POLITECNICO DI MILANO
Facoltà di Ingegneria dell'Informazione

POLO REGIONALE DI COMO
Master of Science in
Computer Engineering

No-Reference Pixel-Based Estimation of Channel-Induced Distortion in H.264/AVC Video

Supervisor: Prof. Marco Tagliasacchi
Assistant Supervisor: Prof. Giuseppe Valenzise

Master Graduation Thesis by: Stefano Magni Id. number 707610

Academic Year: 2009/2010

POLITECNICO DI MILANO
Facoltà di Ingegneria dell'Informazione

POLO REGIONALE DI COMO
Corso di Laurea Specialistica
in Ingegneria Informatica

Stima No-Reference Pixel-Based della Distorsione di Canale in un Video H.264/AVC

Relatore: Prof. Marco Tagliasacchi
Correlatore: Prof. Giuseppe Valenzise

Tesi di Laurea di: Stefano Magni Matr. 707610

Anno Accademico: 2009/2010

Abstract

Nowadays IP networks are used to deliver multimedia contents. Usually the used networks provide only a best effort service, so there is no guarantee about the delivery of contents. The final user and the provider may decide to stipulate a service level agreement (SLA), fixing the perceived video quality at the end user side. If the objective quality is not reached the provider pays a penalty. So it is useful to search for a metric to estimate the perceived video quality working at the end-user terminal. Typically video contents are compressed by the encoder before transmission to the decoder and so the video may suffers from two different types of distortions. Quantization distortion is due to lossy compression while channel distortion is linked to the losses of packets during transmission. In particular, in the second case, the decoder tries to recover the lost information running a specific concealment algorithm, that tries to guess the lost data from the correctly received ones. Obviously this procedure can lead to wrong reconstruction, and visual impairments arise in the reconstructed video.

Our objective is to estimate channel distortion at the decoder side using only the reconstructed video, creating a no reference pixel base (NR-P) video quality monitoring. Conventionally, these methods assume the availability of the corrupted bitstream. However in some situations this is not possible, e.g. because the bitstream is encrypted or processed by third party decoders, and only the decoded pixel values can be used. Our objective is reached thanks to NORM a no reference quality monitoring for channel distortion that uses both decoded video and bitstream information. In particular we estimate bitstream NORM inputs from the reconstructed video, creating a NR-P NORM version. It turns out that the major limitation in this scenario is the lack of knowledge about which slices have been actually lost. Our major effort is so linked to the map of lost macroblock estimation. In particular to solve this problem we search for concealed macroblocks with visual impairments starting from mild assumptions valid for

a large class of concealment techniques. For each frame prediction residuals' energy at macroblock level is used to recognize the desired blocks and a confidence is applied to avoid false positives. Finally also spatial relationship between lost blocks is taken into account thanks to Markov random fields model.

The estimated bitstream inputs are used to run NORM in its NR-P version. The obtained channel distortion estimations are well correlated (linear correlation coefficient larger than 0.9 over a wide range of packet loss rates) wrt the real distortion calculated at frame and sequence level.

Contents

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

The use of IP network for delivery of multimedia contents is gaining an increasing success. Typically, networks used for transmission provide only best effort services, so there is no guarantee that the content is delivered to the final user without distortion. In some circumstances the content provider and the user stipulate a service level agreement (SLA) that fixes a perceived video quality at the end-user terminal. If the SLA is unfulfilled the provider pays a penalty to the user. In this situation, where the network provides only a best effort service, it is useful to search for a metric to estimate the perceived video quality working at the end-user terminal. In IP networks and video broadcasting applications the video contents are compressed before transmission. At the provider side the video is encoded exploiting spatial and temporal redundancy, and some information is discarded to achieve an higher compression. At the end-user terminal

