstructed intensity values.

ZM are generally used in several computer vision applications due to the low sensitivity to image noise and to good feature representation capabilities. Let $\mathbf{x} = [x_1, x_2]$ denote the cartesian coordinates of points in the real plane \mathbb{R}^2 . The polynomials form a complete orthogonal basis set defined on the unit circle $x_1^2 + x_2^2 \leq 1$ and belong to the class of complex, polar separable functions with harmonic angular shape, called circular harmonic function (CHF), as defined in [67]. Specifically, denoting with $\tilde{V}_{nm}(\rho, \theta) = V_{nm}(\rho \cos \theta, \rho \sin \theta)$ the expression of Zernike polynomials $V_{nm}(x_1, x_2)$ of order n and repetition index m in polar

coordinates $\rho = \sqrt{x_1^2 + x_2^2}$ and $\theta = \tan^{-1}(x_2/x_1)$, we have

$$\tilde{V}_{nm}(\rho, \theta) = R_{nm}(\rho)e^{jm\theta}, \quad (4.2)$$

where $R_{nm}(\rho)$ is Zernike radial profile defined as:

$$R_{nm}(\rho) = \begin{cases} \sum_{s=0}^{(n-|m|)/2} \frac{(-1)^s (n-s)! \rho^{n-2s}}{s! (\frac{n+|m|}{2}-s)! (\frac{n-|m|}{2}-s)!} & \text{if } n - |m| \text{ even} \\ 0 & \text{if } n - |m| \text{ odd,} \end{cases} \quad (4.3)$$

The orthogonality condition on Zernike polynomials gives:

$$\int_0^{2\pi} \int_0^1 V_{nl}^*(\rho, \theta) V_{mk}(\rho, \theta) \rho d\rho d\theta = \frac{\pi}{n+1} \delta_{nm} \delta_{lk}, \quad (4.4)$$

where δ_{nm} denotes Kronecker delta.

For a continuous function $f(\mathbf{x})$, inside the unit disk centered at \mathbf{x}_0 , the following Zernike polynomial expansion holds:

$$f(\mathbf{x}) = \sum_{n=0}^{\infty} \sum_{m=-\infty}^{+\infty} A_{nm}(\mathbf{x}_0) V_{nm}(\mathbf{x} - \mathbf{x}_0), \quad (4.5)$$

with expansion coefficients $A_{nm}(\mathbf{x}_0)$ given by:

$$A_{nm}(\mathbf{x}_0) = \frac{n+1}{\pi} \int \int_{\|\mathbf{x}-\mathbf{x}_0\| \leq 1} f(\mathbf{x}) V_{nm}^*(\mathbf{x} - \mathbf{x}_0) dx_1 dx_2. \quad (4.6)$$

Since $\tilde{V}_{nm}(\rho, \theta)$ can be rotated by an angle φ by multiplying it by a factor $e^{-jm\varphi}$, the expansion coefficients $A_{nm}^{(\varphi)}(\mathbf{x}_0)$ of an image $f(\mathbf{x})$ rotated by an angle φ are related to the expansion coefficients $A_{nm}(\mathbf{x}_0)$ of $f(\mathbf{x})$ by the following relationship:

$$A_{nm}^{(\varphi)}(\mathbf{x}_0) = A_{nm}(\mathbf{x}_0) e^{-jm\varphi}. \quad (4.7)$$

This leads to the well-known rotational invariance property $|A_{nm}^{(\varphi)}| = |A_{nm}|$.

To calculate Zernike moments, the image (or the region of interest) is first projected onto the unit disk. Pixels outside the unit circle are not considered.

In practical situations, the reconstruction of the image is performed by using a finite number of Zernike moments i.e.:

$$\hat{f}(\rho, \theta) = \sum_{n=0}^N \sum_{m=-M}^{m=+M} A_{nm} V_{nm}(\rho, \theta). \quad (4.8)$$

The reconstruction error depends on both the number of employed moments and on the size of the image.

The temporal motion estimation can be obtained by comparing ZMs rather than by comparing intensity values. We estimate disparity in the moment space by minimizing the difference between reconstructed intensity values. Let us consider a $N \times N$ block B then the temporal motion estimation between the reconstructed k -frame \hat{f}_k and $(k+1)$ -reconstructed frame $\hat{f}_{(k+1)}$ is:

$$C(x, y, \Delta x, \Delta y) = \sum_{(x, y) \in B} (\hat{f}_{k+1}(x + \Delta x, y + \Delta y) - \hat{f}_k(x, y))^2 \quad (4.9)$$

where $-d_m^{\max} < \Delta m < d_m^{\max}$, $-d_n^{\max} < \Delta n < d_n^{\max}$ are the maximum allowed displacements. The temporal motion estimation is given by:

$$(\Delta x, \Delta y) \in \arg \min_{\Delta x, \Delta y} (C(x, y, \Delta x, \Delta y))$$

Then, the motion vectors are interpolated at mid point to generate side information. This is done by considering the intersection point of each motion vector with a virtual frame at mid-distance from both key frames as shown in fig.4.6.

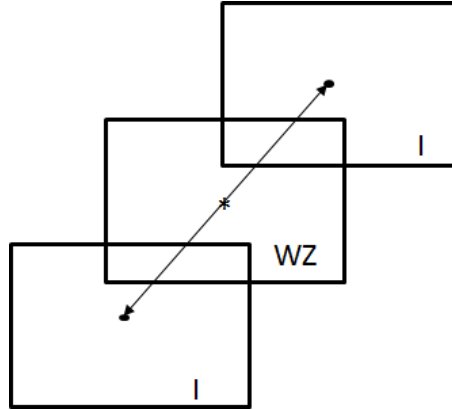


Figure 4.6: Multi-view distributed video coder architecture

4.5.2 Spatial Information

The spatial side information is constructed by homography between central cameras view and side cameras views [64]. More in detail, we first apply state of the art key point extraction and matching algorithms to estimate the parameters characterizing the effects of

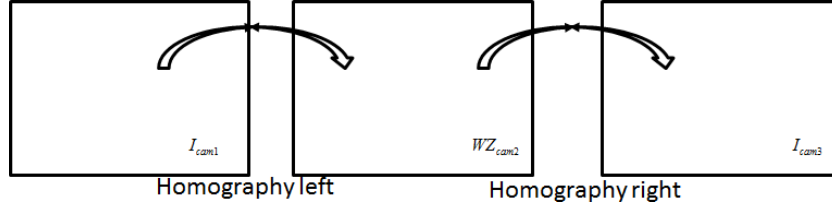


Figure 4.7: spatial side information based on homography

the geometrical transformations among different views in the image planes. The homography is a 3×3 matrix that relates one view to another one in the homogeneous coordinates system. Under homography, we can write the transformation of points 3-D from the first view to the second view as:

$$X_2 = HX_1 \quad (4.10)$$

where $X_1, X_2 \in \mathbb{R}^3$. In the image planes using homogeneous coordinates, we have: $\lambda_1 x_1 = X_1$, $\lambda_2 x_2 = X_2$, therefore $\lambda_2 x_2 = H\lambda_1 x_1$. This means that x_2 is equal to Hx_1 up to a scale. In homogeneous coordinates, we get the following constraint:

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix}$$

This model is suitable when the scene can be approximated by planar surface or when the scene is static and the camera motion is a pure rotation around its optical center. In our case the first assumption applies.

To compute homography matrix, it is necessary to know almost three matching points between the two frames. At this aim, salient points based on invariants have been extracted and then ranked based on the saliency of the key points. We define salient points of a pattern the ones characterized by wide spatial and angular bandwidths (i.e., the corners). Thus, we select as salient points those points corresponding to the local maxima of the spatial density of the gradient energy. In practice, we adopt Harris corner detector for choosing key-points.

4.5.3 Fusion scheme

The fusion step merges the different side information (temporal, homography right and homography left) in order to improve the quality of the final one. The fusion scheme adopted in this work, follows the method proposed in [1]. The idea is to determine a very good estimate of the Wyner-Ziv frame, which is called the fusion mask. The decision for each frame is taken with respect to this fusion mask. At the encoder, each pixel of the Wyner-Ziv frame belonging to the central view (cam2) at the time k is compared to pixels coming from the previous key-frame KF_{k-1}^{cam2} and forward frame KF_{k+1}^{cam2} . If the one from the previous pixel has a closer value, the binary mask at the pixel position is set to one. On the other hand, if the forward pixel has a closer value, it is set to zero. The binary mask then is turbo-encoded. At the decoder, the binary mask is compared with temporal side information, between the previous frame KF_{k-1}^{cam2} of the central camera (cam2) and the forward frame KF_{k+1}^{cam2} of the central view, and spatial side information between the first camera frame I_k^{cam3} (Intra-coded) at the time k and the central camera frame WZ_k^{cam2} at the same time and the spatial side information between the central camera frame WZ_k^{cam2} and the third camera frame I_k^{cam3} (Intra-coded). The fusion mask defines for each pixel which reference to use.

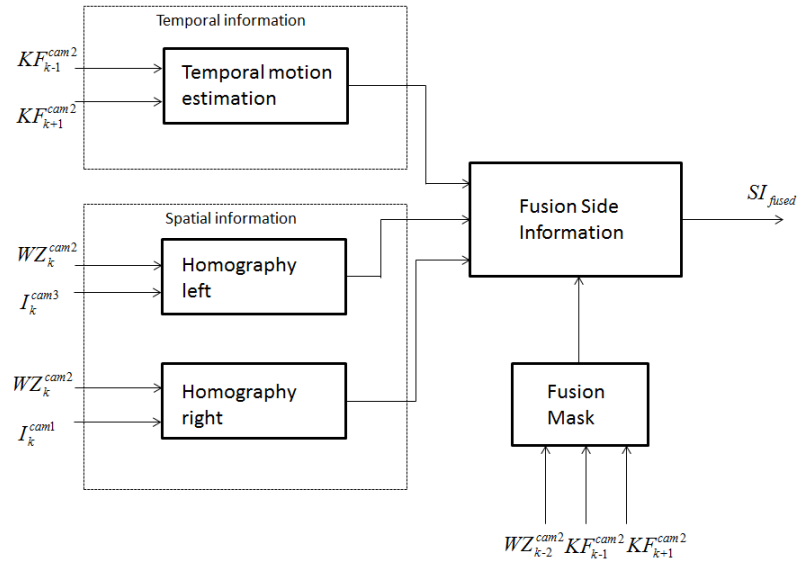


Figure 4.8: Fusion scheme of side information

4.6 Experimental Results

Now we show the performance results for "*Breakdancer*" multi-view sequence available at [68]. The spatial resolution has been halved to 1024x768 to 512x384 pixels, 100 frames were used for the sequence, the capture rate is 15 frames per second, and only the first three cameras were used.

Breakdancer contains significant motion so the motion estimation is a difficult and challenging task. In the DVC architecture, a GOP equal to 2 has been analyzed, where the central view has been Wyner-Ziv coded while the side views are coded with a conventional H.264/AVC. As usual for WZ coding, only luminance data has been coded; the total bit-rate includes the luminance rate for the WZ frames and key frames for the central view to be coded since the side views are always the same.

The side information for WZ frames is created merging motion estimation in Zernike domain and disparity estimation as explained before. For Zernike moments only the first five orders are considered to have a good image reconstruction. The bit-rate of the multi-view coded sequence have been systematically varied and 9 reference bit-rates were considered: 80, 200, 300, 800, 1500, 2000, 5000, 10000 and 20000 Kbit/sec, i.e. ranging from a low bit-rate transmission case where the DVC approach results to be more suitable to a high bit-rate channel case, where a conventional H.264/AVC coder usually results to be more appropriate. For each reference bit rate, the H.264/AVC 3-D coded sequences and the DVC 3-D coded sequence have been considered. Figures 4.9, and Figure 4.10 show the rate distortion (RD) performance for the analyzed WZ stereo coding architecture, by respectively considering the PSNR and SSIM quality models at different bit-rates. These metrics have been adopted to evaluate the quality of the decoded 2-D right sequence of the central camera.

Given that DVC schemes are more suitable for low bit-rate channels because less amount of data need to be transmitted. On the contrary, the reversal of the trend at about 3000 Kbit/sec shows that a conventional H.264/AVC coder results to be more appropriate at high bit-rate even if DVC approach would be still preferred in some cases due

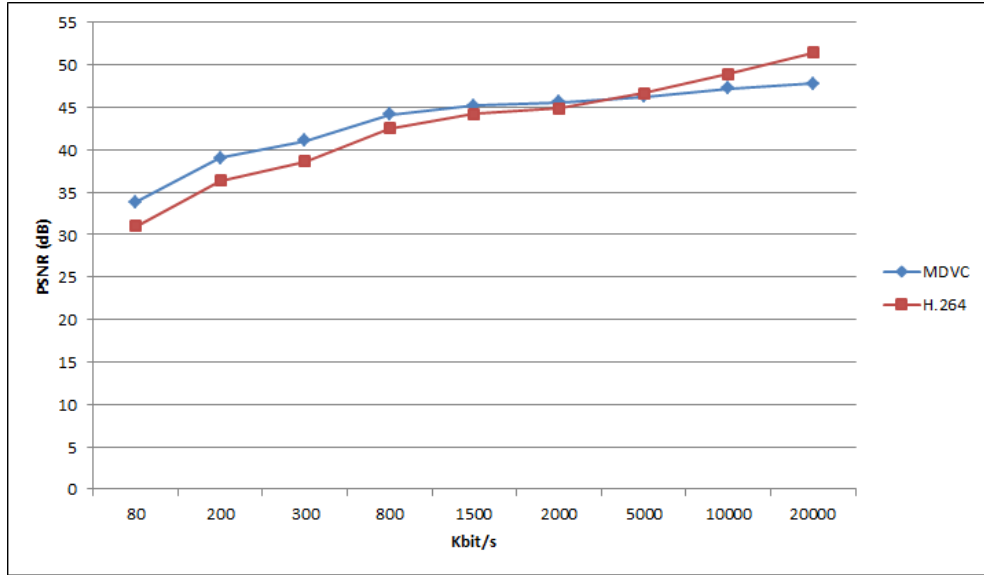


Figure 4.9: RD performance by PSNR evaluation. PSNR is averaged on the whole central camera sequence.

to the advantage of low-complexity encoders.

Figures 4.9, and 4.10 report the same trend for the WZ coded right sequence of the second camera.

The proposed method has been compared with other methods present in literature as [1] and [64], that have been described in paragraphs 4.3.2 and 4.3.3. The RD performance shows that our approach based on Zernike moments outperforms the state of art of side information method. As said before, for low bit-rate MDVC performs better than conventional H.264/AVC.

4.7 Proposed Approach with all WZ Cameras

In Section 4.4 a multiview distributed video coding where only the central camera is WZ coded while the side views are conventionally encoded has been presented. Now we propose a general scheme where all the camera are WZ coded. We adopt the so called symmetric scheme 1/2, which gives identical roles to all cameras: each of them produces alternatively

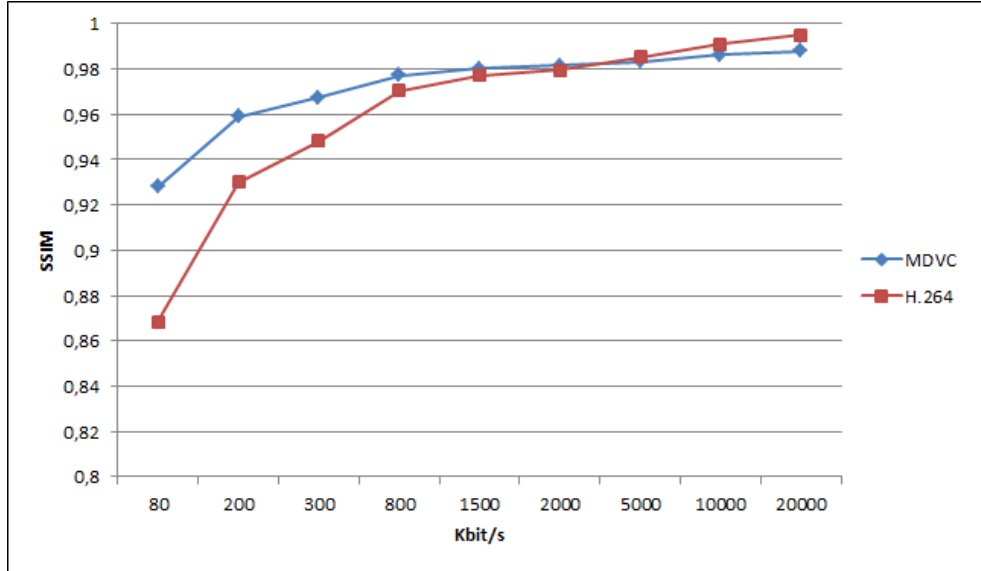


Figure 4.10: RD performance by SSIM evaluation. SSIM is averaged on the whole central camera sequence.

one Key-Frames followed by one Wyner-Ziv frame. A shift is introduced between cameras, in order to obtain a frame repartition in time-view domain. In Figure 4.12, the frames repartition with three cameras. A WZ frame can exploit two interpolation, a temporal estimation based on the previous KF and the forward KF and an inter-view estimation based on KF coming from the left and the right views. The two estimations must be combined in order to build a unique SI for the turbo decoder, while improving the rate distortion performance.

For the central camera, named camera 2, the approach followed in 4.5 will be applied. The central camera can exploit information coming from the two side views for the inter-view estimation. The temporal side information will be calculated in Zernike domain, while the homography between central view and left view and central view and right view is performed in order to build spatial side information. To merge all the side information, a binary mask is created to act as a reference. Then, temporal side information and the two spatial side information are compared with the reference, as explained before.

For the side cameras, camera 1 and camera 3, a slight different approach is followed. The temporal side information is usually created with Zernike moments, while the spatial side

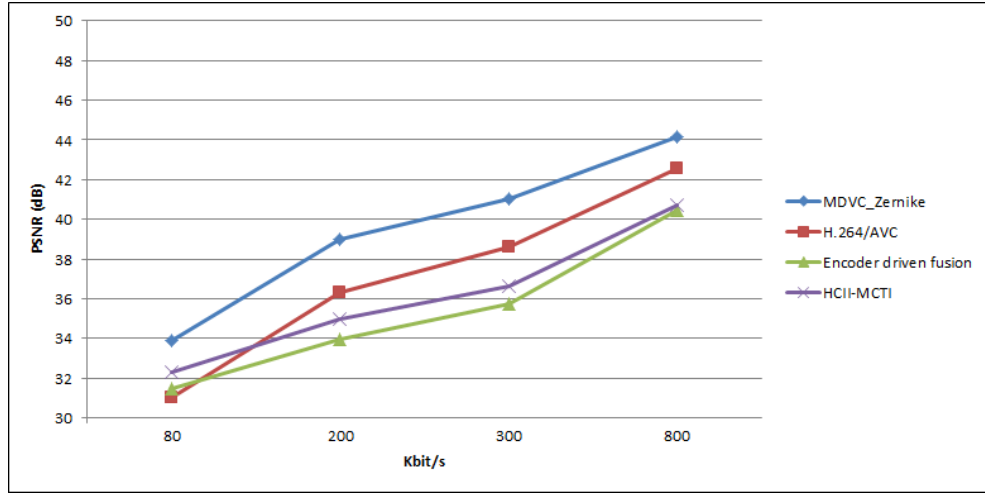


Figure 4.11: RD comparison between the proposed method and the state of art.