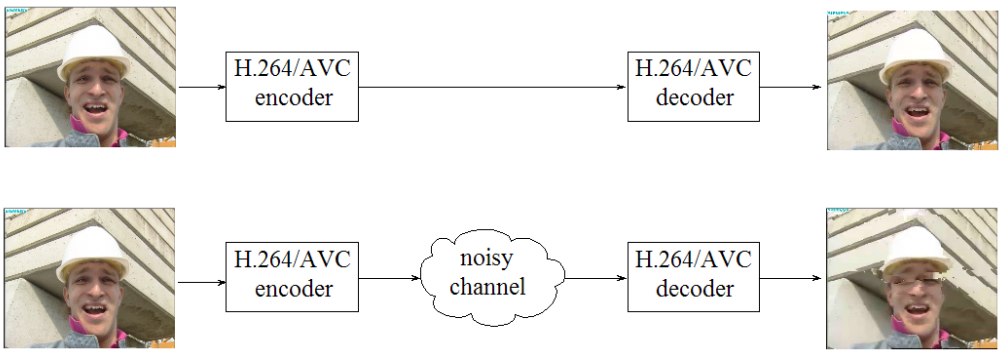

Figure 1.1: Overview of channel distortion effect over received video

a decoder reconstructs the sequence which may suffer from two different kinds of distortion. The first one is the quantization distortion, and it is due to the lossy nature of the encoding process. Since in the coding phase a significant fraction of information is discarded to reach a higher compression, perfect reconstruction is not possible and artifacts could arise in the reconstructed sequence. On the other hand we have channel distortion, in which the loss of information is due to the transmission process, it is in fact possible that some information is not received by the decoder, Figure 1.1. In this case the decoder is not able to reconstruct the original scene and tries to recreate the lost portion running a specific concealment technique. The results obtained by the concealment are not always as expected, and sometimes an appreciable distortion is introduced not only in the frames affected by the loss, but also in other frames, due to predictive nature of the coding process. We focus our attention on this second kind of distortions.

The main objective of this work is to estimate the channel distortion using only the reconstructed video at the decoder side, for video quality monitoring purposes.

In the literature the algorithms proposed to solve the video quality monitoring problem often take advantage of knowing the bitstream. However, in some circumstances, the bitstream may be unavailable, e.g. because it is encrypted and/or processed by third party decoders and only the pixel values of the decoded video sequence can be used. In this case, the no-reference quality monitoring task is pixel based, in the sense that both the coding parameters and the map of pixels that have been lost must be estimated from the pixel values at the decoder side. To solve this problem is therefore necessary to estimate all bitstream parameters from the decoded video, such as:

- Motion Vectors (Real and Concealed)
- Residuals
- Structure of the Group Of Pictures, (GOP)
- Coding Modes of each macroblock

- Map of Lost Macroblocks

We focus our attention on the map of lost macroblocks since is crucial for a good channel distortion estimation and there are no available methods solving this problem in current state of the art quality monitoring systems. This information is so estimated form the bitstream and used to feed NORM, a No-Reference video quality Monitoring proposed by Naccari et al. (2009), which is able to to give an estimation of the mean square error (MSE) at the macroblock level using only data available at the decoder side. The correlation that NORM exhibits against MSE is quite high (correlation coefficient 0.80) with respect to other similar work. Moreover this fine-granularity estimation is particularly beneficial as it can be used to compute more sophisticated perceptual metrics (such as the SSIM metric) that leverage localized distortion information. In the following section we classify the main approaches proposed in the literature. Finally we present our novel contributions to this problem.

1.1 Related Work

First of all it is useful to classify the metrics that can be used to fulfill the video quality monitoring task as described in Winkler (2009). Two main classes can be defined, subjective and objective metrics.

The first class have been formalized in BT 500-10 (2002) and P 910 (2008), suggesting standard viewing conditions, criteria of selection of the observers and test material, assessment procedures, and data analysis methods. The outcome of any subjective experiment are quality ratings from viewers, which are then averaged for each test clip into Mean Opinion Scores (MOS).

On the other side objective quality metrics are designed to characterize the quality of a video with respect to a predictable video viewer opinion, and so trying to predict MOS values and can be classified as follows:

- *Data metrics*: measure the fidelity of the signal without taking into account its content. MSE and PSNR are good examples of this class since none of them takes into account the different visual importance of the pixels.

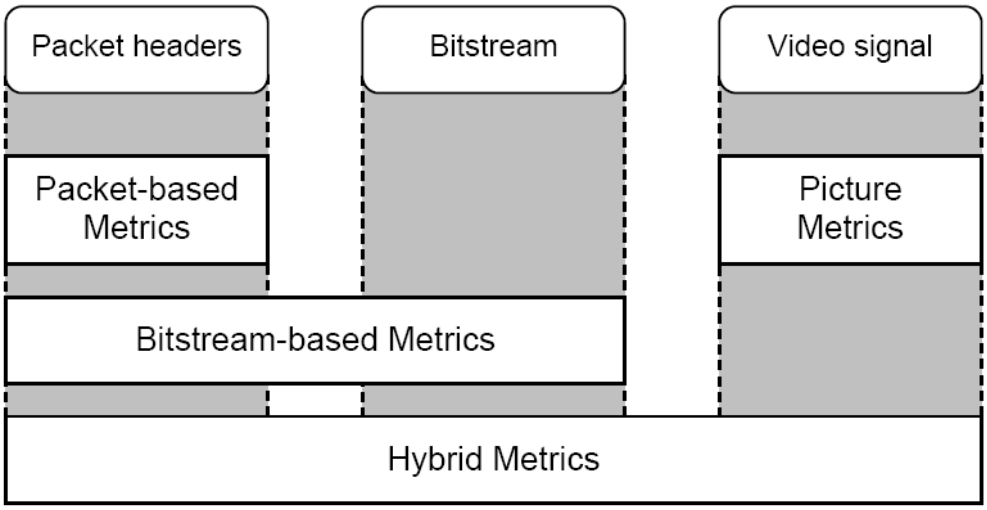

Figure 1.2: Classification of packet-based, bitstream-based, picture and hybrid metrics, adapted from ITU-T.

- *Picture metrics*: measure the video quality using the visual information contained into the sequence. They specifically account for effect of distortion and content on perceived video quality, reeling on human vision system models or extracting specific features and artifacts from the video.
- *Bitstream metrics*: look directly at the information contained into the bitstream without full decoding. These approaches obviously lead to a lower processing requirements, making possible to process different bitstreams in parallel.
- *Hybrid metrics*: which use a combination of the previously described approaches.

This classification is also shown in Figure 1.2. Moreover metrics could be classified with respect to the amount of needed reference information:

- Full Reference (FR) metrics measure the degradation with respect to a reference video. The entire unimpaired and the uncompressed reference video must be available, moreover a precise spatial and temporal alignment must be reached in order to match every pixel with its exact counterpart in the reference video.

- No Reference (NR) metrics estimate the degradation using only the received sequence, without the help of any references. These metrics are completely free from video alignment issues and their main challenge lies in finding an estimation with high correlation with respect to the full reference measure.

- Reduced-reference (RR) metrics are a combination of the previous FR and NR metrics. Features are extracted from reference and tested videos and comparison is made upon these. This class of metrics permits to avoid some necessary assumption made in NR approach keeping amount of reference information manageable.

The FR class is more suitable for offline video quality measurement such as codec tuning or lab testing, while NR and RR classes are used in monitoring of in-service video. Obviously the RR metrics need a back-channel to access to the reference information.

The NR methods can be moreover divided into two classes:

- No-Reference Pixel (NR-P) methods which use only the pixels of the reconstructed video sequence.

- No-Reference Bitstream (NR-B) methods which use only the bitstream information.

The NR-P methods proposed till now do not achieve accurate quality evaluations. In fact due to the lack of original video information it is difficult to distinguish video degradation from video features. In the literature, the NR-P approach has been used only to blindly estimate the degree of blur Marziliano et al. (2002) or blockiness Tan and Ghanbari (2000), while there is not much about channel-induced distortion. On the other hand NR-B methods are used in situation in which it is not possible to access to the final decoded video such as in Reibman et al. (2004); Yamada et al. (2010). So usually a mixture of NR-P and NR-B methods (NR-PB) are used to achieve video quality monitoring.

As described in Winkler (2009), the purposes of video quality measurement can be summarized as follows:

- Defining the meaning of MOS for a given application.
- Defining the method for MOS prediction that is reliable.
- Defining the method for MOS prediction which is reproducible.

Existing standards achieved some of these objective. We summarize now briefly some solutions for channel distortion estimation. This task can be fulfilled either at the transmitter or at the receiver side. At the transmitter original and decoded sequence are available, so challenge is related to the unknown error pattern. Otherwise NR-BP methods working at the decoder side know perfectly the error pattern but the original sequence is unavailable.

The techniques proposed in K. Stuhlmuller and Girod (2000) N. Farber and Girod (1999) R. Zhang and Rose (2000) Yang and Rose (2007) rely on a statistical representation of the channel providing an estimate of the channel distortion at frame macroblock or pixel level. The main goal pursuit in this scenario is to provide a mean of tuning encoder parameters to obtain an optimal end-to-end coding efficiency. At the receiver the deterministic knowledge of the error pattern simplifies the task, however the unavailability of the original video forces to adopt a no-reference method complicating the problem.