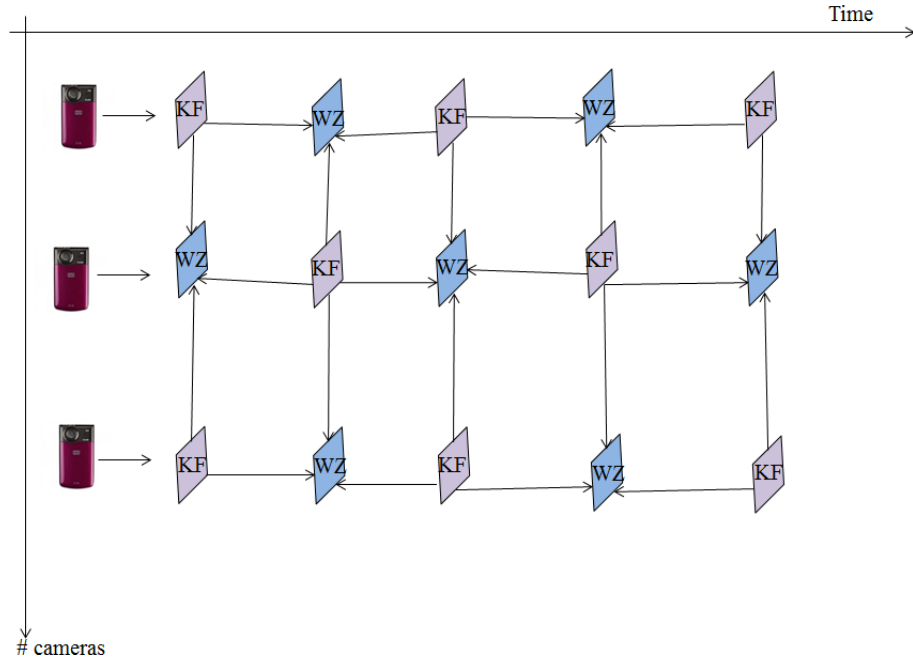


Figure 4.12: Multiview scheme with frame repartition. WZ frame and KF frame are alternated for each camera.

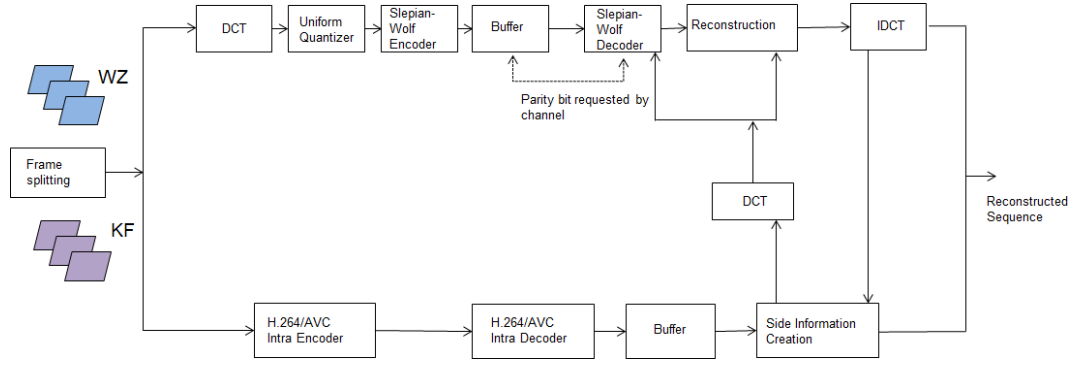


Figure 4.13: Distributed multiview video coding.

information is computed finding disparity with block matching algorithm between KF belonging to the considered side camera and KF belonging to centra camera.

4.8 Experimental Results

The sequence Breakdancers, [68], is used for evaluating the performance of the proposed scheme and for comparing with [1]. Breakdancers contains significant motion. This makes the motion estimation a difficult and challenging task. The spatial resolution is 512x324 for all the sequence and the temporal resolution is 15 fps. In this research, three camera views are used and the performance is evaluated only for all the cameras. For DVC simulations, the following settings have been employed:

- Only luminance data is coded.
- All the cameras all contain WZ frames. All the sequences are split, the odd frame are WZ coded while the even frame are conventionally encoded in Intra mode. A shift is introduced between cameras to have a frame repartition in time-view domain.
- The same Quantization Parameter (QP) is used the key frames of the cameras. A QP is defined per quantization matrix such that the decoded key and WZ frames have a similar quality.
- The GOP size is equal to 2.

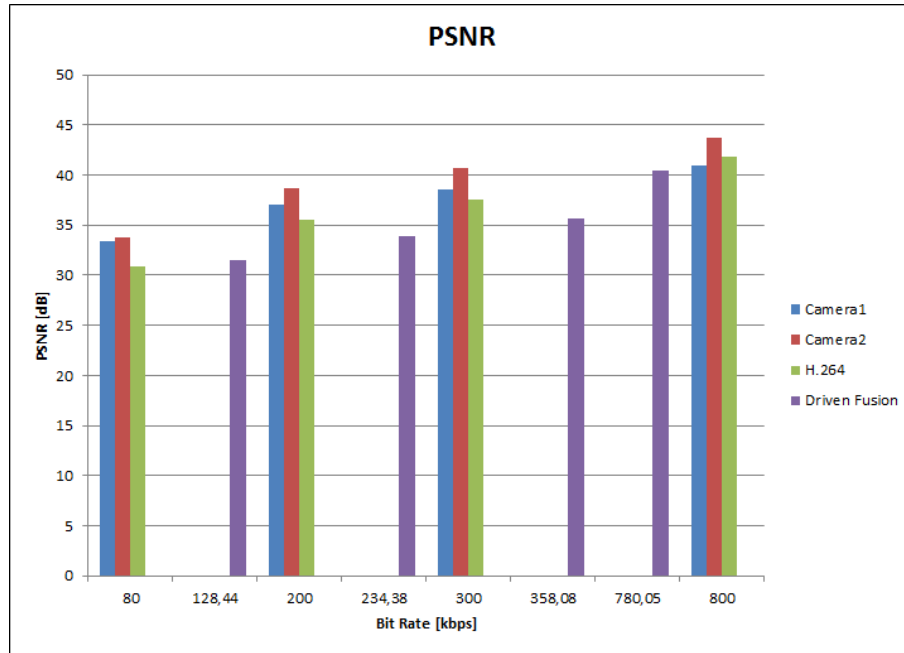


Figure 4.14: PSNR evaluations of the proposed method respect to the state of art.

The rate-distortions are evaluated by PSNR and SSIM. The performance evaluation shows that multi-view DVC shows better trend when compared with conventional Intra codec H.264/AVC at low bit-rates. PSNR of camera 2 is higher than PSNR of camera 1, the reason of these trends is that the central camera (Camera2) takes advantage of three contribution, one coming from temporal information and two from inter-view information. While the side views exploit only one temporal side information and one spatial side information (coming from the central camera). Also the comparison with the state of art is carried out and the simulation results show that the proposed method based on Zernike moments outperforms the state of art.

4.9 Conclusions

In the last years multiview video systems are becoming more and more appealing due also to the wide spread of smart phones equipped with high definition cameras and the availability of powerful uplink 3G connections, also 3DTV application can take advantage of this approach. Respect to mono-view DVC and stereoscopic DVC, multi-view DVC has to deal with more data coming from different cameras. Hence, the side information quality

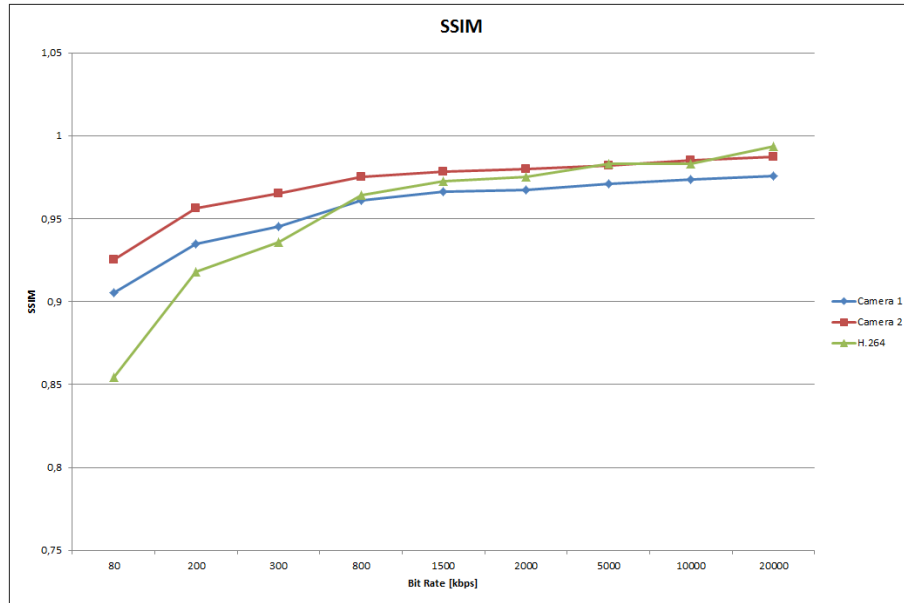


Figure 4.15: RD by SSIM evaluation. The SSIM has been averaged on the all video sequence.

is an fundamental deal and the fusion of temporal and side information is very important. In this chapter, some results related to a novel multiview DVC technique based on Zernike moments have been proposed and compared with the state of art.

Zernike moments allows a good frame reconstruction and motion estimation exploiting only the first five orders of Zernike polynomials. The introduced system allows a 3-D highly defined scene based on DVC framework. It is interesting for mobile applications where it is required a low complexity encoders. Moreover the considered coding approach has been compared with conventional H.264/AVC. Performance assessments showed that DVC has a better quality than H.264/AVC for lower bit-rate; at the contrary, for higher bit-rates conventional stereo video coders result more powerful. Multi-view DVC can be applied in various scenarios e.g. smart phones video conferencing where mobile devices have limited computational resources and power. Further, Multi-view DVC shows a better error resilience and should achieve a good compression efficiency when compared to conventional codec.

Chapter 5

Fountain Code based AL-FEC for Multicast Services in MANETs

5.1 Introduction

The emerging spread of portable devices, e.g. smart phones that can benefit from wireless connection, and the growing wireless connection availability open new scenarios where users can benefit from Internet services wherever and whenever they want.

MANETs (Mobile Ad-hoc NETworks) are autonomous and mobile systems consisting of router and hosts linked by arbitrary wireless distributions, [69], and have been created also to answer to previous requirements. Locations of routers and hosts may change continuously and in unpredictable way forming random and non-optimized nets. Since MANET is a network without infrastructure that can be easily installed and reconfigured, it is very appealing for low-budget commercial services. On one side, the development of ad-hoc networks can be progressively carried out with respect to changing requirements thanks to their scalability property; on the other side it is necessary to deal with reduced performances due to multi-hop routing and distributed control. Moreover in wireless systems, the poor channel quality and link instability, pose challenges to traditional routing schemes.

In this context, fountain codes are an appealing, capacity-approaching Application Layer - Forward Error Correction (AL-FEC) solution for data transmission over lossy packet

networks. The property of being rateless i.e., the ability of adapt the code on the fly, makes fountain codes an attractive solution for data broadcast/multicast applications in a MANET where users may experience varying channel conditions and packets loss rate. A fountain code can be seen as a code that generates a continuous flow of transmitted data packets, simulating the action of water falling from a spring into a collecting bucket, [13]. Once the bucket is full, collection process ends and further processing on decoding the content of the bucket will take place. It does not matter which droplets are falling into the bucket as long as the bucket is full. Luby Transform (LT) codes are the first realization of digital fountain codes, [70], and have been applied in my work due to their near optimality with respect to any erasure channel.

In this chapter, we analyze the use of LT codes, in case of small and long message length, for robust multicast data transmission over MANET, in order to reduce the end to end latencies and packet delays, even in presence of significant packet losses. The idea is to propose a joint source-channel coding method that exploits LT code. In chapter 3, the basic theme of distributed joint source-channel coding has been studied and a method that relies on turbo code has been proposed. Hence, with respect to the previous case, here we present and analyze a joint source-channel coding technique based on LT code that provides reliable and real-time multimedia content services.

To provide advantages in terms of throughput or robustness in multicast delivery over ad-hoc wireless network, network coding has recently emerged as a new appealing field, [71]. In network coding the nodes combine the received packets before the retransmission to neighboring nodes. Respect to this approach, we can, at the same time, improve robustness and reduce end to end latencies, proposing a MANET multimedia streaming that exploits fountain codes, [72].

More specifically, we consider MANET scenarios where nodes are randomly moving and a single source is transmitting multimedia data to N receiving nodes. Data packets are sent from source to destination through intermediate nodes that flood data messages into network exploiting PUMA protocol.

PUMA is a multicast routing protocol that relies on creation of multipaths between router and the core node of the mesh network, [73]. This algorithm shows high robustness to

link failure and losses. Every transmitting node sends data packets through shortest paths existing between router and core node. When a data packet has to reach a mesh member, the data is flooded in to the mesh and every node keeps update a cache with ID packets in order to drop duplicates. When a packet flow, protected by an LT code, is sent to an intermediate destination node, that node has to collect the minimum amount of packets, eventually sent by different one-hop neighbors, that allows LT decoding, and to reconstruct the original information flow. Then, it encodes the reconstructed flow by means of an LT code adapted to the status of the worst link with its one-hop neighbors. Then it scrambles the encoded packets, based on a $p - r$ Fibonacci sequence, [74], re-transmits them. For the scrambling phase, the $p - r$ Fibonacci sequences have been chosen, because with an appropriate selection of p and r parameters values, that sequences allow to scramble the order of every amount of packets.

The rest of the chapter is organized as follow: in Section 5.2, the MANET scenario is depicted, in Section 5.3 PUMA protocol is described. Then in Section 5.4 the description of the LT code design is given and the different LT codes implementations, considered in this work, are presented; in section 5.5, the proposed idea is illustrated, then in Section 5.6 the experimental results are reported. At the end, conclusions are drawn in Section 5.7.

5.2 MANET

The static concept of a fixed network, that allows the inter-communication among devices distributed in known positions in the environment, is now substituted by a new concept of dynamic network, i.e. a network where nodes are dynamically distributed in the environment because they are not only part of fixed devices but they are also embedded in mobile and handled devices. The communication capacity of these mobile devices, is called "*Ubiquitous Communication*".

The interconnection among all these devices, mobile and fixed, is possible thanks to dynamic networks capable of supporting wireless technologies.

Internet Engineering Task Force (IETF), that represents the technician community study-

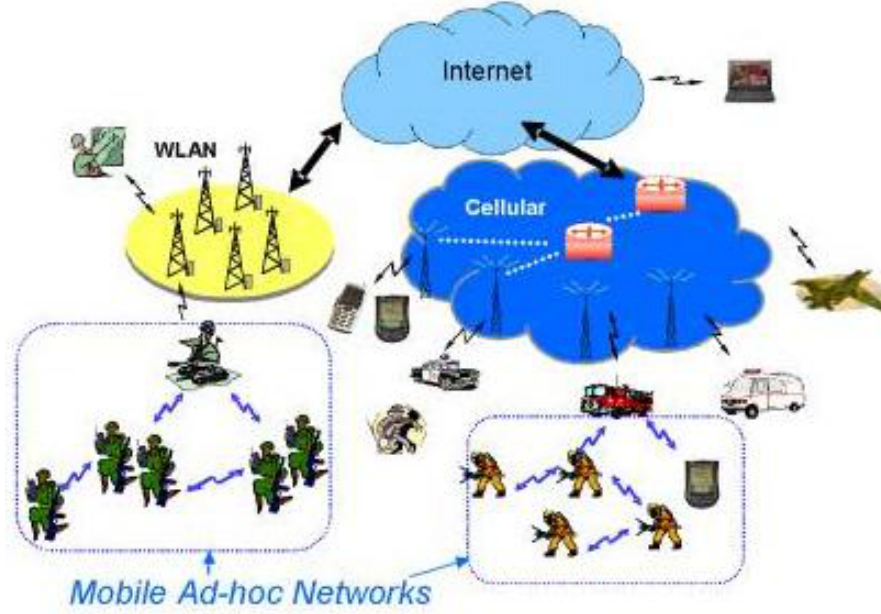


Figure 5.1: Example of MANET applications

ing the evolution of Internet architecture, has defined this new implementation of network as Mobile Ad-hoc NETWORK (MANET), [75]. It is intended to be an autonomous and mobile system composed of router and host that are connected via radio so to create a graph (i.e. an arbitrary distribution of wireless connections). Such a network can not take advantages of classical concepts of data distribution over Internet. In this new approach, routers are continuously changing positions in an unpredictable way, forming random and non-optimal graphs. Ad-hoc network is a set of mobile and auto-configuring nodes without use of pre-existence infrastructure. Without infrastructure, nodes can manage control and networking exploiting distributed algorithms. The underlying idea is that MANET can be created based on specific application and can be handled based on nodes resources. MANET has also the advantage of being robust due to its distributed behavior, nodes' redundancies and the absence of centralized node that can go out of order. All these features makes MANET an appealing solution for military and civil protection applications, [76]. It can be also exploited in commercial applications due to low-cost infrastructure.

The development of ah-hoc network can be progressively increased according to chang-

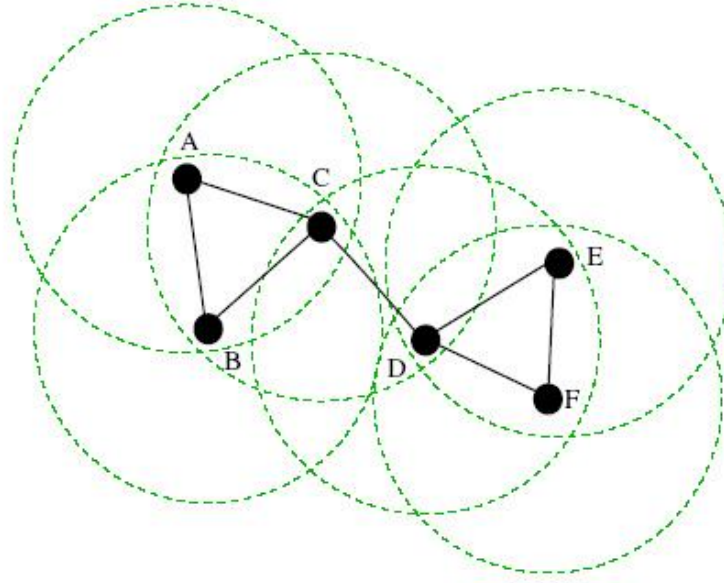


Figure 5.2: Mobile Ad-hoc NETwork

ing requirements and this is possible thanks to its scalability property. At the same time, MANET paradigm has to face limits coming from multi-hop routing and distributed control.

The high level of dynamism in wireless environments, makes the mobility possible unpredictable and the access points are time-varying. If it happens, new change has to be known so that data regarding topological changes can always keep update. In fact, one of the main deal related to ad-hoc network managing lies in the choice of used routing algorithms to adapt the network to the dynamism of the nodes and to notify frequent and random topology changes.

Generally, the amount of signalling data traffic used by a distributed routing algorithm is very high; at this aim, most of the study are focused on optimizing the routing algorithms for Ad-hoc networks and on decreasing signalling data traffic. Routing algorithms for MANET can be classified in three categories: proactive, reactive and hybrid, [77].

5.3 Puma Protocol

The task of a multicast routing protocol for mobile ad hoc networks is to support the dissemination of information from a sender to all the receivers of a multicast group while

trying to efficiently use the available bandwidth even in presence of frequent topology changes. PUMA (Protocol for Unified Multicast Announcement) is a multicast routing protocol used in MANET able to establish and handle a mesh shared network without requiring any unicast routing protocol to operate or pre-assignments of core to groups, [78].