Work described in Reibman et al. (2004) suggests an algorithm able to estimate MSE distortion with any conventional motion compensated video codec. Different granularity level are available: Full Parse (FP), Quick Parse (QP), and No Parse (NP) with different level of complexity and estimation accuracy. The FP method gives an accurate estimation of the channel distortion at pixel level. To achieve this goal it analyzes some parameters by entropy decoding and inverse quantization on the bitstream. The QP method relies on the analysis of the received bitstream at the transport level, giving an estimate of channel induced distortion at slice level. Finally the NP method simply estimates the channel distortion wrt the packet loss rate (PLR) experienced at the decoder side, no

bitstream information are needed in this case. These method provide a low computational complexity and so are particular useful for monitoring by the network provider point of view.

The model proposed in T. Shu and Gu'erin (2005) is a tradeoff between accuracy and computational complexity, that tries to estimate the channel distortion at sequence level. A model to estimate the received video quality is created taking into account parameters such as the used codec, the adopted error concealment strategy, the bit-rate and the packetization used. The relative PSNR (rPSNR) metric is then defined as the difference between the PSNR at the receiver side and the pre-negotiated target PSNR, in this way is possible to avoid the dependence from a particular sequence. The results obtained by this approach on real video sequence and network condition reveal good correlation between real and estimated values.

The approach proposed in Yamada et al. (2007) is embedded within the H.264/ AVC compliant decoder and achieve good results with only a little computational complexity overhead. The method is based on the concept of error concealment effectiveness. Concealment algorithm achieve different performance wrt motion complexity and local texturing of lost macroblock. The authors propose a metric to measure the error concealment effectiveness that relies on motion information and boundary distortion, taking advantage of the know slice pattern. The method achieve reasonably accurate results in the estimation of the channel distortion at sequence level.

In S. Kanumuri and Vaishmpayan (2006) machine learning classifier are used to predict packet loss visibility in H.264 coded bitstream. Training data were collected in extensive subjective campaigns and a NR and RR approach are proposed to achieve video quality assessment.

The NORM algorithm, proposed in Naccari et al. (2009) and shown in 1.3, is a NR-PB method that receives as input a H.264/AVC compliant bitstream that has been transmitted over a noisy channel and the reconstructed video. The received bitstream is processed by the H.264/ AVC decoder, which applies its own embedded concealment strategy over lost data. The decoded frame, together with

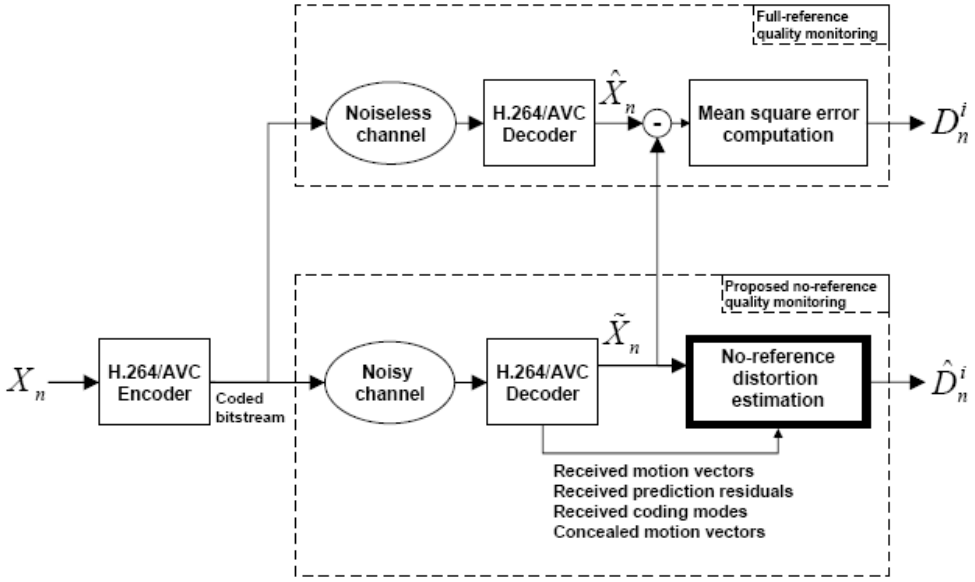

Figure 1.3: Block diagram of the proposed NORM algorithm.

the received/ concealed motion vectors, prediction residuals and coding modes are fed into NORM, which provides an estimate of the channel induced distortion $\widehat{MSE}_n^i$, for the ith macroblock in frame n. Also, NORM needs to know the pattern of channel errors, which consists of a binary map of the macroblocks that have been lost during transmission. The complexity of motion affects the accuracy of NORM estimate of true MSE, giving correlation values around 0.76-0.83 for a complex motion, and around 0.81-0.93 for a simpler one.

1.2 Novel Contributions

Our objective is to estimate bitstream information from the decoded video in order to feed NORM obtaining a NR-P method for channel distortion estimation. As already said, the bitstream is not always available. Different scenarios fit this assumption. First of all we can think about a bitstream which is not accessible since is encrypted by third party decoder. In other situation, the video could suffer of multiple coding-decoding processes. Since the available bitstream is related only to the last transmission, no information about previous transmissions losses can be reached, and the only way to exploit an estimation of the total channel

distortion is to extract information from the reconstructed video sequence.

An illustrative example of this situation is given in Figure 1.4, where a video signal, X, is first coded through a H.264/AVC compliant ITU (2003) encoder, and the resulting bitstream b is transmitted over an error-prone network. The noisy channel drops packets with some unknown packet loss rate (PLR), thus the received bitstream $\tilde{b}$ may differ from the original b. A H.264/AVC decoder processes the corrupted bitstream, possibly applying an error concealment startegy as in Sullivan et al. (2003) to partially alleviate the effect of packet losses, and produces a reconstructed video $\tilde{X}$ in the pixel domain. This decoded video $\tilde{X}$ is all the information we postulate to have in order to produce an estimate of the mean square error distortion, $\widehat{MSE}$, between the error-free decoded video $\hat{X}$ and the noisy one $\tilde{X}$, as in the NORM setting. The distortion introduced by lossy coding, indeed, can be approximately considered to be uncorrelated with channel-induced distortion He et al. (2002), so the two terms can be summed up in order to obtain the overall distortion with respect to X.

Our challenge is to estimate motion vectors, prediction residuals and map of lost macroblocks from the reconstructed decoded video. Motion vectors are found by performing motion estimation on the decoded sequence. Any motion estimation algorithm can be used for this purpose. We set a number of reference frames k on which the search is carried out, as it is not known which is the exact number of reference frames used by the encoder. Prediction residuals can be readily computed once MVs have been found.

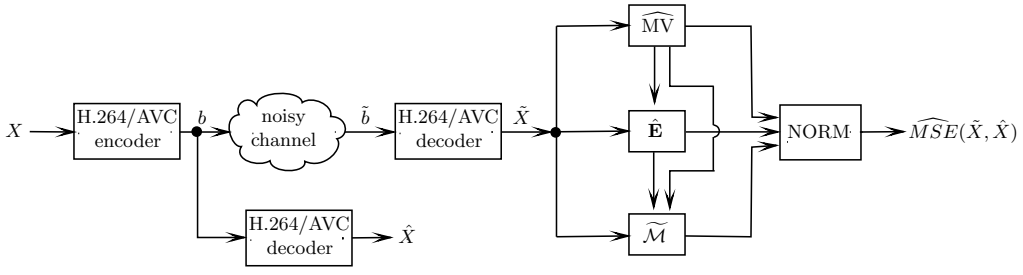

Figure 1.4: Overview of the blind no-reference quality assessment system. We estimate the missing parameters from the corrupted decoded video $\tilde{X}$ and use them as input to NORM. The results is a macroblock-level map of MSE distortion between the noisy decoded $\tilde{X}$ and the video reconstructed at the encoder $\hat{X}$.

In literature map of lost macroblock estimation problem is not solved and this map is crucial for NORM to achieve a good channel estimation. The concealment procedures are not standardized and a simple retro-engineering of the processes is not possible. To avoid an estimation which is linked to a particular type of concealment it is necessary to made mild assumptions that are valid for a large class of concealment techniques. Basically two types of concealment can run on a decoder: spatial concealment and temporal concealment. Spatial concealment is applied when a macroblock belonging to I frame is lost and reconstructs the block as a combination of its neighborhood. Lost macroblocks belonging to a P frame are on the other hand reconstructed with a temporal concealment technique, copying a macroblock from previous reference frame. Summarizing the mild assumptions that can be done above the concealed macroblocks are:

- Spatial concealed macroblocks are a combination of neighborhood with residual's energy near to zero.
- Temporal concealed macroblocks must have a predictor, in a previous reference frame, with prediction residual's energy near to zero.

From these consideration we built an estimation of the lost macroblocks map able to feed NORM achieving good results in channel distortion estimation.