The novelty in PUMA derives from its use of very simple signaling (multicast announcements) to accomplish all the functions needed in the creation and maintenance of a multicast routing structure in a MANET. Multicast announcements are used to: elect cores dynamically, determine the routes for sources outside a multicast group to multicast data packets towards the group, join and leave the mesh of a group, and maintain the mesh of the group. PUMA protocol shows higher PDR (Packet Delivery Ratio) values with lower overhead when compared to conventional multicast protocols as MAODV and ODMRP, [73]. PUMA supports IP multicast service model, allowing at every source to transmit multicast packets to a certain group, without knowing which are the nodes that belong to that group. This protocol is based on receiver-initiated approach, i.e. it uses a special node called core in order to avoid flooding of control and data packets of the sources of different groups. The selected protocol implements a distributed algorithm to select a core node among multicast groups' receivers and to inform of their distance and next-hops to the core. The main key-factor is that multi paths are created between a router and a node, based also on the distance between nodes making this algorithm very robust to link failure and losses in the network. Every router is connected to a core node through the shortest paths. The set of all the paths between a router and core node is called mesh. Every transmitting node will send data packets through shortest paths existing between router and core node. When a data packet has to reach a mesh member, the data will be flooded in to the mesh and every node will keep update a cache with ID packets in order to drop duplicates. PUMA exploits a single control message, *multicast announcements*(MA), to realize all these functionalities. Every multicast announcements is composed by a sequence number, group addresses (ID group), core identification (ID core), core distance and a parent node that selects the preferred neighbor to reach the core. Through these messages, nodes elect core node, determine the routes for sources

outside a multicast group to multicast data packets towards the group, join and leave the mesh of a group, and maintain the mesh of the group. When multicast announcements are sent through the net, connectivity list are created by nodes in order to set the mesh and route data into the network. When a node will send data to a group, this packet will be flooded by nodes with the best MA. Also if a link falls down, the next node with the second best MA will be chosen. This means that multi paths are created into the network to reach the core.

5.4 LT code

LT codes were proposed by M. Luby in 2004, [70]. They represent the first practical realization of digital fountain codes. These codes are rateless i.e. the rate does not need to be fixed in advance and the encoding symbols can be generated on the fly.

The two parameters that characterize an LT code ensemble are the length of information sequence K and ρ the output degree distribution that determines the degrees of the output nodes in the decoding graph.

Each encoded packet t_n is produced from K source data $\mathbf{s} = [s_1, s_2, s_3, \dots, s_K]$ in two simple steps:

- For every t_n packet, the number of the source packet related to it, is chosen; this number is the degree d_n of the encoded packet. d_n is randomly chosen from a degree distribution $\rho(d)$; the appropriate choice of ρ depends on the source data size K .
- d_n distinct input packets are chosen and t_n is set equal to the bitwise sum, modulo 2 of those d_n packets.

Stopping condition for the encoding algorithm can be specified, e.g., by setting the number of packets beforehand, or by managing the acknowledgements that each recipient can send when enough packets have been received. The encoding operation defines a graph connecting t_n encoded packets to the source packets.

Luby showed the existence of an output degree distribution that provides an high probability of successful decoding at rates just below the channel capacity on erasure

channels, [79]. In addition, the probability of successful decoding can be made arbitrarily close to 1 in the asymptotic lengths of data source, i.e. when $K \rightarrow \infty$. This distribution is called *ideal soliton* distribution and is given by:

$$\Psi_i = \begin{cases} 1/K, i = 1 \\ 1/i(i-1), 2 \leq i \leq K \end{cases} \quad (5.1)$$

However, in practical scenario the ideal soliton distribution performs poorly, due to an high sensitivity. In fact, at each stage of decoding pruning, the number of singly connected output nodes is expected to be 1 and whenever it becomes zero prior to decoding end, the decoding fails. So, it was necessary to introduce a modification, called "*robust soliton*" distribution:

$$T_i = \begin{cases} R/(iK), 1 \leq i \leq \frac{K}{R} - 1 \\ (R/K) \ln(R/K), i = \frac{K}{R} \\ 0, \frac{K}{R} + 1 \leq i \leq K \end{cases} \quad (5.2)$$

where $R = c\sqrt{K} \ln \frac{K}{\delta}$, and c and δ are suitable chosen parameters. The parameter δ is a bound on the probability that the decoding fails to run to completion after a certain number K' of packets have been received. The parameter c is a constant of order 1. Decoding is done iteratively by using information of which source blocks are added together in a received packets. The decoder task is to recover \mathbf{s} from $\mathbf{t} = \mathbf{G}\mathbf{s}$ where \mathbf{G} is the matrix associated with the graph. Both side of the transmission know this matrix, even when it is pseudo-randomly generated.

In the decoding algorithm, all the messages are either completely uncertain messages or completely certain messages. Uncertain messages assert that a message packets s_k could have any value, with equal probability; certain messages assert that s_k has a particular value with probability one.

The decoding algorithm finds a t_n received packet that is connected to only one source packet s_k ; if there is not such t_n packet, the decoding fails. Otherwise s_k is set equal to t_n and is added to all check nodes t_n that are connected to s_k ; then all the connections related to source packet s_k are removed. After that, the decoding algorithm finds another t_n connected to only one source packet and continues the described process.

The key-factor of the LT codes is the degree distribution in the encoding procedure,

because it is the only component responsible for the efficiency of these codes. It has been shown that LT codes perform very well for long messages length K . In our work, we consider two different degree distribution, for a small message length, we apply an algorithm for iterative optimization of the degree distribution by using an approach based on importance sampling [80]; otherwise for long message length, we consider optimized "Robust Soliton Distribution" as suggested in [81]. The used distributions are based on "Soliton Distribution", so the probability of degree-one symbol is less than the probability for degree-two symbol.

In our work, for small message length case, the source length K is set to 1000 and the following distribution, Eq.5.3, is applied:

$$p_i = \begin{cases} \eta_1, & \text{for } i = 1, \\ \eta_2, & \text{for } i = 2, \\ \eta_3, & \text{for } i = 100, \\ \frac{1}{i(i-1)}, & \text{for } i = 3, \dots, 99 \text{ and } i = 101, \dots, n. \end{cases} \quad (5.3)$$

The distribution is then normalized.

The optimized parameters are $\eta_{opt} = (0.083, 0.487, 0.032)$. The choice of parameter values is very important, because a bad choice could lead to poor performance of the decoding. Otherwise, for long message length case, the source length K is set to 10000 and the characterized parameters of Robust Soliton Distribution, c and δ , where c suitable positive constant and δ is the decoding failure probability, are respectively set to 0.02 and 0.01.

5.5 The proposed approach

In this research, we consider Manet scenarios where nodes are moving and a single source is transmitting multimedia data to N receiving nodes, [72]. Data packets are sent from source to destination through intermediate nodes that flood data messages into network exploiting PUMA protocol.

The entire network can be seen as sum of different sub-layers with intermediate destination nodes that have the task to decode information flow once they have received $K(1 + \varepsilon)$ packets i.e. the useful amount of packets for decoding. Then, those intermediate destina-

tion nodes recode the original information flow, scramble the order of data packets and finally, re-transmit encoded packets into the next sub-layer. Doing so, every intermediate destination nodes receive information flow from multiple source nodes. In this way, the intermediate node can collect the enough amount of packets before the end of the transmission period.

The purposes of this approach is to propose a joint source-channel coding technique based on LT code that can improve the robustness and reduce the end to end latencies of a multicast service in a scenario with multiple sensors distributed in the surrounding environment.

An exhaustive description of the presented approach is detailed in Section 5.6.

5.6 Experimental Results

The considered network is based on IEEE 802.11 standard. It is composed of three sub-layers that represent three main steps. In Fig.5.3 the overall network is shown. The entire

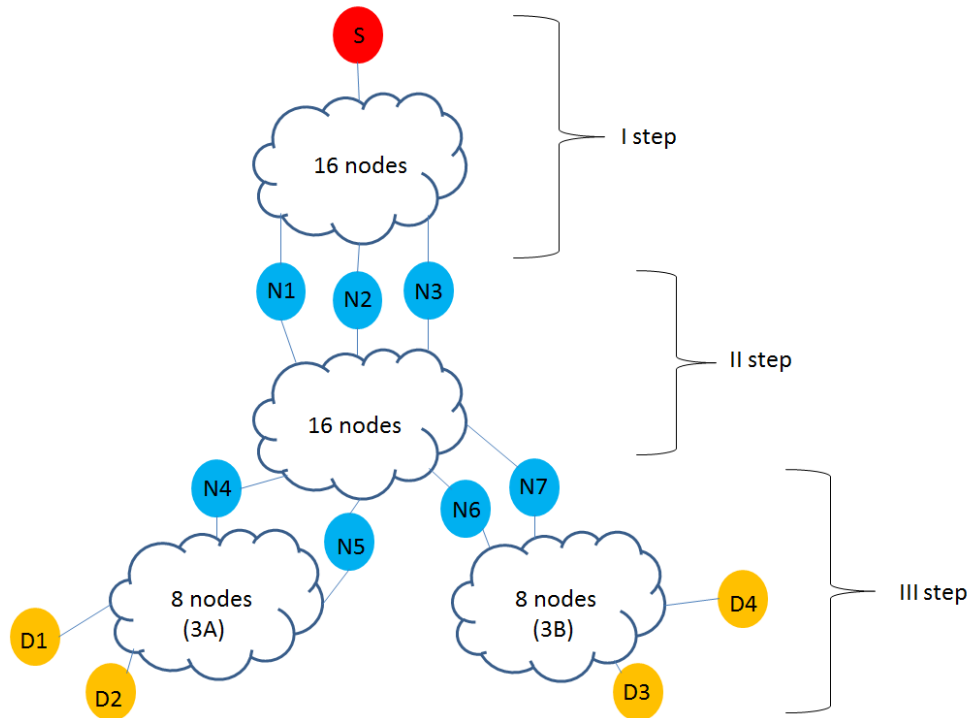


Figure 5.3: Example of network composed by three main sub-layers used for simulations.

network is the sum of 4 sub-networks, each of 16 or 8 nodes that are moving randomly with a velocity of 5 m/s, according to "Random Waypoint Model".

Refereing to Fig. 5.3, node S is the initial source and it performs source flow encoding and then it transmits the encoded data into the networks. N1,N2, N3...N7 nodes perform decoding and they send data into the new sub-network. D1, D2, D3 and D4 are the final destination nodes with the task to decode the entire multimedia content.

Three scenarios have been considered to evaluate our method: the first one is the ideal case where no packet loss is supposed to be; the second case is, instead, corrupted by packet loss and in particular in every sub-networks, a different Packet Loss Rate (PLR) has been added and consequently a different overhead has been considered for every step. The simulated losses are due MAC collisions, random losses and link failures. In the third scenario, the number of packets is increased till reaching 10000 pkts (packets) and the network is subject to different packet loss rates. In addition, we have evaluated the performances of scrambling technique for routing data packets compared to data routing without scrambling for each proposed scenarios.

In the first part of the network, the source encodes data flow using modified LT codes for a reduced number of packets, [80]. Once LT coding has been done, node S transmits 1000 packets plus an overhead of 200 packets that represents almost 20% of useful packets. Data packets are routed among mobile nodes till reaching intermediate nodes N1, N2 and, N3. At this point, intermediate receiving nodes will decode multimedia content once the necessary number of packets has been reached to regenerate the entire data flow. Source node (S) floods packets into the network without scrambling the order so that the net results robust to losses but there is no time saving.

In the second step, the intermediate nodes will become sources of the second sub-network and they will encode data flow exploiting the same LT encoding matrix. Data packets are sent through the network without following a sequential order but a random order i.e. scrambling of data packets has been performed according to Fibonacci $p - r$ sequence,[82], [74]. Every stream has a different permutation, which is generated by three keys. The choice of the keys is optimized for each sub-network. Every destination node does not know the used keys, and can generate different keys when it re-transmits the

collected packets. This technique has been applied also in the next step to N4, N5, N6 and N7 nodes when they become sources of the other two sub-networks but using different Fibonacci sequence values.

Packets sent from new sources and characterized by 25% of overhead, are transmitted into the net till they reach N4, N5, N6 and N7 destination nodes. Receiving nodes of the second step will LT decode and in the third step they are the sources of two new sub-networks and they LT encode and relay coded data flow in the network.

The third step is characterized of two different sub-nets. N4 and N5 nodes share the same encoding matrix and a different encoding matrix is shared by N6 and N7. N4 and N5 send data packets plus 19% of overhead inside a smaller sub-net composed of 8 nodes. The final receivers are D1 and D2 with the task to LT decode multimedia content. Likewise, N6 and N7 will send, following Fibonacci $p - r$ sequence, data flow into the left sub-net to the final receivers D3 and D4.

The transmission velocity is equal to 10 pkt/s. For simulations, NS-2 software [83] has been used on a laptop Intel Core 2, CPU T5200 with a frequency clock of 1.66 Hz and 1GB for RAM. For all the simulations, decoding starts once 1100 pkts have been collected (10% of overhead). In the first step, encoding time of the source for 1200 packets is 17.03 seconds, the transmission lasts 120 seconds and no losses for collision at MAC level or other losses are considered.

5.6.0.1 First Scenario (no PLR)

For each step of the network, the arrival times of the $k(1 + \varepsilon) = 1100$ packets ($\varepsilon = 0.1$) have been evaluated comparing the case where scrambling technique is performed and the case where no scrambling is considered, the results are shown in Figures 5.4, 5.5 and, 5.6.

Decoding time of the first 1100 packets has been evaluated for the 3 destination nodes (N1, N2 and N3) of the first step and reported in Table 5.1.

In the second step encoding times of 1250 packets have been evaluated for N1, N2 and N3, Table 5.2.

In this second phase, transmission lasts 125 seconds and 1% of packets is lost due to

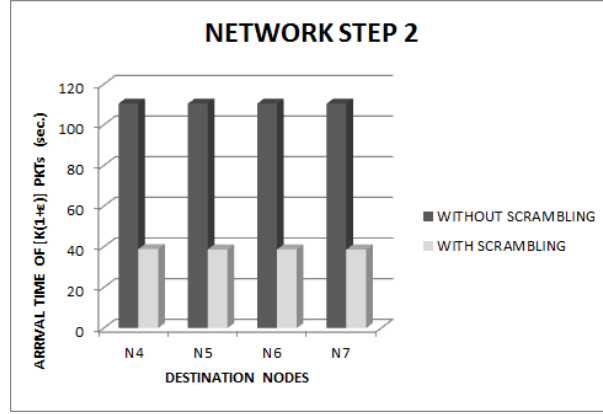


Figure 5.4: Comparison of the arrival time of $k(1+\varepsilon)$ packets between network that performs scrambling and the case where scrambling is not performed for the second step.

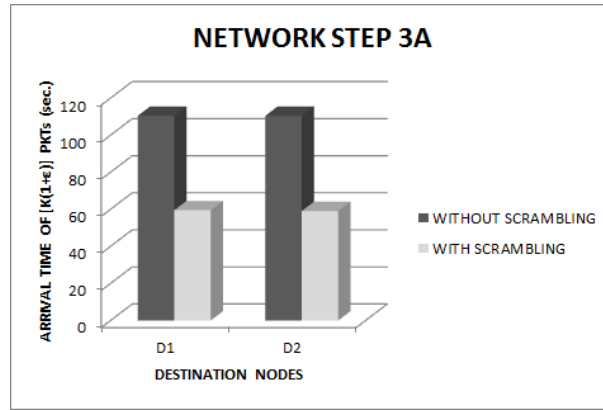


Figure 5.5: Comparison of the arrival time of $k(1+\varepsilon)$ packets between network that performs scrambling and the case where scrambling is not performed for 3A sub-network of the third step.

	N1	N2	N3
Decoding time (sec.)	5.78	5.79	5.80

Table 5.1: Decoding time in the first step

	N1	N2	N3
Encoding time (sec.)	18.83	18.83	18.83

Table 5.2: Encoding time in the second step

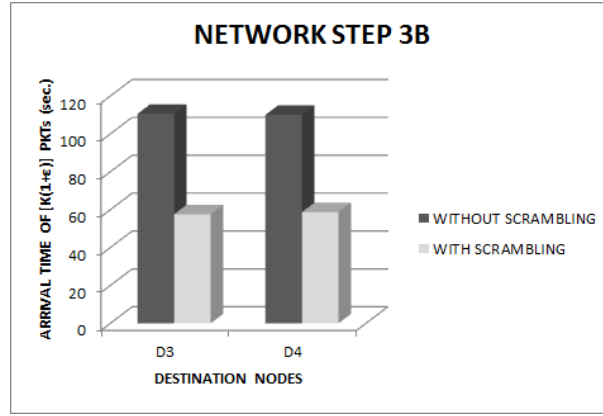


Figure 5.6: Comparison of the arrival time of $k(1+\epsilon)$ packets between network that performs scrambling and the case where scrambling is not performed for 3B sub-network of the third step.

	N4	N5	N6	N7
Decoding time (sec.)	6.25	6.14	6.00	6.00

Table 5.3: Decoding time in the second step

MAC collisions. Each receiver obtains data from all the sources. Decoding times for N4, N5, N6 and N7 addresses nodes are reported in the Table 5.3.

Let us consider the 3A sub-net with D1 and D2 as final receivers. Encoding times of 1190 pkts for N4 and N5 nodes have been evaluated and reported in the Table 5.4.

Transmission lasts 119 seconds and 1% of pkts are lost due to MAC collisions, and each addressee node obtains data from all the source. Decoding time for D1 and D2 are reported in Table 5.5.

Finally, sub-net with final addresses D3 and D4 is considered. Encoding times of 1144 pkts for N6 and N7 nodes are reported in Table 5.6. Transmission lasts 114 seconds and 1% of packets are lost for MAC collisions. Each addressee node obtains data from all the

	N4	N5
Encoding time (sec.)	17.12	17.12

Table 5.4: Encoding time for 3A in the third step

	D1	D2
Decoding time (sec.)	5.87	5.79

Table 5.5: Decoding time for 3A in the third step

	N6	N7
Encoding time (sec.)	15.92	15.92

Table 5.6: Encoding time for 3B in the third step

sources. Decoding time of the first 1100 pkts for D1 and D2 are reported in Table 5.7.

5.6.0.2 Second scenario (with PLR)

For the second scenario, different PLRs have been added in each part of the network. In the sub-network related at step 2, PLR is equal to 11%; in the sub-network called 3A , PLR=6% and finally in 3B sub-network PLR=3%. In figures 5.7, 5.8 and, 5.9 the results are shown. As for the first scenario, performance comparisons have been made evaluating routing with scrambling approach and routing without scrambling.

In Figures 5.10, 5.11 and 5.12 arrival times comparisons at different packet loss rates, are presented for each step of the network. Performance results show that applying scrambling technique to data routing, arrival times of packets to destination are reduced in contrast to conventional multicast routing.

5.6.0.3 Third Scenario (Increased number of pkts and with PLR)

LT codes are generally designed for high number of packets (almost 10000 pkts) and for the previous scenarios an optimized version of LT codes has been implemented. In this final scenario we have increased the number of packets delivered into a network subject to a different packet loss rates and the degree distribution for LT code is the "Robust

	D3	D4
Decoding time (sec.)	5.49	5.62

Table 5.7: Decoding time for 3B in the third step

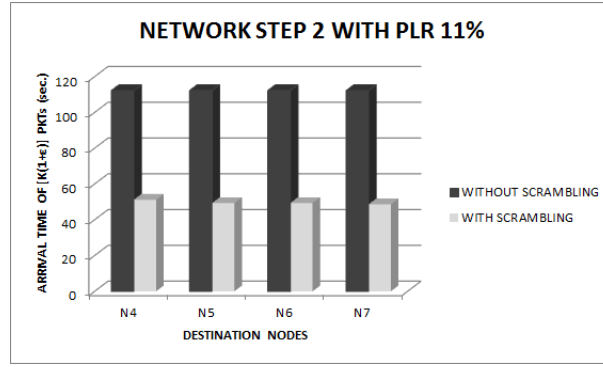


Figure 5.7: Comparison of the arrival time of $k(1+\varepsilon)$ packets between network that performs scrambling and the case where scrambling is not performed for the second step.

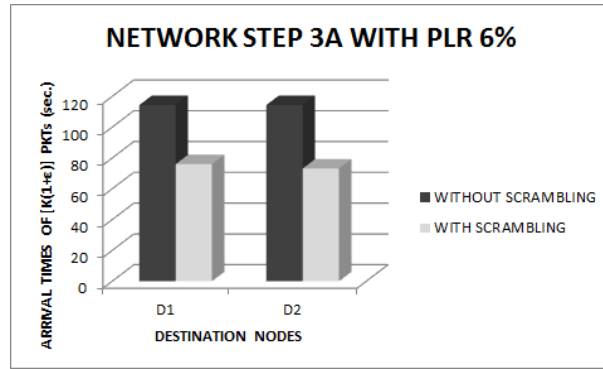


Figure 5.8: Comparison of the arrival time of $k(1+\varepsilon)$ packets between network that performs scrambling and the case where scrambling is not performed for 3A sub-network of the third step.

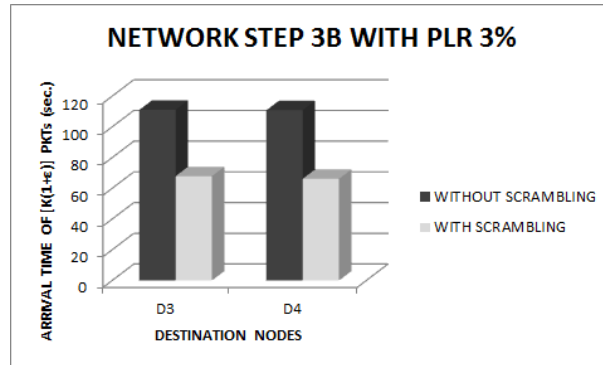


Figure 5.9: Comparison of the arrival time of $k(1+\varepsilon)$ packets between network that performs scrambling and the case where scrambling is not performed for 3B sub-network of the third step.

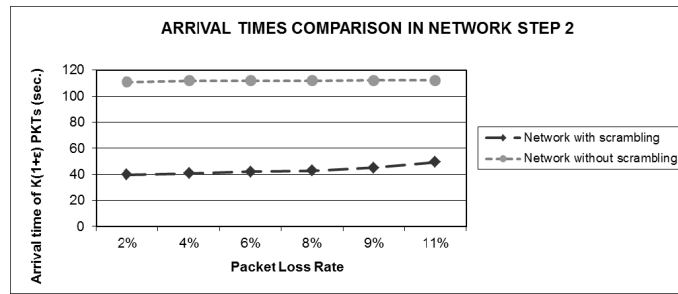


Figure 5.10: Arrival times comparison at different PLRs in network step 2.

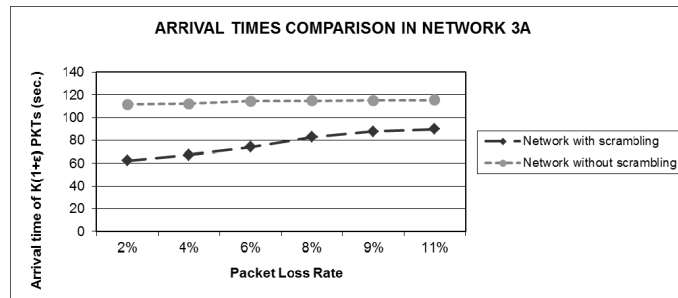


Figure 5.11: Arrival times comparison at different PLRs in network 3A of the third step.

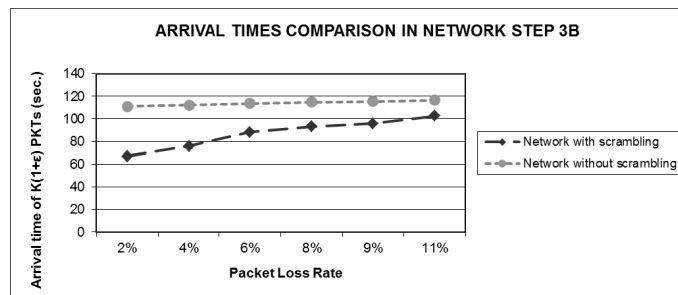


Figure 5.12: Arrival times comparison at different PLRs in network 3B of the third step.

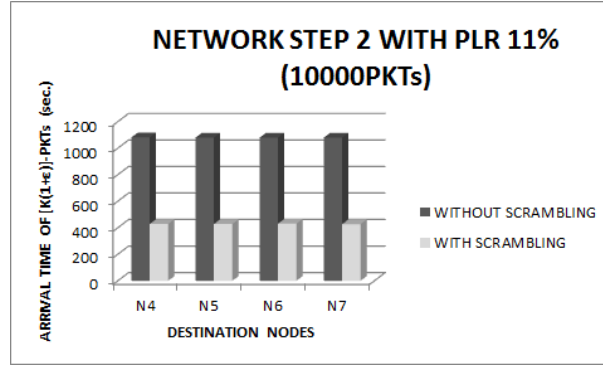


Figure 5.13: Comparison on arrival times for 10000 packets between network that performs scrambling and the case where scrambling is not performed for the second step.

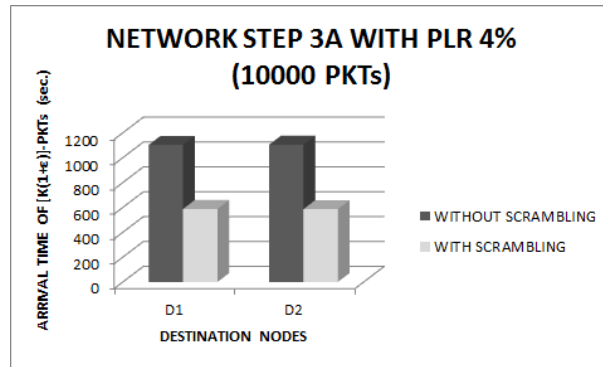


Figure 5.14: Comparison on arrival times for 10000 packets between network that performs scrambling and the case where scrambling is not performed for 3A sub-network of the third step.

Soliton” with optimized characteristics parameters c and δ as suggested in [81]. In the second step of the network, PLR is equal to 11%; in network step 3A, $PLR = 4\%$ whereas in network step 3B, $PLR = 3\%$. The used simulation parameter values are the same of previous scenarios instead simulation times are increased due to the increased number of pkts. For this case, LT decoding starts on average once 10600 packets are collected ($\varepsilon = 0.06$). Based on these considerations, we expected consistent performance values with previous cases and the performed simulations have confirmed our expectations. In Figures 5.13, 5.14 and 5.15, performance comparisons have been made evaluating routing with scrambling approach and routing without scrambling.

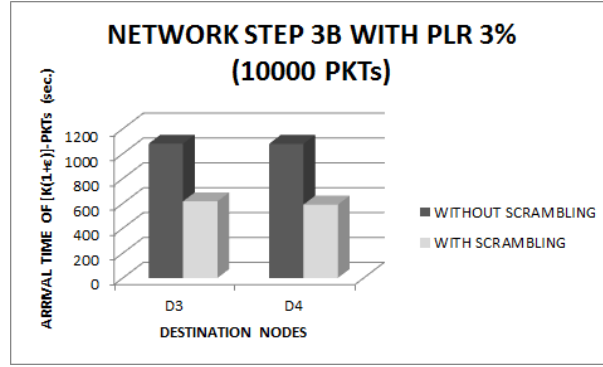


Figure 5.15: Comparison on arrival times for 10000 packets between network that performs scrambling and the case where scrambling is not performed for 3B sub-network of the third step

5.6.1 Remarks

In the first step of the proposed network, no scrambling is performed and this means that there is no time saving for packets decoding. However, the net is robust to losses for two main reasons: first, PUMA is based on a multi-path approach so that destination node receives packets from more sources and second, a modified LT coding is applied. In the second sub-net composed of 16 nodes, the number of collisions is greater than 8 nodes sub-net, nevertheless packet loss is 1% in both cases. The reason lies in the fact that in 16 nodes sub-net all the addressees obtain data from all the tree sources instead in the third step, only 2 nodes are sources for each 8 nodes sub-net.

5.7 Conclusion

The present chapter has addressed stream reliability problems for multimedia delivering in a MANET. In particular, a method based on joint source-channel coding based on LT code has been described. Some applications as mobile wireless sensors networks distributed in bounded environments can benefit from this technique because introducing a joint source-channel coding and a scrambling approach, it is possible to provide robust and real-time multimedia content services. Multimedia communications can benefit from our proposed method that reduces receiving time of useful packets quantity. At the same time, it

maximizes the quality of service reconstructing lost packets. Moreover, a key-factor of the proposed approach is the use of the considered scrambling technique that allows an improvements on delivery times.

Chapter 6

Image Search

6.1 Introduction

Content discovery, delivery, and streaming are the basic functionalities of current content-centric Internet services. The first generation of retrieval systems was based on the use of metadata describing the semantic content of a multimedia document, usually extracted by manual procedures. However, Future Internet content aware services will require more efficient functionalities for inspection, crawling, recognition, categorization, and indexing of digital content with minimal human intervention. Thus, their implementation requires fast and reliable algorithms for locating and tracking complex objects irrespective of their actual orientation and scale. The estimation of position, rotation, and scale of a given template in a complex scene is a classic task in computer vision applications as printed circuit board inspection, autonomous vehicle guidance, remote monitoring, and surveillance.

At this aim, different techniques have been proposed. Matching the details of an observed image with a given template can be performed by a straightforward solution in which the likelihood functional map, or any other similarity index such as the Normalized Cross Correlation (NCC) or the Sum of Absolute Differences (SAD), related to a discrete set of rotated and scaled versions of the template is computed. Although conceptually simple, this solution is highly inefficient and susceptible of converging to local maxima if either scale or orientation are under-sampled. To face the computational complexity problem, sequential detection and coarse to fine estimation procedures based on multires-

olution hierarchical template matching have been investigated [84], [85], [86]. Towards this goal, a technique for reducing the possible candidate locations based on thresholding of NCC between the unrotated template and a sliding window of the searched image has been proposed in [87]. As an alternative to full search, different optimization schemes, such as genetic [88] and particle swarm [89] algorithms have been introduced.

More effective techniques based on the algebraic invariants theory allowing the design of rotation and/or scale invariant template decomposition have been investigated [90]. In general, a feature vector of rotation and scale invariants first extracted from the template. The same procedure is applied to a sliding window in the under-test image. Then, for each location, the similarity between the template and the sliding window feature vectors is computed and the maximum is extracted. Template detection can then be performed by thresholding the maximum similarity index, while the position corresponding to that maximum constitutes the estimated pattern location. Once the template location has been determined, orientation and scale can be finally estimated.

These techniques mainly differ for the invariants constituting the feature vector. Simple invariants derived from the image central moments were proposed by Hu in 1962, [91]. A comprehensive reference concerning invariant moments can be found in [92]. Among the others, invariants based on Zernike moments have attracted the interest of many researchers. In fact, it has been demonstrated that Zernike moments and pseudo-Zernike moments have the best overall performance in the retrieval among the used moments, [93]. These moments are obtained by the decomposition of a template on complex Circular Harmonic Functions (CHFs) that form a complete orthogonal basis on a unit disc. Due to a general property of the CHFs, a pattern can be easily steered by multiplying the expansion coefficients by complex exponential factors whose phase is proportional to the rotation angle. As a consequence, rotation invariants can be easily obtained by considering the magnitude of the expansion coefficients.

Moreover, when orthogonal expansions based on CHFs are employed, the localization procedure proposed in [94] for Gauss-Laguerre CHFs can be generalized, and a fast Maximum Likelihood (ML) procedure for joint location, orientation, and scale estimation can be devised. For instance, in the case of Zernike CHFs, in the query image the circular

area surrounding the object to be localized is first selected, then the portion of the image inside the circle is approximated by a truncated expansion in terms of Zernike polynomials up to a given order. The Likelihood functional is then expressed in terms of Zernike expansion coefficients. The steering property of the CHF's allows to apply fast optimization techniques based, for example, on the Newton-Raphson method.

Many applications require detection and localization of complicated patterns that have to be distinguished from similar objects differing only for a few fine details. In this case, direct use of Zernike moments for computing the ML functional requires a great number of expansion terms.

In this chapter, two image recognition techniques based on Zernike moments and Gauss-Laguerre Transform are analyzed, [95], [96], [97]. Here, in order to manage objects of arbitrary shape, while reducing the computational cost, we partition the pattern to be localized into small square blocks using a quadtree decomposition.

When dealing with Zernike moments, the size of each block is adapted to the local image content and is controlled by the quadratic norm of the error corresponding to the truncated Zernike expansion, [95]. The quadtree blocks are then ranked with respect to the energy of the low pass filtered gradient or, equivalently, to Fisher's information on location and rotation. ML estimation of the location and the orientation of the first block of the quadtree is performed by means of an iterative quasi-Newton procedure making use of Zernike moments. The estimation algorithm is an extension of the technique proposed for Gauss-Laguerre approximation in [94].

Compared to the traditional ML technique based on the matching of a candidate image with a whole set of rotated versions of the pattern, this procedure requires a local maximization of functionals derived by Zernike coefficients.

The estimated location and orientation are then used for verifying whether the current image contains the second block of the rank ordered template quadtree list. If the quadratic norm of the difference between the subset of the reference template, constituted by the first and the second square of the quadtree, and the current image falls below a predefined threshold, the next block of the quadtree list is analyzed. The procedure ends either if the energy of the difference exceeds a threshold or the last list element has been processed.

Then a novel method based on a Riesz hypercomplete basis whose elements are the Laguerre-Gauss Circular Harmonic functions is presented, [97].

Laguerre-Gauss Circular Harmonic functions are complex, polar separable filters characterized by harmonic angular shape, a useful property to build rotationally invariant descriptors. Using this basis we can efficiently represent complex images containing many details. In fact, given a region of interest of an image, the support is partitioned into smaller and smaller square blocks whose width is controlled by the norm of the error corresponding to a truncated expansion in terms of Gauss Laguerre CHF's.

To further reduce the computational complexity of the Maximum Likelihood estimation implementation, the elements of the quadtree blocks are ranked with respect to their amount of Fisher's information on location and rotation, proportional to the energy of the low pass filtered gradient. Then, a procedure based on the sequential matching of each block of the ranked quadtree list is applied.

The chapter is organized as follows: two similar methodologies for image search are described. First, the mathematical properties of Zernike polynomial expansion are summarized in Section 6.2. In Section 6.3 the overall scheme of the proposed approach is presented. In subsection ?? the quadtree based procedure is described, and the template detection and orientation and location estimation algorithms are illustrated in subsection 6.3.2. Performances assessments are reported in Section 6.4. Then, the image retrieval procedure based on Laguerre-Gauss polynomials is explained and in Section 6.5.1, their mathematical properties are summarized. In Sections 6.5.2 and 6.5.3, the followed algorithm is reported and in Section 6.6 the experimental results are illustrated. Finally conclusions are drawn in Section 6.7.

6.2 Zernike polynomial expansion

Let $\mathbf{x} = [x_1, x_2]$ denote the cartesian coordinates of points in the real plane \mathbb{R}^2 . The polynomials introduced by Zernike in 1934 form a complete orthogonal basis set defined on the unit circle $x_1^2 + x_2^2 \leq 1$ and belong to the CHF class, i.e., to the class of complex, polar separable functions with harmonic angular shape, as defined in [98], [67]. Specifi-

cally, denoting with $\tilde{V}_{nm}(\rho, \theta) = V_{nm}(\rho \cos \theta, \rho \sin \theta)$ the expression of Zernike polynomials $V_{nm}(x_1, x_2)$ of order n and repetition index m in polar coordinates $\rho = \sqrt{x_1^2 + x_2^2}$ and $\theta = \tan^{-1}(x_2/x_1)$, we have

$$\tilde{V}_{nm}(\rho, \theta) = R_{nm}(\rho)e^{jm\theta}, \quad (6.1)$$

where $R_{nm}(\rho)$ is Zernike radial profile defined as:

$$R_{nm}(\rho) = \begin{cases} \sum_{s=0}^{(n-|m|)/2} \frac{(-1)^s (n-s)! \rho^{n-2s}}{s! (\frac{n+|m|}{2}-s)! (\frac{n-|m|}{2}-s)!} & \text{if } n-|m| \text{ even} \\ 0 & \text{if } n-|m| \text{ odd,} \end{cases} \quad (6.2)$$

in Figure ?? some examples of Zernike polynomials are shown.

The orthogonality condition on Zernike polynomials gives:

$$\int_0^{2\pi} \int_0^1 V_{nl}^*(\rho, \theta) V_{mk}(\rho, \theta) \rho d\rho d\theta = \frac{\pi}{n+1} \delta_{nm} \delta_{lk}, \quad (6.3)$$

where δ_{nm} denotes Kronecker delta. The radial polynomials $R_{nm}(\rho)$ satisfy the relation:

$$\int_0^1 R_{nl}(\rho) R_{ml}(\rho) \rho d\rho = \frac{1}{2(n+1)} \delta_{nm}. \quad (6.4)$$

For a continuous function $f(\mathbf{x})$, inside the unit disk centered at \mathbf{x}_0 , the following Zernike polynomial expansion holds:

$$f(\mathbf{x}) = \sum_{n=0}^{\infty} \sum_{m=-\infty}^{+\infty} A_{nm}(\mathbf{x}_0) V_{nm}(\mathbf{x} - \mathbf{x}_0), \quad (6.5)$$

with expansion coefficients $A_{nm}(\mathbf{x}_0)$ given by:

$$A_{nm}(\mathbf{x}_0) = \frac{n+1}{\pi} \int \int_{\|\mathbf{x}-\mathbf{x}_0\| \leq 1} f(\mathbf{x}) V_{nm}^*(\mathbf{x} - \mathbf{x}_0) dx_1 dx_2. \quad (6.6)$$

Since $\tilde{V}_{nm}(\rho, \theta)$ can be rotated by an angle φ by multiplying it by a factor $e^{-jm\varphi}$, the expansion coefficients $A_{nm}^{(\varphi)}(\mathbf{x}_0)$ of an image $f(\mathbf{x})$ rotated by an angle φ are related to the expansion coefficients $A_{nm}(\mathbf{x}_0)$ of $f(\mathbf{x})$ by the following relationship:

$$A_{nm}^{(\varphi)}(\mathbf{x}_0) = A_{nm}(\mathbf{x}_0) e^{-jm\varphi}. \quad (6.7)$$

This leads to the well-known rotational invariance property $|A_{nm}^{(\varphi)}| = |A_{nm}|$.