CHAPTER 2

BACKGROUND

In this chapter are presented the tools used in our work. First of all some information about the H.264/AVC coding standard are given, with particular emphasis on the prediction process and packetization. Then an overview of the non-standardized concealment technique of the used reference software are presented. Finally the already introduced NORM algorithm is presented focusing on the estimation process of the different types of channel distortion.

2.1 H.264/AVC Standard

Each frame of a video sequence is encoded to produce a coded picture, which is composed by a certain number of macroblocks. Moreover, inside each picture, macroblocks are grouped into slices. An I slice can contain only intra predicted macroblocks, while a P slice can contain inter and intra macroblocks. A video picture can be coded with a chosen number of slices and no assumptions are made upon the number of macroblocks in each slice, which can spawn from one (1 macroblock per slice) to the total number of macroblocks in a picture (1 slice per picture). Finally macroblocks number per slice need not to be constant within a picture. Different type of slices are available and a picture can be composed of a mixture of them (Table 2.1). Baseline profile, for example, contains I and P slices only, while Main Profile could also contain B slices. A simplified illustration of the syntax of a coded slice is shown in Figure 2.1. Slice type and reference to

the picture to which slice belongs, are stored inside slice header. The slice data contains the coded macroblocks, each one containing a series of header elements and coded residual data. It must be noticed that P slices can also contain skipped macroblocks. When a skipped macroblock is signalled in the bitstream no further information about it are sent. The decoder simply reconstructs a vector for the skipped macroblock reconstructing it by motion-compensated prediction.

Slyce Type	Description	Profile(s)
I (Intra)	Contains only I macroblocks (each block or macroblock is predicted from previously coded data within the same slice).	All
P (Predicted)	Contains P macroblocks (each macroblock or macroblock partition is predicted from one t 0 reference picture) and/or I macroblocks.	All
B (Bi-predictive)	Contains B macroblocks (each macroblock or macroblock partition is predicted from a list 0 and/or a list 1 reference picture) and/or I macroblocks.	Extended and Main
SP (Switching P)	Facilitates switching between coded streams; contains P and/or I macroblocks.	Extended
SI (Switching I)	Facilitates switching between coded streams; contains SI macroblocks (a special type of intra coded macroblock).	Extended

Table 2.1: H.264 slice modes

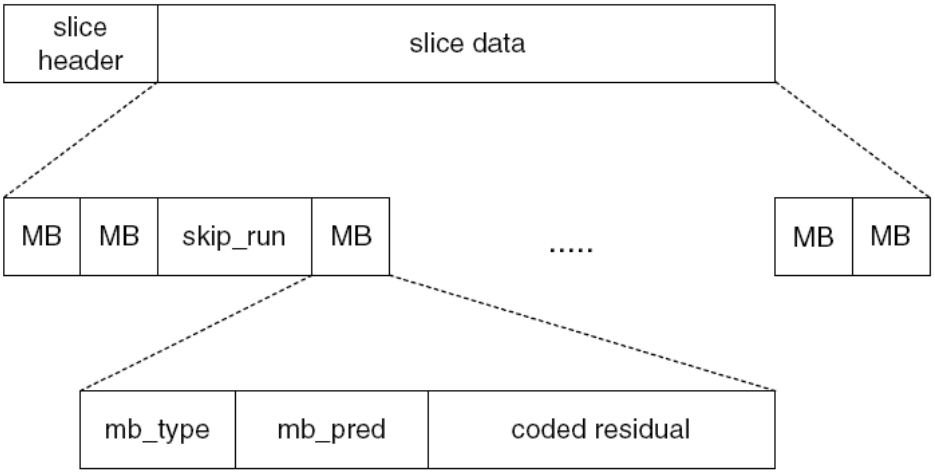

Figure 2.1: Slice Syntax

Each coded macroblock is predicted from previously decoded data by H.264. Intra macroblocks are predicted from samples in the same frame that have al-

ready been encoded, decoded and reconstructed. On the other side inter macroblocks are predicted from samples belonging to previously reconstructed frames. A prediction can be defined as a model that resembles the macroblock under consideration as much s possible. Once the prediction is created from already encoded data, is subtracted from the current macroblock. The obtained residuals are compressed and sent to the decoder, together with all information useful to repeat the prediction process. The decoder recreates the prediction and adds the residuals.