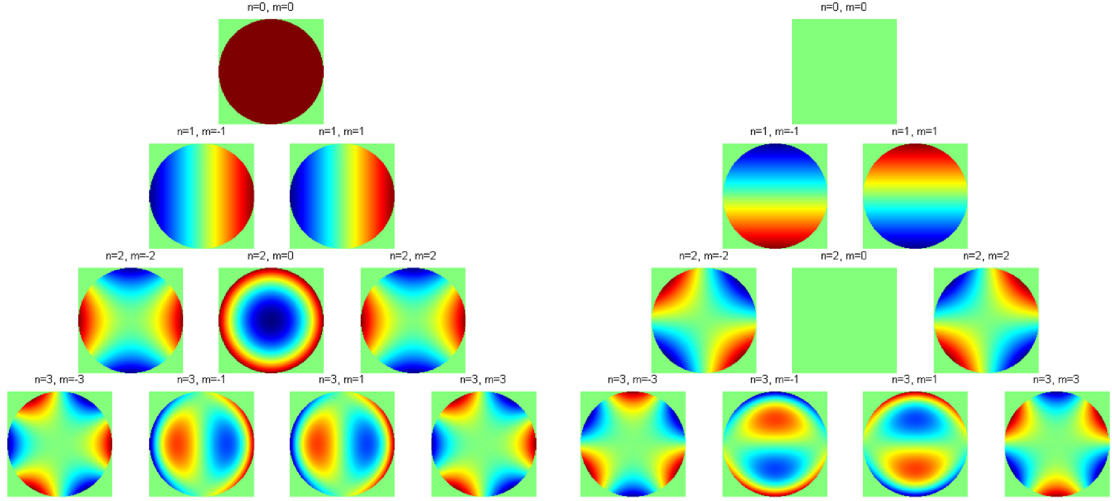


Figure 6.1: Zernike filters with order up to $n = 3$ and $m = 3$, real part and imaginary part, respectively.

To calculate Zernike moments, the image (or the region of interest) is first projected onto the unit disk following the approach proposed in, [99], [100]. Pixels outside the unit circle are not considered.

The accuracy of Zernike moments computed via Equation (6.6) is affected by the geometric approximation error, [101], [102]. This is due to the fact that the area covered by the square pixels involved in the computation of Zernike moments is not exactly the unit disk. However, by computing Zernike moments in polar coordinates this error can be minimized [101].

In practical situations, the reconstruction of the image is performed by using a finite number of Zernike moments i.e.:

$$\hat{f}(\rho, \theta) = \sum_{n=0}^N \sum_{m=-M}^{m=+M} A_{nm} V_{nm}(\rho, \theta). \quad (6.8)$$

The reconstruction error depends on both the number of employed moments and on the size of the image.

6.3 The Proposed approach in Zernike domain

In this section the proposed approach in Zernike domain is described, [95]. In Figure 6.2 the overall scheme of the proposed method is shown. Our approach is based on Zernike moments, used to compute the likelihood between the query image and the images in the database. Zernike moments are computed by using the quadtree decomposition, this ensures that, for each point, Zernike moments are a good representation of the neighborhood. Once Zernike quadtree decomposition is performed, the likelihood map is computed and used for the image recognition and for the rotation angle estimation.

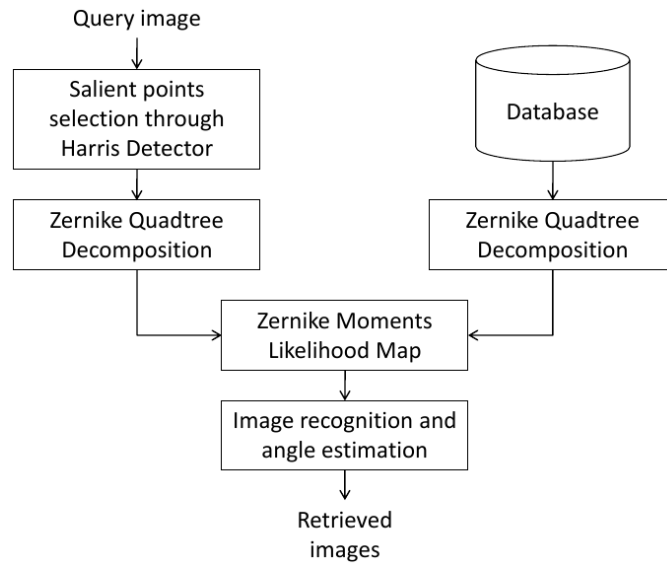


Figure 6.2: The architecture of the proposed image retrieval system. First, salient points are extracted by means of Harris detector and a Zernike moments quadtree decomposition is applied. Then a sequential detection and estimation procedure is performed to retrieve the candidate image inside the database.

6.3.1 The quadtree decomposition

When dealing with image retrieval techniques, descriptors based on Zernike moments can represent a very powerful tool due to their discriminating power, noise resilience, information redundancy, and reconstruction capability. However, in order to manage complex objects of arbitrary shape that have to be distinguished from similar objects

differing only for a few fine details, the required number of Zernike moments is very high. Hence, splitting the region of interest containing the under-test object into smaller squares using a *quadtree* decomposition, it is possible to reduce the computational complexity, see Figure 6.3.

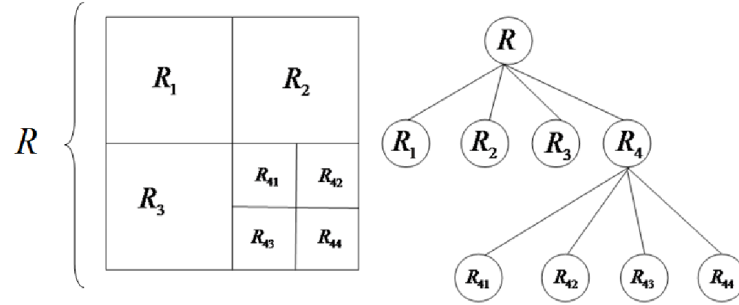


Figure 6.3: Example of how an image can be split in more blocks according to a quadtree decomposition.

Then, for each square of the decomposition, the truncated Zernike expansion is applied to the portion of the pattern inside the circumcircle of the actual square.

Let us define with $f(\mathbf{x})$ the original image, with $\hat{f}(\mathbf{x})$ the reconstructed image with a predefined number of Zernike moments, and with $w_T(\mathbf{x})$ a square window of unitary width. Then, given a square region of interest R , centered on ξ and with width δ , let $P(R)$ be the predicate

$$P(R) = \left\{ \left\| w_T \left(\frac{\mathbf{x} - \xi}{\delta} \right) [f(\mathbf{x}) - \hat{f}(\mathbf{x})] \right\|^2 < \gamma \right\}, \quad (6.9)$$

which is *True* if the squared norm of the approximation error is smaller then the threshold γ .

At the first step, $P(R)$ is evaluated. If $P(R) = \text{True}$ the approximation is good enough and the decomposition stops. Otherwise, to reduce the approximation error, without increasing the number of Zernike moments, we split R into four squares R_k , $k = 1, \dots, 4$, with halved width and we apply the truncated Zernike expansion in each of them. If for any region R_i , $P(R_i)$ is *False* we further partition that region in other four *regions* R_{i_k} , $k = 1, \dots, 4$ and iterate the whole procedure.

As illustrated in Figure 6.4, using the quadtree decomposition, R is partitioned into smaller and smaller square regions $R^{(n)}$, so that for each region $R^{(n)}$, $P(R^{(n)}) = True$.

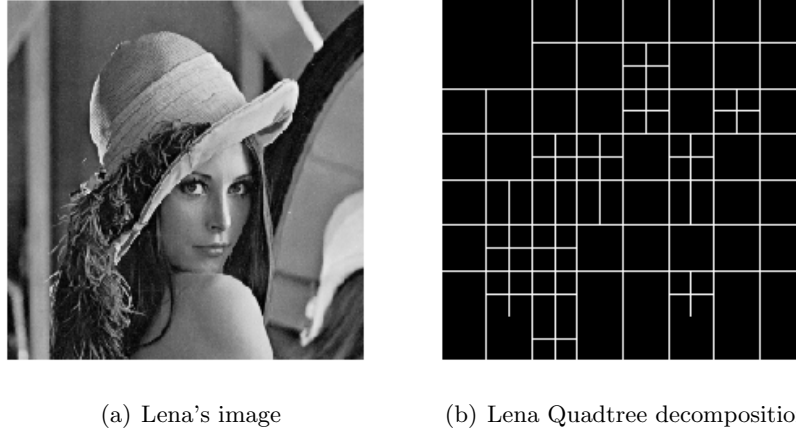


Figure 6.4: Example of quadtree decomposition by means of Zernike moments computation on Lena image.

A sequential procedure that verifies whether the candidate image contains each square of the quadtree list and refines, step by step, the location and orientation estimates, [96], is employed in order to search for complex pattern in large multimedia database. In order to design a fast sequential detection and estimation procedure and to reduce the retrieval time, we propose to start the search of the quadtree elements starting from those which are simpler to locate, i.e. those elements that can represent the local properties of the pattern.

Pattern location, rotation, and scale estimation accuracies are strictly related to Fisher's information and this quantity is proportional to the magnitude of the energy of the image gradient and to the energy of the angular derivative, or, equivalently, to the square of the effective spatial and angular bandwidths [94].

Thus, defining as salient points of a pattern those points characterized by wide spatial and angular bandwidths, the quadtree building procedure can be summarized as follows:

- for each reference pattern, a ranked list of salient points is computed on the basis of the local Fisher's information on location, translation and scaling;
- Zernike expansion based on Quadtree decomposition is applied to the neighborhood of each salient point;

- the sequential procedure verifying whether the candidate image contains each neighborhood of each salient point as described in the next paragraphs is applied until a stopping condition is verified.

In practice, we employ Harris corner detector algorithm, [103], to select salient points motivated by the fact that corners meet the wide spatial and angular bandwidth condition of salient points.

To illustrate the rational of the use of Harris detector we observe that, for a template $f(\mathbf{x})$, eventually rotated by an angle φ , Fisher's information matrix on its location $J_{\mathbf{b}}$ is proportional to the energy tensor of the image gradient $\nabla f_{\mathbf{x}} = [f_{x_1} f_{x_2}]^T$. In fact, as demonstrated in [94],

$$J_{\mathbf{b}} = \frac{4}{N_0} R_{\varphi} E_{\nabla f_{\mathbf{x}}} R_{\varphi}^T, \quad (6.10)$$

where R_{φ} is the rotation matrix, defined as:

$$R_{\varphi} = \begin{bmatrix} \cos \varphi & \sin \varphi \\ -\sin \varphi & \cos \varphi \end{bmatrix}. \quad (6.11)$$

and

$$E_{\nabla f_{\mathbf{x}}} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} M_f(\mathbf{x}) dx_1 dx_2, \quad (6.12)$$

where $M_f(\mathbf{x})$ is the spatial density of the gradient energy:

$$M_f(\mathbf{x}) = \nabla f_{\mathbf{x}}(\mathbf{x}) \nabla f_{\mathbf{x}}^T(\mathbf{x}). \quad (6.13)$$

Thus we select as salient points those points corresponding to the local maxima of the spatial density of the gradient energy, irrespective of the rotation matrix. This in turn implies that we look for those points characterized by positive, large eigenvalues of $M_f(\mathbf{x})$. By denoting with $\lambda_1(\mathbf{x})$ and $\lambda_2(\mathbf{x})$ the eigenvalues of $M_f(\mathbf{x})$ we select as salient points the local maxima of Harris corner detector functional, [103]:

$$\Gamma_f(\mathbf{x}) = \lambda_1(\mathbf{x}) \lambda_2(\mathbf{x}) - k(\lambda_1(\mathbf{x}) + \lambda_2(\mathbf{x}))^2, \quad (6.14)$$

where $0 < k < 0.25$ is a constant.

The main advantage of choosing $\Gamma_f(\mathbf{x})$ is that it can be evaluated without explicitly computing the eigenvalues. In fact it can be demonstrated that

$$\Gamma_f(\mathbf{x}) = \det[M_f(\mathbf{x})] - k \text{trace}^2[M_f(\mathbf{x})]. \quad (6.15)$$

In addition, to prevent false detections, only those points for which $\Gamma_f(\mathbf{x})$ exceeds a predefined threshold are considered. Moreover, since Fisher's information on location is proportional to the integral of $M_f(\mathbf{x})$ on the template support, in order to sort the quadtree elements with respect to the achievable location estimate accuracy, salient points can be equivalently ranked based on the local average of $\Gamma_f(\mathbf{x})$.

As an example, in Figure 6.5(b) the neighborhoods of the first ten salient points ranked with respect to $\Gamma_f(\mathbf{x})$ are shown. In Figure 6.5(c) the reconstructed image by means of the first 5 terms of Zernike expansion are also reported.

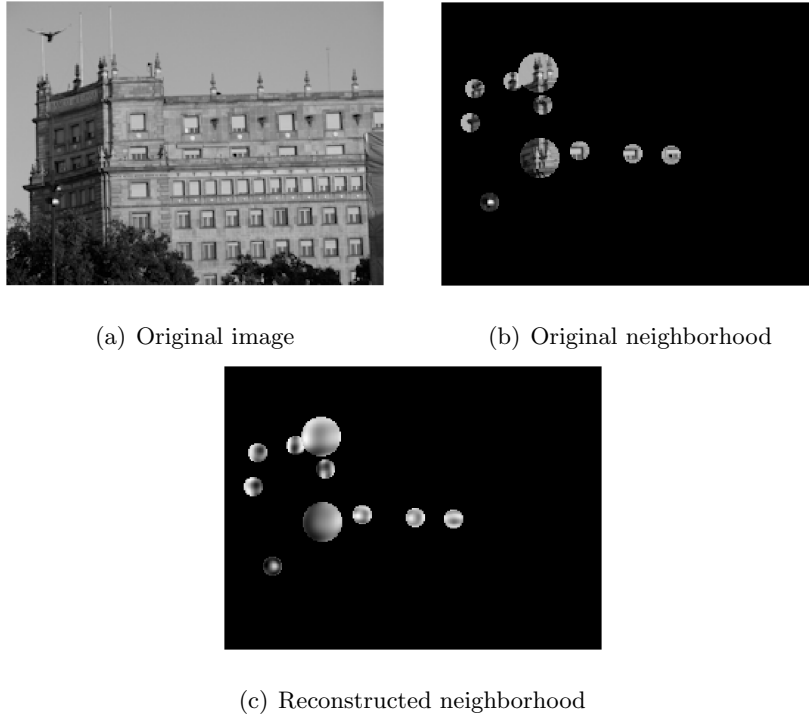


Figure 6.5: In this Figure, the original image (a), the obtained neighborhoods with Zernike expansions (b) and the reconstructed neighborhoods (c) of the salient points are shown. The size of the neighborhoods is chosen according to the quadtree decomposition.

Finally we observe that, since the eigenvalues of $M_f(\mathbf{x})$ are rotation invariant, Harris detector is rotation invariant too.

6.3.2 Rotation and Location Estimation Procedure

To detect the pattern belonging to the first block of the ranked quadtree list, an iterative quasi-Newton procedure with also the Maximum Likelihood estimate of both location and orientation of the pattern of the considered block in each candidate image is calculated, since rotation of a pattern simply produces a linear phase shift of each Zernike expansion coefficient, proportional to the order of the angular harmonic.

Let $f(\mathbf{x})$ be the candidate image that potentially contains a noisy, translated, and rotated version of the quadtree detail $g(\mathbf{x})$ of the template. At the position \mathbf{b} and orientation φ we have:

$$w[R_\varphi(\mathbf{x} - \mathbf{b})]f(\mathbf{x}) = \omega[R_\varphi(\mathbf{x} - \mathbf{b})]g[R_\varphi(\mathbf{x} - \mathbf{b})] + v(\mathbf{x}), \quad (6.16)$$

where $w(\mathbf{x})$ is a generic window, $v(\mathbf{x})$ is the observation noise modeled as a white, zero-mean Gaussian random field with power density spectrum equal to $(N_0/4)$ and R_φ is the rotation matrix (6.11).

The estimation of \mathbf{b} and φ can be performed by maximizing the Log-Likelihood functional $\Lambda[f(\mathbf{x}); \mathbf{b}, \varphi]$:

$$\ln \Lambda[f(\mathbf{x}); \mathbf{b}, \varphi] = -\frac{2}{N_0} \int \int |w[R_\varphi(\mathbf{x} - \mathbf{b})]|^2 |f(\mathbf{x}) - g[R_\varphi(\mathbf{x} - \mathbf{b})]|^2 d\mathbf{x}. \quad (6.17)$$

Thus the Zernike Moments Likelihood Map is defined as follows:

$$ZMLM(\mathbf{b}) = \max_{\varphi} \left\{ \ln \hat{\Lambda}[f(\mathbf{x}); \mathbf{b}, \varphi] \right\} \quad (6.18)$$

where $\hat{\Lambda}$ is the Likelihood functional computed by using Zernike moments. The detailed description of the representation of $\hat{\Lambda}$ in terms of Zernike moments and the derivation of the rotation estimator are reported in A.

The estimated location $\hat{\mathbf{b}}$ is then given by the value of \mathbf{b} corresponding to the maximum of $ZMLM(\mathbf{b})$:

$$\hat{\mathbf{b}} = Arg \left\{ \max_{\mathbf{b}} [ZMLM(\mathbf{b})] \right\}. \quad (6.19)$$

Then, in the case of content discovery, the template image is, first, compared with all the images in the database through $ZMLM^{Max} = Max[ZMLM(\mathbf{b})]$ evaluated on the first block of the quadtree. The database images are ranked according to the associated

$ZMLM^{Max}$. The one showing the highest value is retained as the best candidate for matching the first ranked quadtree block, while the location $\hat{\mathbf{b}}_1$ of the absolute maximum and the corresponding $\hat{\varphi}_1 = \hat{\varphi}(\hat{\mathbf{b}}_1)$ constitute the initial estimate of the pattern position and orientation.

This estimate is iteratively refined, block by block. Specifically, let $\hat{\mathbf{b}}_k$ be the estimate of the location of the first block and $\hat{\varphi}_k$ the estimate of the template orientation after processing the first k blocks of the quadtree.

The estimate $[\hat{\mathbf{b}}_{k-1}, \hat{\varphi}_{k-1}]$ is employed to verify whether the candidate image contains the k -th block of the rank ordered list of the quadtree elements as follows. Let $\delta\mathbf{x}_k$ be the displacement of the center of the k -th block of the quadtree with respect to the center of the first block. Then, the predicted location $\tilde{\mathbf{c}}_k^p$ of the center of the k -th block on the basis of the processing of the first $(k-1)$ blocks is computed:

$$\tilde{\mathbf{c}}_k^p = \hat{\mathbf{b}}_{k-1} + R_{\hat{\varphi}_{k-1}} \delta\mathbf{x}_k, \quad (6.20)$$

where $R_{\hat{\varphi}_{k-1}}$ is the rotation matrix.

The ZMLM of the k -th block is evaluated only for a limited set of possible locations, falling inside a small neighborhood of $\tilde{\mathbf{c}}_k^p$. In addition, the quasi-Newton procedure, adopted for building the ZMLM of the n -th block, is initialized using $\hat{\varphi}_{k-1}$. The neighborhood width δw_k accounts for the prediction error and can be set proportional to its standard deviation for which the following relationship holds:

$$\sigma_{\tilde{\mathbf{c}}_k}^2 = \sigma_{\hat{\mathbf{b}}_{k-1}}^2 + \sigma_{\hat{\varphi}_{k-1}}^2 |\delta\mathbf{x}_k|^2, \quad (6.21)$$

where $\sigma_{\hat{\mathbf{b}}_{k-1}}^2$ and $\sigma_{\hat{\varphi}_{k-1}}^2$ are the variances of $\hat{\mathbf{b}}_{k-1}$ and $\hat{\varphi}_{k-1}$, respectively.