Inter prediction creates a predictor from previously reconstructed frames by block-based motion compensation. Each 16x16 macroblock can be partitioned as shown in Figure 2.2, and each partition need a separate motion vector. Choosing

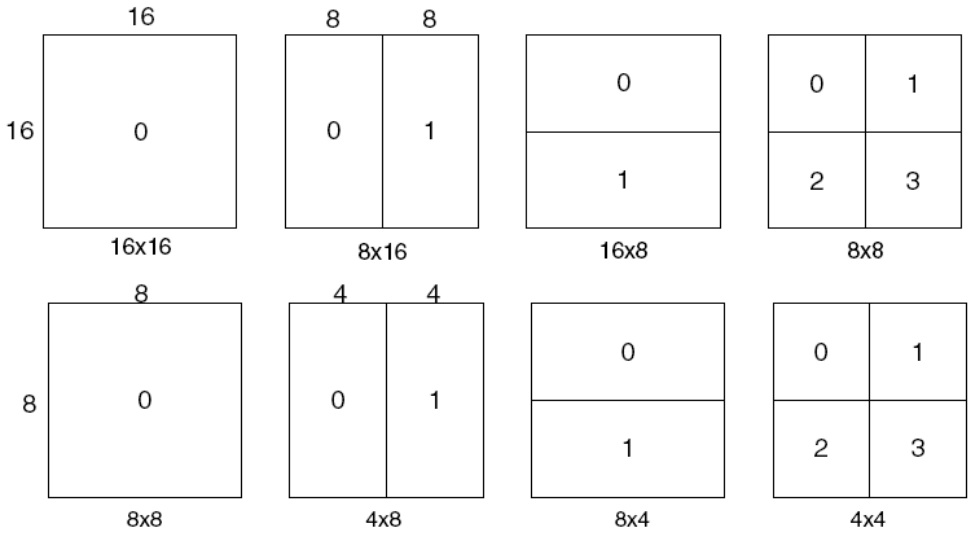

Figure 2.2: Macroblock and Sub-Macroblock Partition

large partitions assure a small number of bits during transmission but the motion compensated residuals may contain a significant amount of energy for high textured area. On the other hand small partitions may achieve better predictions but a larger number of bits is needed. Roughly speaking large partitions are suitable for homogenous areas while small ones for high detailed areas. It is trivial to understand that choose the appropriate partition can lead to higher compression performances. Each partition is so predicted from an area of the same size in the reference picture. Motion vectors has quarter-sample resolution, the miss-

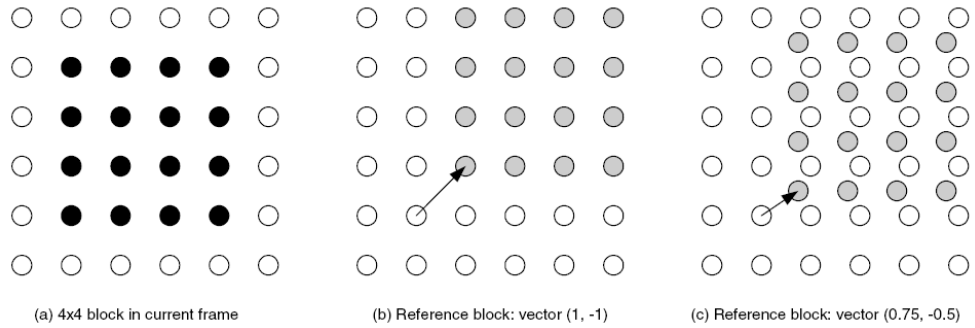

Figure 2.3: Example of integer and sub-sample prediction

ing samples are recreated by interpolation from nearby coded samples (Figure 2.3). The predictor is so chosen minimizing a cost function. Finally to achieve an higher compression a motion vector predictor (MVP) is created from vectors of nearby, since often motion vectors are highly correlated wrt its neighbor previously coded partitions.

Intra prediction creates a predictor based upon spatially near previously encoded and reconstructed blocks. As for inter prediction residuals are calculated by subtracting the predictor to the current macroblock. There are a total of nine optional prediction modes for each 4x4 block and more four modes for a 16x16 block. In Figure 2.5 it is possible to see how different modes work. The predic-

M	A	B	C	D	E	F	G	H
I	a	b	c	d				
J	e	f	g	h				
K	i	j	k	l				
L	m	n	o	p				

Figure 2.4: Labeling of prediction samples (4x4)

tion of the samples a, b, c, . . ., p is done wrt the samples A-M (Figure 2.4). All the available modes are tested and the one achieving the best Sum of Absolute Errors (SAE) is chosen. The same considerations hold for 16x16 macroblock prediction which is alternative to the 4x4 prediction. Table 2.2 summarizes all possible the modes.