Computation of these variance would require the knowledge of the statistics of the image data base. In practice, the neighborhood width can be computed by resorting to the inverse of Fisher's information. At this aim we recall that the information on template location $J_{\mathbf{b}}$ is given by (6.10), while the information J_{φ} on the orientation of a template $g(\mathbf{x})$ is equal to, [94]:

$$J_{\varphi} = \frac{4}{N_0} E_{g_{\varphi}} \quad (6.22)$$

where E_{g_φ} is the total energy of the template angular derivative

$$E_{g_\varphi} = \int_0^\infty \int_0^{2\pi} \left[\frac{\partial g(\rho, \theta)}{\partial \theta} \right]^2 \rho d\theta d\rho. \quad (6.23)$$

It can be easily verified that E_{g_φ} is proportional to the effective angular bandwidth of the template. In addition J_φ can be expressed in terms of the coefficients Z_{nm}^g of Zernike expansion of $g(\mathbf{x})$ as follows:

$$J_\varphi = \frac{4}{N_0} \sum_{n=0}^\infty \sum_{m=-\infty}^{+\infty} \frac{\pi n^2}{(n+1)\sigma^2} |Z_{nm}^g|^2. \quad (6.24)$$

Further details on Fisher's information matrix and on Rao-Cramer lower bound for the estimation of the location, rotation, and scale of a pattern in [94].

If the energy of the difference between the subset of the reference template, constituted by the first k blocks of the quadtree and the current image falls below a predefined threshold, location, and rotation of the image are refined and the next block analyzed.

In particular, let $Z_{nm}^{g_h}$ be Zernike coefficients of the h -th block of the quadtree associated to g , at the k -th stage the following functional is maximized:

$$\ln \Lambda^{(k)}[f(\mathbf{x}); \mathbf{b}, \varphi] = -\frac{2}{N_0} \sum_{n=0}^N \sum_{m=-M}^{+M} \sum_{h=1}^K \frac{\pi}{(n+1)\sigma^2} \left| Z_{nm}^f(\mathbf{b} + R_\varphi \delta \mathbf{x}_h) - Z_{nm}^{g_h} e^{jm\varphi} \right|^2. \quad (6.25)$$

In practice, only a few blocks, determined by the stopping condition, are employed in order to refine the initial estimate. We intend for stopping condition: the energy of the gradient of the subset of the template image constituted by the first K blocks of the ordered quadtree list. If the energy exceeds a fraction of the energy of the gradient of the whole template we stop adding blocks to the subset. In addition, when the magnitude of $\delta \mathbf{x}_k$ is sufficiently high, the actual rotation can be estimated on the basis of the orientation of the line connecting the centers of the first and the k -th block. Nevertheless, evaluation of (6.25) is required in order to decide about the presence of the searched template in the current image. In fact, if at k -th stage $\ln \Lambda^{(k)}[f(\mathbf{x}); \hat{\mathbf{b}}_k, \hat{\varphi}_k]$ falls below a predefined threshold, or equivalently the energy of the difference between the first k blocks of the template quadtree and the current image exceeds a corresponding threshold, the current image is discarded and the next item of the dataset corresponding to the highest $ZMLM^{Max}$ is considered

as candidate for pattern matching. The procedure ends all the blocks in the list have been processed.

6.4 Experimental Results

6.4.1 Experiments and performance evaluation

The performance evaluation the *COREL-1000-A Database*, [104] has been employed. It consists of 1000 JPEG colored images of size 384x256 or 256x384 pixels, split in 10 categories with 100 images each, see Figure 6.6. For each original image a second copy, rotated with respect to the original by a random angle has also been inserted in the data base resulting in a testset of 2000 images. This database has been chosen because it covers a wide range of semantic categories, from natural scenes to artificial objects.

In the performed simulations the following conditions and the following set of parameters have been employed:

- two levels quadtree decomposition with Zernike filter diameters respectively of 27 and 13 pixels;
- quadtree splitting threshold for the Euclidean distance between the original image and reconstructed image equal to 0.007, (see Equation 6.9);
- the maximum order of truncated Zernike expansion is equal to $n = 5$, this value has been chosen so that sampling step, given by the size of the filter, is sufficient to follow all the oscillations of the filter itself;
- the image is represented in YUV format and the retrieval procedure is only applied to the luminance component.

These setup values have been experimentally determined because they minimized the prediction errors.

Although the sequential matching should be applied to the whole set of quadtree blocks, the analysis of the performed simulations shows that a limited number of blocks (e.g., usually 5) is in general sufficient in order to achieve an high probability of finding the

correct image. This number is directly related to the threshold η of the stopping condition and specifically, it has been set to $\eta = 60\%$ in the reported simulations.

For the performance assessment, the average percentage of recovered relevant images, the average percentage of angle estimates error less than 5 degrees, and the mean square error of angle estimate have been computed. The results are summarized in Table 6.1.

Table 6.1: Performance results of the proposed method.

Average percentage of recovered relevant images	95%
Average percentage of angle estimate error < 5 deg	91.95%
Root Mean square error of Angle Estimate	0.34 deg

In order to further reduce the complexity, the template orientation has been estimated on the basis of the orientation of the line passing through the center of the first and the second block of the tree.

6.4.2 Comparison with other methods

The retrieval accuracy and efficiency of the proposed method have been compared with both conventional and most recent methods. Conventional methods are based on combination of the most commonly used image features: color and texture, (CT), [105]. In addition, conventional methods include global search and regional search [106] and the method employed in SIMPLIcity [107]. In global search (GS), for each query image, a feature vector is selected and compared with those of images from the database. In regional search (RS) the user selects only the desired block (the region of interest) from the query image, the system then performs a search comparing the feature vectors of corresponding blocks from the image database. SIMPLIcity uses a wavelet-based feature extraction method. Recent methods include a technique based on Genetic Algorithms (GA) and a relevance feedback based modified version, [105]. It is important to underline that respect to the majority of existing image retrieval systems where a retrieved image is considered a match if it belongs to the same category of the query image, the proposed technique allows to retrieve the exact copy of the query image and its rotated version. In addition

it allows to estimate the angle with which the image has been rotated. The comparison results, reported in Table 6.2, show that the accuracy of the proposed method outperforms both conventional and state of the art methods. The proposed approach, tested by using a non-optimized MATLAB code, is time consuming with respect to the others methods. Nevertheless this value has been achieved with a rather small order with respect to Zernike expansion and with the simplified rotation estimation.

Table 6.2: Average retrieval precision of the proposed ZM method compared with conventional (global and regional search, color-texture, and SIMPLicity) and recent methods (Genetic Algorithm GA). The simulations have been tested on *COREL-1000-A Database*.

	ZM	GS	RS	Color-Texture	SIMPLicity	GA
Average retrieval precision (%)	95	87.2	84.8	43	46	53

6.4.3 Computational complexity

To assess the computational complexity of the method an evaluation of the number of elementary arithmetic operations has been performed. More in detail, let:

- R_{f_i}, C_{f_i} be the filter size;
- R, C be the image size;
- i be the number of employed quadtree levels;
- L be the number of filters that depends on the order N and the repetition M ;
- k be the number of the blocks considered.

The computation complexity of the image analysis phase is:

1. Zernike moments computation (repeated i times)
 - additions: $L \cdot R_{f_i} \cdot C_{f_i} \cdot R \cdot C$,
 - multiplications: $L \cdot R_{f_i} \cdot C_{f_i} \cdot R \cdot C$,

2. Reconstruction of the image by using the Zernike moments (repeated i times)

- additions: $L \cdot R_{f_i} \cdot C_{f_i} \cdot R \cdot C + L \cdot R \cdot C$,
- multiplications: $L \cdot R_{f_i} \cdot C_{f_i} \cdot R \cdot C$,

3. Reconstruction error computation (repeated i times)

- additions: $2 \cdot R_{f_i} \cdot C_{f_i}$.

Summarizing the approximated number of additions is: $2 \cdot i \cdot (L \cdot R \cdot C \cdot R_{f_i} \cdot C_{f_i})$ and the approximated number of multiplications is: $2 \cdot i \cdot (L \cdot R \cdot C \cdot R_{f_i} \cdot C_{f_i})$. The computational complexity of the comparison between the query and the generic image in the DB is shown in Table III. Finally, the total number of additions is:

$$\begin{aligned}
 & 2 \cdot R \cdot C + L \cdot (R \cdot C + 1) + 200 \cdot L + 2 + (k - 1) \cdot \left[7 + L \cdot \left(\frac{R \cdot C + 1}{8} \right) + 200 \cdot L + 2 \right] = \\
 & = R \cdot C \cdot (2 + L) + 201 \cdot L + 2 + (k - 1) \cdot \left\{ 9 + L \cdot \left[\left(\frac{R \cdot C + 1}{8} \right) + 200 \right] \right\} \cong \\
 & \cong R \cdot C \cdot L + (k - 1) \cdot \left(L \cdot \frac{R \cdot C}{8} \right),
 \end{aligned} \tag{6.26}$$

and the total number of multiplications is:

$$\begin{aligned}
 & 2 \cdot R \cdot C + L \cdot (R \cdot C + 4) + 1 + 300 \cdot L + 2 + (k - 1) \cdot \left[5 + L \cdot \left(\frac{R \cdot C + 4}{8} \right) + 1 + 300 \cdot L + 2 \right] = \\
 & = R \cdot C \cdot (2 + L) + 304 \cdot L + 3 + (k - 1) \cdot \left\{ 8 + L \cdot \left[\left(\frac{R \cdot C + 4}{8} \right) + 300 \right] \right\} \cong \\
 & \cong R \cdot C \cdot L + (k - 1) \cdot \left(L \cdot \frac{R \cdot C}{8} \right).
 \end{aligned} \tag{6.27}$$

The proposed approach, tested by using a non-optimized MATLAB code on a 2,67GHz CPU and 2GB RAM desktop, is time consuming with respect to the others methods. However, the use of optimized compiled C code for the core algorithm blocks (the Zernike decomposition and the Maximum likelihood estimation) allows to reduce the computational time of a factor of ten.



(a)



(b)



(c)



(d)



(e)



(f)



(g)



(h)



(i)



(j)

Figure 6.6: Examples of images for each categories present in the database *COREL-1000-A*.

Table 6.3: Number of the additions and multiplications performed during the comparison between the query image and the image in the DB.

	Additions	Multiplications
Computation of the energy of the image	$2 \cdot R \cdot C$	$2 \cdot R \cdot C$
Maximum Likelihood estimation	$L \cdot (R \cdot C + 1)$	$L \cdot (R \cdot C + 4) + 1$
Angle estimation	$200 \cdot L$	$300 \cdot L$
Update energy of the image	2	2
Estimation of the Region of Interest (repeated $k - 1$ times)	7	5
Maximum Likelihood estimation on Region of Interest (repeated $k - 1$ times)	$L \cdot \left(\frac{R \cdot C + 1}{8}\right)$	$L \cdot \left(\frac{R \cdot C + 4}{8}\right) + 1$
Angle estimation (repeated $k - 1$ times)	$200 \cdot L$	$300 \cdot L$
Update energy of the image (repeated $k - 1$ times)	2	2

6.5 The proposed method in Laguerre-Gauss domain

6.5.1 Laguerre-Gauss Transform

Let $\mathbf{x} = [x_1, x_2]$ be the coordinates in the real plane \mathbb{R}^2 . Any image $f(\mathbf{x}) \in L^2(\mathbb{R}^2, d^2\mathbf{x})$ can be expanded on orthogonal basis under a Gaussian weighting function, $w(\mathbf{x}) = e^{-\pi|\mathbf{x}|^2}$, complete over the entire plane, around a given point $\xi = (\xi_1, \xi_2)$, [94]:

$$f(\mathbf{x}) w\left(\frac{\mathbf{x} - \xi}{s}\right) = \sum_{n=0}^{\infty} \sum_{k=0}^{\infty} D_{n,k}(\xi) \frac{1}{s} \mathcal{L}_k^{(n)}\left(\frac{|\mathbf{x} - \xi|}{s}, \theta(\mathbf{x} - \xi)\right) \quad (6.28)$$

where $\theta()$ denotes the angular coordinate defined by the relationship

$$\theta(\mathbf{x}) = tg^{-1}\left(\frac{x_2}{x_1}\right),$$

$\mathcal{L}_k^{(n)}(r, \theta)$ are the *Laguerre-Gauss* functions defined as:

$$\mathcal{L}_k^{(n)}(r, \theta) = (-1)^k 2^{(|n|+1)/2} \pi^{|n|/2} \left[\frac{k!}{(|n|+k)!} \right]^{1/2} \cdot r^{|n|} L_k^{(n)}(2\pi r^2) e^{-\pi r^2} e^{jn\theta}, \quad (6.29)$$

$L_k^{(n)}$ are generalized Laguerre polynomials defined by Rodriguez's formula:

$$L_k^{(n)}(t) = \frac{t^{-n} e^t}{k!} \frac{d^k}{dt^k} [t^{k+n} e^{-t}] = \sum_{h=0}^k (-1)^h \binom{n+k}{k-h} \frac{t^h}{h!}, \quad (6.30)$$

and the expansion coefficients are defined as:

$$D_{n,k}(\xi) = \left\langle f(\mathbf{x}) w\left(\frac{\mathbf{x} - \xi}{s}\right), \frac{1}{s} \mathcal{L}_k^{(n)}\left(\frac{|\mathbf{x} - \xi|}{s}, \theta(\mathbf{x} - \xi)\right) \right\rangle.$$

We incidentally observe that the parameter s controls the width of the weighting function.

The expansion in terms of Gauss-Laguerre functions can be derived, for instance, by first applying the Fourier's series expansion to the representation of the image f in polar coordinates with respect to the angular coordinate, and then expanding the radial profile of each harmonic using the Laguerre polynomials $L_k^{(n)}(t)$.

The Gauss-Laguerre functions are members of the wider class of Circular Harmonic Functions (CHFs), successfully used for many low level vision tasks, thanks to their selectivity with respect to basic visual patterns, [108],[109]. CHFs of n -th order are, by definition, polar separable functions of the form $h(r)e^{jn\theta}$.

By virtue of their harmonic angular shape, CHF's are indeed natural detectors for different classes of features: CHF's of order $n=1$ for example are tuned to edges, $n = 2$ to lines, $n = 3$ to forks, etc..

In addition, every Laguerre-Gauss function generates a dyadic Circular Harmonic Wavelet. This means that every image $f(\mathbf{x})$ can be represented by its continuous wavelet transform $W_{\mathcal{L}_k^n}[f](\mathbf{b}, \alpha, \sigma)$ where \mathbf{b} , α and σ are the parameters representing respectively the translated, rotated and scaled version of the mother wavelet $\mathcal{L}_k^{(n)}$.

With reference to localization of complicated patterns, a rather relevant property is the following.

PROPERTY I. *Given an image f defined over a finite support $I \subset \mathbb{R}^2$ and a lattice $\Xi = \{\xi_m \in I, m = 1, \dots, M\}$ the set of Laguerre-Gauss functions*

$$\left\{ \frac{1}{s} \mathcal{L}_k^{(n)} \left(\frac{|\mathbf{x} - \xi_m|}{s}, \theta(\mathbf{x} - \xi_m) \right), m = 1, \dots, M \right\}$$

defines a Riesz basis for f .

Proof. The orthogonality of the *Laguerre-Gauss* functions implies that

$$\sum_m \int_I \left| w \left(\frac{\mathbf{x} - \xi_k}{s} \right) \right|^2 |f(\mathbf{x})|^2 d\mathbf{x} = \sum_m \sum_n \sum_k |D_{n,k}(\xi_m)|^2,$$

therefore

$$\gamma \|f(\mathbf{x})\|^2 \leq \sum_m \sum_n \sum_k |D_{n,k}(\xi_m)|^2 \leq \Gamma \|f(\mathbf{x})\|^2,$$

with

$$\gamma = \min_{\mathbf{x}} \sum_{m=1}^M \left| w \left(\frac{\mathbf{x} - \xi_k}{s} \right) \right|^2,$$

and

$$\Gamma = \sum_{m=1}^M \int_I \left| w \left(\frac{\mathbf{x} - \xi_k}{s} \right) \right|^2 d\mathbf{x}.$$

q.e.d.

Thus in turn implies that the inner product between two images f and g with expansion coefficients $D_{n,k}(\xi_m)$ and $C_{n,k}(\xi_m)$, respectively, satisfies the following condition:

$$\frac{\gamma}{\Gamma} \langle f(\mathbf{x}), g(\mathbf{x}) \rangle \leq \frac{1}{\Gamma} \sum_m \sum_n \sum_k D_{n,k}(\xi_m) C_{n,k}^*(\xi_m) \leq \langle f(\mathbf{x}), g(\mathbf{x}) \rangle$$

The magnitude of the approximation error strictly depends on the ratio γ/Γ , that can be a priori computed. Moreover, it could be demonstrated that the more general set of Gauss-Laguerre functions

$$\left\{ \frac{1}{s_k} \mathcal{L}_k^{(n)} \left(\frac{|\mathbf{x} - \xi_m|}{s_k}, \theta(\mathbf{x} - \xi_m) \right), m = 1, \dots, M \right\}$$

it is a Riesz basis too.

Here the quadtree decomposition adaptive scheme for choosing the lattice Ξ and the shape parameters $\{s_k\}$ that realizes a good trade off between accuracy and complexity is applied, with a similar technique proposed for Zernike moments.

6.5.2 Maximum Likelihood Localization

Let now $f(\mathbf{x})$ be the observed region of interest that contains a noisy, translated, rotated and scaled copy of a given template pattern $g(\mathbf{x})$ so that we have:

$$w[R_\varphi(\mathbf{x} - \mathbf{b})]f(\mathbf{x}) = w[R_\varphi(\mathbf{x} - \mathbf{b})]g\left[R_\varphi\left(\frac{\mathbf{x} - \mathbf{b}}{a}\right)\right] + v(\mathbf{x}),$$

where the parameters a , \mathbf{b} and φ represent respectively scale, position and rotation of the observed image and R_φ is the rotation matrix defined as:

$$R_\varphi = \begin{bmatrix} \cos \varphi & \sin \varphi \\ -\sin \varphi & \cos \varphi \end{bmatrix}.$$

Let $\theta = [\mathbf{b}, a, \varphi]$ be the unknown parameter vector, the Likelihood functional is given by the conditional probability of f w.r.t. θ , divided by any arbitrary function that does not depend on θ :

$$\ln \Lambda[f(\mathbf{x}); \mathbf{b}, a, \varphi] = -\frac{2}{N_0} \int \int \left| w\left[R_\varphi\left(\frac{\mathbf{x} - \mathbf{b}}{a}\right)\right] \right|^2 \times \left| f(\mathbf{x}) - g\left[R_\varphi\left(\frac{\mathbf{x} - \mathbf{b}}{a}\right)\right] \right|^2 d\mathbf{x}. \quad (6.31)$$

Direct Maximum Likelihood solution evaluation is not very simple because the search of the maximum for \mathbf{b} , φ and a is a search in a four dimensional space. However the choice of LG functions as expansion basis and the choice of gaussian window which is rotation invariant leads to a simpler iterative procedure,[110]. In fact, considering that any n -th

order CHF can be steered in any direction φ by simple multiplication by the complex factor $e^{-jn\varphi}$, and denoting with $\eta_{n,k}(\mathbf{x}; a)$ the expansion coefficients of $g(\mathbf{x}/a)$, we can approximate the ML functional as follows:

$$\ln \Lambda[f(\mathbf{x}); \mathbf{b}, a, \varphi] \simeq \text{const} - \frac{2}{N_0 \Gamma} \sum_m \sum_n \sum_k \left| D_{n,k}(\xi_m) - \eta_{n,k} \left[R_\varphi \left(\frac{\xi_m - \mathbf{b}}{a} \right); a \right] e^{-jn\varphi} \right|^2. \quad (6.32)$$

On the other hand, denoting with $C_{n,k}(\mathbf{x})$ the expansion coefficients of $g(\mathbf{x})$ for $a = 1$ (i.e. $C_{n,k}(\mathbf{x}) = \eta_{n,k}(\mathbf{x}; 1)$) the following interscale relationship holds

$$\eta_{n,k}(\mathbf{x}; a) = \sum_{l=k}^{\infty} B(a; n, k, l) C_{n,k}(\mathbf{x}),$$

where

$$B(a; n, k, l) = (-1)^{l-k} \sqrt{\frac{(n+l)!!}{(n+k)!k!}} \cdot \frac{a^{-n-2k}}{(l-k)!} \left(1 - \frac{1}{a^2} \right)^{l-k}, \quad (l \geq k). \quad (6.33)$$

Thus, the ML functional can be further approximated as follows:

$$\ln \Lambda[f(\mathbf{x}); \mathbf{b}, a, \varphi] \simeq \text{const} - \frac{2}{N_0 \Gamma} \times \sum_m \sum_n \sum_k \left| D_{n,k}(\xi_m) - \sum_{l=k}^{\infty} B(a; n, k, l) C_{n,k} \left[R_\varphi \left(\frac{\xi_m - \mathbf{b}}{a} \right) \right] e^{-jn\varphi} \right|^2$$

The maxima of the expression of the above functional w.r.t. scale a and orientation φ represent the *Laguerre-Gauss Likelihood Map* (GLLM):

$$GLLM(\mathbf{b}) = \max_{a, \varphi} \{ \ln \Lambda[f(\mathbf{x}); a, \varphi, \mathbf{b}] \}$$

The local estimate of the maxima can be performed by means of quasi-Newton maximization procedure as the Broyden-Fletcher-Goldfarb-Shanno algorithm. The location of the absolute maximum of this map provides the estimated position of the pattern. The resulting *Laguerre-Gauss Likelihood Map* indicates, point by point, the best matches between the two images under all possible orientations and scales.

6.5.3 Quadtree Decomposition

Since $f(\mathbf{x})$ may contain multiple objects with arbitrary shape, direct use of Gauss Laguerre expansion as well of other CHF's expansions, as those in Zernike's moments, for computing the ML functional would require a larger and larger number of expansion terms. Thus, in order to reduce the computational complexity, we resort to the hypercomplete Riesz basis that allows to partition the region of interest into smaller squares, so that for each of them a truncated Laguerre-Gauss expansion with a reduced number of terms can be utilized. More in detail, let R represent the region of interest, eventually coincident with the whole image, and let P be a predicate equal to *True* whenever the accuracy of the approximation of the current Riesz basis can be considered satisfactory. R is partitioned into smaller and smaller square regions $R^{(i)}$, so that for each $R^{(i)}$, $P(R^{(i)}) = \text{True}$. Initially the basis set is empty and the current region $R^{(0)}$ is set equal to the given ROI. At the i -th step of the recursion, the center ξ_i of the current region $R^{(i)}$ is evaluated and the subset of functions

$$\left\{ \frac{1}{s_i} \mathcal{L}_k^{(n)} \left(\frac{|\mathbf{x} - \xi_i|}{s_i}, \theta(\mathbf{x} - \xi_i) \right), k = 1, \dots, K, n = 1, \dots, N \right\}$$

is added to the current basis set as a potential candidate. Then the predicate P is computed.

In order to control the computational complexity of the whole procedure, we chose as predicate P the comparison of the L^2 norm of the approximation error in the reconstruction of a square block of the image with a predefined number of Gauss Laguerre coefficients with a threshold t . If the norm of the error between the image itself $f(\mathbf{x})$ and the reconstructed image $\hat{f}(\mathbf{x})$ using the current basis exceeds a predefined threshold, P is set to false. Let us denote with δ_i the width of $R^{(i)}$, and with $w_T(\mathbf{x})$ a square window of unitary width, then

$$P(R^{(i)}) = \left\{ \left\| w_T \left(\frac{\mathbf{x} - \xi_i}{\delta_i} \right) [f(\mathbf{x}) - \hat{f}(\mathbf{x})] \right\|^2 < t \right\}.$$

Pattern location, rotation and scale estimation accuracy is strictly related to the Fisher's information. However, as demonstrated in [94] this quantity is proportional to the magnitude of the energy of the derivatives along two orthogonal directions and to

the energy of the angular derivative, or, equivalently, to the effective spatial and angular bandwidths.

In order to reduce the search time, the ranking of template quadtree blocks is based on saliency of key-points extracted for each block as suggested in [111].

When a ROI of a given image has to be searched in a database, the comparison is accomplished the first block of the ranked list and Laguerre-Gauss coefficients of each database candidate image. The expansion in Laguerre-Gauss domain is made using the same base employed for the current quadtree block.

Since rotation of a pattern simply produces a linear phase shift of each expansion coefficient proportional to the order of the angular harmonic, detection of the pattern belonging to the first square of the ranked quadtree list can be performed by means of a quasi-Newton maximization procedure as the Broyden-Fletcher-Goldfarb-Shanno algorithm maximizing, for each \mathbf{b} the quantity

$$GLLF^{(1)}(\mathbf{b}, a, \varphi) = -\frac{2}{N_0\Gamma} \times \sum_{n=0}^N \sum_{k=0}^K \left| D_{n,k}(\xi_c) - \sum_{l=0}^L B(a; n, k, l) C_{n,k}(\xi_c - \mathbf{b}) e^{-jn\varphi} \right|^2 \quad (6.34)$$

where ξ_c denotes the center of the current region.

Thus, for each discrete location of a grid, the rotation and the scale maximizes the $GLLF^{(1)}$ functional are determined and then a discrete direct search is performed to determine its absolute maximum. Thus, at the first step the parameter estimate is

$$[\hat{\mathbf{b}}^{(1)}, \hat{a}^{(1)}, \hat{\varphi}^{(1)}] = Arg \left\{ \max_{\mathbf{b}, a, \varphi} [GLLF^{(1)}(\mathbf{b}, a, \varphi)] \right\}$$

Once for each image of the dataset the local maximum of $GLLF^{(1)}$ has been computed, the images are ranked on the basis of this absolute maximum. Then the image corresponding to the highest $GLLF^{(1)}$ is selected as the potential candidate for image matching, and $[\hat{\mathbf{b}}^{(1)}, \hat{a}^{(1)}, \hat{\varphi}^{(1)}]$ is employed as coarse estimate in order to verify whether the candidate image contains the second block of the rank ordered list of quadtree elements, too.

With respect to the first block, the $GLLF^{(2)}$ map is built only for a limited set of possible locations, falling inside a small neighbor of the site predicted on the basis of the coarse estimates. In addition, the quasi-Newton procedure utilized to maximize $GLLF^{(2)}$ is initialized using the coarse estimate too.

If the energy of the difference between the subset of the reference template, constituted by the first and the second square of the quadtree and the current image falls below a predefined threshold, location and rotation of the image are refined and the next square analyzed.

In general, at the h -th stage the $GLLF^{(h)}$ map is computed using the first h points of the lattice, ranked according to the saliency indicator, i.e.,

$$GLLF^{(h)}[\mathbf{b}, a, \varphi] = -\frac{2}{N_0\Gamma} \times \sum_m \sum_n \sum_k \left| D_{n,k}(\xi_m) - \sum_{l=k}^{\infty} B(a; n, k, l) C_{n,k} \left[R_{\varphi} \left(\frac{\xi_m - \mathbf{b}}{a} \right) \right] e^{-jn\varphi} \right|^2$$

The procedure ends when the last block in the list has been processed.

6.6 Experimental Results

The proposed method has been tested on a 52 images database. For each grey-level image, we have 4 different views (256x256 pixels) corresponding to different orientations and scales, the first one is the original image, the second image is scaled, the third one is rotated and the last one is scaled and rotated. See fig.6.7. In the performed simulations, the Laguerre-Gauss expansion has been truncated to the (angular) order $n=6$, and to the (radial) order $k=7$. This gives, for each quadtree block a descriptor array of 173 elements. The value s_m of the weighting gaussian window is matched to quadtree block size. In fig.(6.8) an example of the Likelihood map for the "Einstein" image is showed. In table (6.4) some results on angle and scale estimate error for some images from the multimedia database are reported. The angle and scale estimate errors are quite low and the algorithm is capable to find the searched points in the candidate image, estimating rotation and scale of the image with a low error rate.





















Image	Angle Estimate Error [deg]	Scale Estimate Error
 airplane	0	0
 airplane-rot10-scal90	0.19	0.01
 airplane-rot15	0.58	0.01
 airplane-scal92	0.07	0.02
 einstein	0	0
 einstein-rot40-scal97	2.58	0.03
 einstein-rot220	3.13	0
 einstein-scal97	0	0.03
 tree	0	0
 tree-rot195-scal87	0.38	0.01
 tree-rot35	0.09	0.01
 tree-scal80	0.11	0.01
 peppers	0	0
 peppers-rot78-scal95	5.81	0.07
 peppers-rot98	1.65	0.01
 peppers-scal82	0.78	0.07
 clock	0	0
 clock-rot90-scal92	0.05	0.01
 clock-rot85	0.22	0.08
 clock-scal98	135 0.59	0.01

Table 6.4: Angle and scale estimate error for some images from the database



Figure 6.7: Three database samples with different orientations

6.7 Conclusions

Content discovery, delivery, and streaming are the basic functionalities of current content-centric Internet services. The first generation of retrieval systems was based on the use of metadata describing the semantic content of a multimedia document, usually extracted by manual procedures. However, Future Internet content aware services will require more efficient functionalities for inspection, crawling, recognition, categorization, and indexing of digital content with minimal human intervention. It is very important to employ fast and reliable algorithms for locating and tracking complex objects irrespective of their actual orientation and scale.

At this aim, two novel techniques for searching complex patterns in large databases based on the decomposition of the template in smaller blocks whose size is adapted to the local image content have been proposed and analyzed through this chapter. The represen-

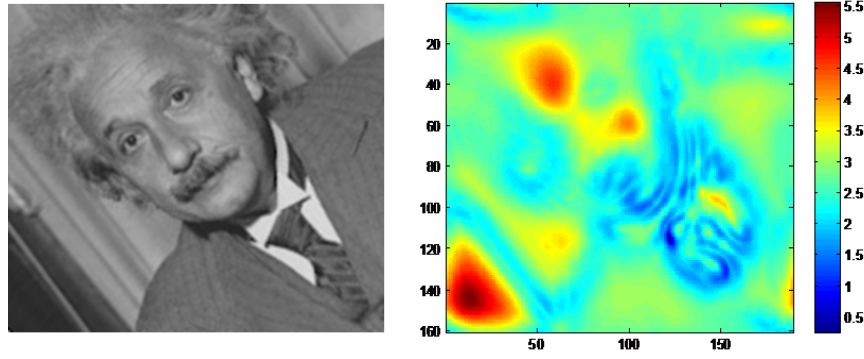


Figure 6.8: Laguerre-Gauss Likelihood map of "Einstein" image

tation of the image blocks by means of Zernike moments and Laguerre-Gauss transform allows the design of an effective maximum likelihood matching procedure. With respect to state of the art methods that represent the whole pattern in terms of orthogonal basis and eventually extract an invariant feature vector from the representation coefficients, the use of the quadtree decomposition keeps low the number of terms of the truncated expansions still assuring the required accuracy in representing the original image and to efficiently manage even template supports with irregular shape. The experimental results show a very good detection rate (about 95%) and an high accuracy in estimating the template rotation.

The Laguerre-Gauss Transform allows a simpler iterative Likelihood functional estimate compared to the traditional Maximum Likelihood based on searching the image with the whole set of rotated and scaled images. In particular, it allows an approximated ML solution with a low computational cost. Thanks to the Gaussian windowing, this method is well suited for localization of patterns of complex objects. The experimental results show an high detection rate and an accurate location estimate and show how this class of Circular Harmonic filters performs very well in presence of scaling and rotation.

Respect to the state of art where the majority of image recognition techniques are capable to detect the image class at which the given image belongs and not the image itself or its scaled and rotated version, with these new methodologies it is possible also to identify the image itself in a large database. Although in the proposed methods, the ML

functional has been thresholded to detect the template presence in a collection of images, it can be directly employed to rank the collection images with respect to the similarity of their content to the given template.

Appendix A

Maximum Likelihood Estimation in Zernike domain

Direct Maximum Likelihood estimation is of difficult solution because the search of maximum of the rotation φ and the location \mathbf{b} implies the search in three dimensions. However the choice of a disk of radius σ as weighting window $w(\mathbf{x})$ and the use of Zernike expansion lead to a simpler and faster iterative procedure. In fact, expanding both $w[R_\varphi(\mathbf{x} - \mathbf{b})]f(\mathbf{x})$ and $w(\mathbf{x})g(\mathbf{x})$ by means of Zernike moments we obtain:

$$w[R_\varphi(\mathbf{x} - \mathbf{b})]f(\mathbf{x}) = \sum_{n=0}^{\infty} \sum_{m=-\infty}^{+\infty} Z_{nm}^f(\mathbf{x}_0) \frac{1}{\sigma} V_{nm} \left(\frac{\mathbf{x} - \mathbf{x}_0}{\sigma} \right), \quad (\text{A.1})$$

$$w(\mathbf{x})g(\mathbf{x}) = \sum_{n=0}^{\infty} \sum_{m=-\infty}^{+\infty} Z_{nm}^g(\mathbf{x}_0) \frac{1}{\sigma} V_{nm} \left(\frac{\mathbf{x}}{\sigma} \right), \quad (\text{A.2})$$

where

$$Z_{nm}^f(\mathbf{b}) = \frac{n+1}{\pi} \int \int_{\|\mathbf{x}-\mathbf{b}\| \leq \sigma} f(\mathbf{x}) \frac{1}{\sigma} V_{nm}^* \left(\frac{\mathbf{x} - \mathbf{b}}{\sigma} \right) dx_1 dx_2, \quad (\text{A.3})$$

$$Z_{nm}^g = \frac{n+1}{\pi} \int \int_{\|\mathbf{x}-\mathbf{b}\| \leq \sigma} g(\mathbf{x}) \frac{1}{\sigma} V_{nm}^* \left(\frac{\mathbf{x}}{\sigma} \right) dx_1 dx_2. \quad (\text{A.4})$$

Substituting (A.3) and (A.4) in (6.17) and considering the orthogonality of Zernike polynomials, we have:

$$\begin{aligned}
 \ln \Lambda[f(\mathbf{x}); \mathbf{b}, \varphi] &= -\frac{2}{N_0} \int \int_{\|\mathbf{x}-\mathbf{b}\| \leq \sigma} |f(\mathbf{x}) - g[R_\varphi(\mathbf{x} - \mathbf{b})]|^2 dx_1 dx_2 \\
 &= -\frac{2}{N_0} \sum_{n=0}^{\infty} \sum_{m=-\infty}^{+\infty} \frac{\pi}{(n+1)\sigma^2} \left| Z_{nm}^f(\mathbf{b}) - Z_{nm}^g e^{jm\varphi} \right|^2 \\
 &= -\frac{2}{N_0} \sum_{n=0}^{\infty} \sum_{m=-\infty}^{\infty} \frac{\pi}{(n+1)\sigma^2} \left[\left| Z_{nm}^f(\mathbf{b}) \right|^2 + \left| Z_{nm}^g \right|^2 \right] + \\
 &\quad + \frac{4}{N_0} \sum_{m=-\infty}^{\infty} \frac{\pi}{(n+1)\sigma^2} \operatorname{Re} \left\{ \sum_{n=0}^{\infty} [Z_{nm}^f(\mathbf{b})]^* Z_{nm}^g e^{jm\varphi} \right\}. \quad (\text{A.5})
 \end{aligned}$$

Let us denote the maxima of the truncated version of the above expression w.r.t. rotation versus the pattern location \mathbf{b} as the *Zernike Moments Likelihood Map*:

$$ZMLM(\mathbf{b}) = \max_{\varphi} \left\{ -\frac{2}{N_0} \sum_{n=0}^N \sum_{m=-M}^{+M} \frac{\pi}{(n+1)\sigma^2} \left| Z_{nm}^f(\mathbf{b}) - Z_{nm}^g e^{jm\varphi} \right|^2 \right\}. \quad (\text{A.6})$$

This map indicates, point by point, the best matching between the region of interest and the reference image under all possible orientations.

In order to derive a fast estimator of $\varphi(\mathbf{b})$ we observe that, by posing

$$\mu_m(\mathbf{b}) e^{j\eta_m(\mathbf{b})} = \frac{1}{(n+1)} \sum_{n=0}^N [Z_{nm}^f(\mathbf{b})]^* Z_{nm}^g, \quad (\text{A.7})$$

we can write

$$\begin{aligned}
 ZMLM(\mathbf{b}) &= -\frac{2}{N_0} \sum_{n=0}^{\infty} \sum_{m=-\infty}^{\infty} \frac{\pi}{(n+1)\sigma^2} \left[\left| Z_{nm}^f(\mathbf{b}) \right|^2 + \left| Z_{nm}^g \right|^2 \right] + \\
 &\quad \max_{\varphi} \left\{ \frac{4\pi}{N_0\sigma^2} \sum_{m=-M}^M \mu_m(\mathbf{b}) \cos[m\varphi + \eta_m(\mathbf{b})] \right\}. \quad (\text{A.8})
 \end{aligned}$$

For each position, the evaluation of the rotation $\hat{\varphi}(\mathbf{b})$ can be performed by Newton-Raphson iterative procedure that in the actual case specifies as follows:

$$\hat{\phi}_{ML}^{(h)}(\mathbf{b}) = \hat{\phi}_{ML}^{(h-1)}(\mathbf{b}) - \frac{\sum_{m=-M}^M m \mu_m(\mathbf{b}) \sin \left[m \hat{\phi}_{ML}^{(h-1)}(\mathbf{b}) + \eta_m(\mathbf{b}) \right]}{\sum_{m=-M}^M m^2 \mu_m(\mathbf{b}) \cos \left[m \hat{\phi}_{ML}^{(h-1)}(\mathbf{b}) + \eta_m(\mathbf{b}) \right]}. \quad (\text{A.9})$$

Bibliography

- [1] M. Ouaret, F. Dufuax, and T. Ebrahimi. Multi-view distributed video coding with encoder driven fusion. In *European Signal Processing Conference (EUSIPCO-2007)*, September 2007.
- [2] D. Slepian and J.K. Wolf. Noiseless coding of correlated information sources. *IEEE Transactions on Information Theory*, 19:471–480, July 1973.
- [3] J. Venkaratam. Analisis of slepian wolf coding. Technical report, University of Notredame, February 2003.
- [4] A.D. Wyner and J. Ziv. The rate-distortion function for source coding with side information at the decoder. *IEEE Transactions on Information Theory*, 22(1):1–10, January 1976.
- [5] R. Zamir and S. Shamai. Nested linear/lattice codes for wyner-ziv encoding. In *Proceeding of Information Theory Workshop*, pages 92–93, June 1998.
- [6] S.S. Pradhan and K. Ramachandran. Distributed source coding using syndromes (discus): Design and construction. *Proceeding of IEEE Data Compression Conference*, pages 158–167, March 1999.
- [7] A. Liveris, Z. Xiong, and C. Georghiades. Compression of binary sources with side information a he decoder using ldpc codes. *IEEE Communications Letters*, 6(10):440–442, October 2002.

- [8] C. Yeo and K. Ramchandran. Robust distributed multi-view video compression for wireless camera networks. In *Proceeding of SPIE Visual Communications and Image Processing*, January 2007.
- [9] J.D. Areia, J. Ascenso, C. Brites, and F. Pereira. Wyner-ziv stereo video coding using a side information fusion approach. In *IEEE 9th Workshop on Multimedia Signal Processing, MMSP 2007*, pages 453–456, October 2007.
- [10] M. Tagliasacchi, G. Prandi, and S. Tubaro. Symmetric distributed coding of stereo video sequences. In *IEEE International Conference on Image Processing, ICIP 2007*, volume 2, October 2007.
- [11] B. Girod, A.M. Aaron, S. Rane, and D. Rebollo-Monedero. Distributed video coding. In *Proceedings of the IEEE*, volume 93, pages 71 –83, 2005.
- [12] Mourad Ouaret. *Selected Topics on Distributed Video Coding*. PhD thesis, Ecole Polytechnique Federale de Lausanne, 2009.
- [13] J. C. Moreira and P. G. Farrell. *Essentials of Error-Control Coding*. J. Wiley and Sons, Ltd, 2006.
- [14] M. Naccari, M. Tagliasacchi, S. Tubaro, P. Zontone, R. Rinaldo, and R. Bernardini. Forward error protection for robust video streaming based on distributed video coding principles. In *Proceeding of 5th International Conference on Visual Information Engineering*, pages 747–752, August 2008.
- [15] F. Pereira, P. Correia, J. Ascenso, E. Acosta, L.Torres, C. Guillemot, C. Bandeira, M. Ouaret, F. Dufaux, T. Ebrahimi, R. Leonardi, M. Dalai, and S. Klomp. Application scenarios and functionalities for dvc, final version. Deliverable d19, DISCOVER, 2007.
- [16] "<http://en.wikipedia.org/wiki/videoconferencing>".
- [17] A. Gotchev, S. Jumisko-Pyyko, A. Boev, and D. Strohmeier. Mobile 3dtv system: quality and user perspective. In *4th Int. Mobile Multimedia Communications Conf. MobiMedia*, 2008.

- [18] A. Boev, D. Hollosi, A. Gotchev, and K. Egiazarian. Classification and simulation of stereoscopic artifacts in mobile 3dtv content. In *Proceeding of SPIE, Stereoscopic Displays and Applications XX*, Feb. 2009.
- [19] L. Meesters, W. Ijsselsteijn, and P. Seuntjens. Survey of perceptual quality issues in three-dimensional television system. In *Proceeding of SPIE, Stereoscopic Displays and Virtual Reality Systems X*, volume 5006, 2003.
- [20] A. Smolic, K. Mueller, N. Stefanoski, J. Ostermann, A. Gotchev, G.B. Akar, G. Triantafyllidis, and A. Koz. Coding algorithms for 3dtva survey. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(11):1606–1621, November 2007.
- [21] M. Cancellaro, V. Palma, and A. Neri. Stereo video artifacts introduced by a distributed coding approach. In *Proceeding of International Workshop on Video Processing and Quality Metrics (VPQM 2010)*, Scottsdale, Arizona, January 2010.
- [22] A. Boev, D. Hollosi, and A. Gotchev. Classification of stereoscopic artefacts. Technical report, MOBILE3DTV, April 2010.
- [23] M. Yuen. *Coding artifacts and visual distortion*. H.R. Wu and K.R. Rao, Digital Video Image Quality and Perceptual Coding, CRC Press, 2005.
- [24] C.T.E.R. Hewage, S.T. Worrall, S. Dogan, and A.M. Kondoz. Prediction of stereoscopic video quality using objective quality models of 2-d video. *Electronics Letters*, 44(16):963–965, 31 2008.
- [25] Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.
- [26] "<http://en.wikipedia.org/wiki/peaksignaltonoise>".
- [27] A. Loza, L. Mihaylova, D. R. Bull, and C. N. Canagarajah. Structural similarity-based object tracking in video sequences. In *Proceeding of International Conference on Information Fusion*, pages 1–6, Florence, Italy, July 2006.

- [28] M. Pinson and S. Wolf. A new standardized method for objectively measuring video quality. *IEEE Transaction on Broadcasting*, 50(3):312–322, September 2004.
- [29] ITU-T Recommendations J.143. *User requirements for perceptual video quality measurements in digital cable television*. Recommendations of the ITU, Telecommunication Standardization Sector.
- [30] ITU-R BT.500-11. Methodology for the subjective assessment of the quality of television pictures. In *Recommendations of the International Telecommunications Union - Radiocommunication Sector*, 2002.
- [31] Feng Xiao. Dct-based video quality evaluation. In *MSU Graphics and Media Lab (Video Group)*, Winter 2000.
- [32] A.B. Watson. Towards a perceptual videoquality metric. In *Human Vision, Visual Processing, and Digital Display VIII*, volume 3299, pages 139–147, 1998.
- [33] available at <http://tev.fbk.eu/databases/>.
- [34] T. Cover and J. Thomas. *Elements of Information Theory, Second Edition*. John Wiley & Sons, Inc, 2005.
- [35] J. Kurose and K. Ross. *Computer Networking: A Top-down Approach Featuring the Internet*. Pearson Addison Wesley, 2003.
- [36] O. Y. Bursalioglu, M. Fresia, G. Caire, and H. V. Poor. Lossy joint source-channel coding using raptor codes. *International Journal of Digital Multimedia Broadcasting*, 2008, 2008.
- [37] M. Fresia and G. Caire. A practical approach to lossy joint source-channel coding. In *Proceedings of Information Theory and Applications Workshop*, San Diego, California, USA, February 2007.
- [38] Gastpar M., Rimold B., and M. Vetterli. To code, or not to code: lossy source-channel communication revisited. *Information Theory, IEEE Transactions on*, 49(5):1147 – 1158, May 2003.

- [39] S. Shamai (Shitz) and S. Verdú. Capacity of channels with uncoded side information. *European Transactions on Telecommunications and Related Technologies (ETT)*, 6(5):587–600, September-October 1995.
- [40] S. Shamai (Shitz), S. Verdú, and R. Zamir. Systematic lossy source-channel coding. *IEEE Transaction on Information Theory*, 44:564–579, March 1998.
- [41] Y. Zhang, C. Zhu, and K.-H. Yap. A joint source-channel video coding scheme based on distributed source coding. In *Multimedia, IEEE Transactions on*, volume 10, December 2008.
- [42] Q. Xu, V. Stankovic, and Z. Xiong. Distributed joint source-channel coding of video using raptor codes. In *Selected areas in communications, IEEE Journal on*, volume 25, 2007.
- [43] A.D. Liveris, Z. Xiong, and C.N. Georgiades. Joint source-channel coding of binary sources with side information at the decoder using ira codes. *Multimedia Signal Processing, 2002 IEEE Workshop on*, pages 53 – 56, 2002.
- [44] X. Zhu, Y. Liu, and L. Zhang. Distributed joint source-channel coding in wireless sensor networks. In *Sensors*, number 6, pages 4901–4917, 2009.
- [45] P. Mitran and J. Bajcsy. Turbo source coding: A noise-robust approach to data compression. *Data Compression Conference*, 0:465, 2002.
- [46] H. Jin, A. Khandekar, and R. McEliece. Irregular repeataccumulate codes. *2nd International Symposium on Turbo Codes and related topics*, pages 1–8, 2000.
- [47] A. Shokrollahi. Raptor codes. *IEEE Transaction on Information Theory*, 52(6):2551–2567, June 2006.
- [48] Q. Xu, V. Stankovic, A. Liveris, and Z. Xiong. Distributed joint source-channel coding of video. In *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, volume 2, pages II – 674–7, September 2005.

- [49] M. Grangetto, E. Magli, and G. Olmo. Distributed joint source-channel arithmetic coding. In *Image Processing (ICIP), 2010 17th IEEE International Conference on*, pages 3717 –3720, September 2010.
- [50] J. Garcia-Frias and Y. Zhao. Data compression of unknown single and correlated binary sources using punctured turbo codes. In *Proceeding of the 39th Annual Allerton Conference on Communication, Control, and Computing*, Allerton, Illinois, October 2001.
- [51] M. Sartipi and F. Fekri. Distributed source coding using short to moderate length rate-compatible ldpc codes: the entire slepian-wolf rate region. *IEEE Transactions on Communications*, 56(3):400–411, March 2008.
- [52] J. Garcia-Frias, Y. Zhao, and W. Zhong. Turbo-like codes for transmission of correlated sources over noisy channels. *IEEE Signal Processing Magazine*, 24(5):58–66, September 2007.
- [53] V. Palma, M. Cancellaro, and A. Neri. Joint distributed source-channel coding for 3d video. In *Proceeding of SPIE International Conference on Electronic Imaging 2011*, San Francisco, California, USA, 2011.
- [54] J. Hagenauer and N. Gortz. The turbo principle in joint source-channel coding. In *Information Theory Workshop, Proceedings IEEE of*, pages 275 – 278, 2003.
- [55] C. Berrou, A. Glavieux, and P. Thitimajshima. Near Shannon limit error-correcting coding and decoding: Turbo codes. *Proceedings IEEE International Conference on Communications*, pages 1064–1070, May 1993.
- [56] A. Neri, D. Blasi, L. Gizzi, and P. Campisi. Joint security and channel coding for OFDM communications. *16th European Signal Processing Conference EUSIPCO*, August 2008.
- [57] Xun Guo, Yan Lu, Feng Wu, and Wen Gao. Distributed video coding using wavelet. *IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 5427–5430, 2006.

- [58] X. Artigas, E. Angeli, and L. Torres. Side information generation for multi-view distributed video coding using a fusion approach. In *7th Nordic Signal Processing Symposium (NORSIG)*, 2006.
- [59] X. Guo and Y. Lu. Wyner-ziv based multiview video coding. *IEEE Transaction on Circuits and System for Video Technology*, 18(6):713–724, June 2008.
- [60] T. Maugey, W. Miled, M. Cagnazzo, and B. Pesquet-Popescu. Fusion schemes for multiview distributed video coding. In *17th European Signal Processing Conference, EUSIPCO*, 2009.
- [61] M.T. Pourazad, P. Nasiopoulos, and R.K. Ward. A new prediction structure for multi-view video coding. *International conference on digital signal processing*, July 2009.
- [62] X. Artigas, F. Tarres, and L. Torres. A comparison of different side information generation methods for multiview distributed video coding. In *International Conference on Signal Processing and Multimedia Applications*, Spain, 2007.
- [63] "<http://www.jpeg.org/jbig/index.html>".
- [64] M. Ouaret, F. Dufuax, and T. Ebrahimi. Recent advances in multi-view distributed video coding. In *SPIE Mobile Multimedia/Image Processing for Military and Security Applications*, April 2007.
- [65] E. Martinian, A. Behrens, J. Xin, and A. Vetro. View synthesis for multiview video compression. In *Picture Coding Symposium*, Beijing, China, April 2006.
- [66] V. Palma, M. Carli, and A. Neri. 3d scene reconstruction based on multiview distributed video coding in the zernike domain for mobile applications. In *Proceeding SPIE International Conference on Electronic Imaging 2011, Multimedia on Mobile Devices*, San Francisco, California, USA, January 2011.
- [67] R. Mukundan and K.R. Ramakrishnan. *Moment Function in Image Analysis: Theory and Applications*. World Scientific, 1998.

- [68] available at <http://research.microsoft.com/en-us/um/people/sbkang/3dvideodownload/>.
- [69] C. Liu and J. Kaiser. A survey of mobile ad hoc network routing protocols. Report Series 2003-08, University of Ulm. Tech, 2005.
- [70] M. Luby. Lt codes. *Proceedings of The 43rd Annual IEEE Symposium on Foundations of Computer Science*, pages 271–282, 2002.
- [71] R. Ahlswede, N. Cai, S. R. Li, and R. W. Yeung. Network information flow. *IEEE Transactions on Information Theory*, 46(4):1204–1216, July 2000.
- [72] E. Mammi, V. Palma, M. Carli, and A. Neri. Fountain code based al-fec for multicast services in manets. In *Wireless Telecommunication Symposium, (under submission)*, New York, USA, April 2011.
- [73] O. S. Badarneh and M. Kadoch. Multicast routing protocols in mobile ad hoc networks: A comparative survey and taxonomy. *EURASIP Journ. On Wireless Telecommunications and Networking*, June 2009.
- [74] M. Carli A. Neri K. Egiazarian E. Mammi, F. Battisti. A novel spatial data hiding scheme based on generalized fibonacci sequences. In *Proceedings of IS&T/SPIE Electronic Imaging Science and Technology*, 2008.
- [75] S. Corson and J. Macker. Mobile ad hoc networking (manet): Routing protocol performance issues and evaluation considerations. Technical report, IETF, October 1998.
- [76] T. Camp, J. Boleng, and V. Davies. A survey of mobility models for ad hoc network research. *Wireless Communications and Mobile Computing*, 2(5):483–502, September 2002.
- [77] T. Clausen. Comparative study of routing protocols for mobile ad-hoc networks. Technical report, INRIA, March 2004.
- [78] R. Vaishampayan and J. J. Garcia-Luna-Aceves. Efficient and robust multicast routing in mobile ad hoc networks. In *IEEE International Conference on Mobile Ad-hoc and Sensor System*, October 2004.

- [79] David J.C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, March 2005.
- [80] E. Hyytia, T. Tirronen, and J. Virtamo. Optimizing the degree distribution of lt codes with an importance sampling approach. In *RESIM 2006, 6th International Workshop on Rare Event Simulation*, 2006.
- [81] P. Cataldi, M.P. Shatarski, M. Grangetto, and E. Magli. Implementation and performance evaluation of lt and raptor codes for multimedia applications. In *Intelligent Information Hiding and Multimedia Signal Processing*, pages 263–266, December 2006.
- [82] J. Zou, R. K. Ward, and D. Qi. The generalized fibonacci transformations and application to image scrambling. In *Proceeding of the IEEE international conference on Acoustic, speech and signal processing*, pages 385–388, 2004.
- [83] available at <http://www.isi.edu/nsnam/ns/>.
- [84] G. J. Vanderbrug and A. Rosenfeld. “two-stage template matching,”. *IEEE Transaction on Computer*, 26:384–393, April 1977.
- [85] A. Hornberg. *Handbook of Machine Vision*. Wiley, October 2006.
- [86] H.-J. Cho and T.-H. Park. Template matching method for smd inspection using discrete wavelet transform. In *SICE Annual Conference*, pages 3198–3201, August 2008.
- [87] S. Sassanapitak and P. Kaewtrakulpong. An efficient translation-rotation template matching using pre-computed scores of rotated templates. In *6th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, volume 2, pages 1040–1043, May 2009.
- [88] A. Crispin and V. Rankov. Automated inspection of pcb components using a genetic algorithm template-matching approach. *The International Journal of Advanced Manufacturing Technology*, 35:293–300, December 2007.

- [89] C.-H. Wu, D.-Z. Wang, A. Ip, D.-W. Wang, C.-Y. Chan, and H.-F. Wang. A particle swarm optimization approach for components placement inspection on printed circuit boards. *Journal of Intelligent Manufacturing*, 35:610–620, June 2008.
- [90] M.R. Teague. Image analysis via the general theory of moments. *IEEE Journal Optical Society of America*, 70:920–930, August 1980.
- [91] M.K. Hu. Visual pattern recognition by moment invariants. *IRE Transaction on Information Theory*, IT-8:179–187, February 1962.
- [92] J. Flusser and T. Suk. Rotation moment invariants for recognition of symmetric objects. *IEEE Transaction on Image Processing*, 15:3784 – 3790, December 2006.
- [93] W.-Y. Kim and P. Yuan. A practical pattern recognition system for translation, scale and rotation invariance. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 391–396, June 1994.
- [94] A.Neri and G. Iacovitti. Maximum likelihood localization of 2-d patterns in the gauss-laguerre transform domain: theoretic framework and preliminary results. *IEEE Transaction on Image Processing*, 13:72–86, January 2004.
- [95] A. Neri, M. Carli, V. Palma, and L. Costantini. Image search based on quadtree zernike decomposition. *Journal of Electronic Imaging*, 19(4), October-December 2010.
- [96] A. Neri, M. Carli, V. Palma, and L. Costantini. Template matching based on quadtree zernike decomposition. In *IS&T SPIE’s Annual Symposium Electronic Imaging Science and Technology*, San José, California, USA, January 2009.
- [97] L. Capodiferro, M. Carli, L. Costantini, A. Neri, and V. Palma. Adaptive riesz basis decomposition for image search. In *EUSIPCO 2009, 17th European Signal Processing Conference*, Glasgow, Scotland, August 2009.
- [98] Y.Bin and P.J.-Xiong. Improvement and invariance analysis of zernike moments using as a region based on shape descriptor. In *XV Brazilian Symposium on Computer Graphics and Image Processing*, pages 120–127, 2002.

- [99] G. Amayeah, A. Erol, G. Bebis, and M. Nicolescu. Accurate and efficient computation of high order zernike moments. *Advances in Visual Computing*, 3804, 2005.
- [100] C. Chong, R. Mukundan, and P. Raveendran. An efficient algorithm for fast computation of pseudo-zernike moments. *International Journal Of Pattern Recognition and Artificial Intelligence*, 17, 2003.
- [101] S. M.Pawlak. On the accuracy of zernike moments on image analysis. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 20, 1998.
- [102] Y. Xin, M. Pawlak, and S. Liao. Image reconstruction with polar zernike moments. *Pattern Recognition and Image Analysis*, 3687, 2005.
- [103] C. Harris and M. Stephens. A combined corner and edge detector. In *The Fourth Alvey Vision Conference*, pages 147–151, 1988.
- [104] Corel Gallery Magic. Available at “<http://www.corel.com>”. 2000.
- [105] Z. Stejic, Y. Takama, and K. Hirota. Genetic algorithm-based relevance feedback for image retrieval using local similarity patterns. *Information Processing & Management*, 39, 2003.
- [106] S. Rudinac, M. Rudinac, B. Reljin, M. Uscumlic, and G. Zajic. Global image search vs. regional search in cbir systems. In *VIII International Workshop on Image Analysis for Multimedia Interactive Services*, pages 14–17, June 2007.
- [107] J. Z.Wang. Simplicity: Semantics-sensitive integrated matching for picture libraries. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 23, 2001.
- [108] L. Sorgi, N. Cimminiello, and A. Neri. Keypoint selection in the laguerre-gauss transformed domain. In *17th British Machine Vision Association*, volume 2, pages 539–548, September 2006.
- [109] Z. Zalevsky, I. Ouzieli, and D. Mendlovic. Wavelet-transform-based composite filters for invariant pattern recognition. *Applied Optics*, 35:3141–3147, June 1996.

- [110] M. Carli, F. Coppola, G. Iacovitti, and A. Neri. Translation, orientation and scale estimation based on laguerre-gauss circular harmonic pyramids. In *SPIE Conference Photonics West*, 2002.
- [111] L. Capodiferro, E. D. Di Claudio, G. Iacovitti, and F. Mangiatordi. Application of local fisher information analysis to salient points extraction. In *Proceedings of the IASTED International Conference on Signal Processing, Pattern Recognition, and Applications, SPPRA 2008*, Innsbruck, Austria, February 2008.