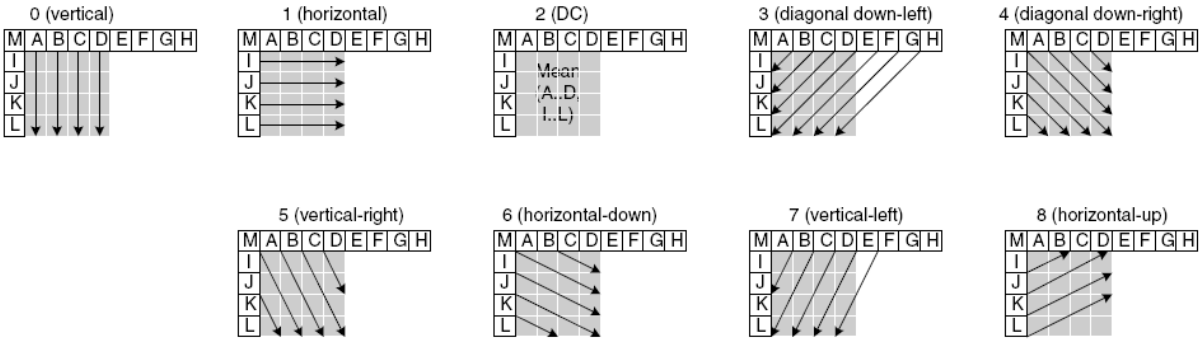

Figure 2.5: 4x4 prediction mode

Mode 0 (Vertical)	The upper samples A, B, C, D are extrapolated vertically.
Mode 1 (Horizontal)	The left samples I, J, K, L are extrapolated horizontally.
Mode 2 (DC)	All samples in P are predicted by the mean of samples A . . . D and I . . . L.
Mode 3 (Diagonal Down-Left)	The samples are interpolated at a 45. angle between lower-left and upper-right.
Mode 4 (Diagonal Down-Right)	The samples are extrapolated at a 45. angle down and to the right.
Mode 5 (Vertical-Right)	Extrapolation at an angle of approximately 26.6. to the left of vertical (width/height = 1/2).
Mode 6 (Horizontal-Down)	Extrapolation at an angle of approximately 26.6. below horizontal.
Mode 7 (Vertical-Left)	Extrapolation (or interpolation) at an angle of approximately 26.6. to the right of vertical.
Mode 8 (Horizontal-Up)	Interpolation at an angle of approximately 26.6. above horizontal.

Mode 0 (vertical)	Extrapolation from upper samples (H)
Mode 1 (horizontal)	Extrapolation from left samples (V)
Mode 2 (DC)	Mean of upper and left-hand samples (H + V).
Mode 4 (Plane)	A linear 'plane' function is fitted to the upper and left-hand samples H and V. This works well in areas of smoothly-varying luminance.

Table 2.2: H.264 spatial prediction modes

At the end of the prediction process to attenuate the blocking distortion a filter is applied to each decoded macroblock. In particular the deblocking filter is applied after the inverse transform in the encoder and in the decoder. The objective of the filter is to smooth edges, achieving better visual appearance in the decoded frames. Moreover the filtered macroblocks are used for motion-compensated prediction, this choice can improve compression performances since the filtered image is often a better prediction.

2.2 Concealment

In real applications there is no assurance that all packets are received by the decoder. So it may happen that a packet is lost due to network transmission problems. All slices and macroblocks information packetize inside it get lost too. The decoder tries so to reconstruct the lost macroblocks running a concealment algorithm. No standardization are made upon concealment procedures. For example in some cases no efforts are done and simply a green macroblock is shown. We prefer to focus on higher level concealment techniques:

- Spatial concealment: reconstructs the block as a linear combination of its neighborhood.
- Temporal concealment: copies a macroblock from previous reference frame.

In our work we used concealment algorithms implemented in the reference software. We briefly describe their implementations to better understand how concealment can lead to artifacts in the reconstructed video scene.

First of all each macroblock is tagged wrt its status: "correctly received" in which macroblock is included was available for decoding; "lost" otherwise (Figure 2.6). All correctly received macroblocks are then decoded, and if status map contains "lost" macroblocks, concealment starts. The algorithm works at macroblock level starting from the slice structure and the status map of each frame. Each 16x16 macroblock tagged as "lost" in the status map is concealed and tagged

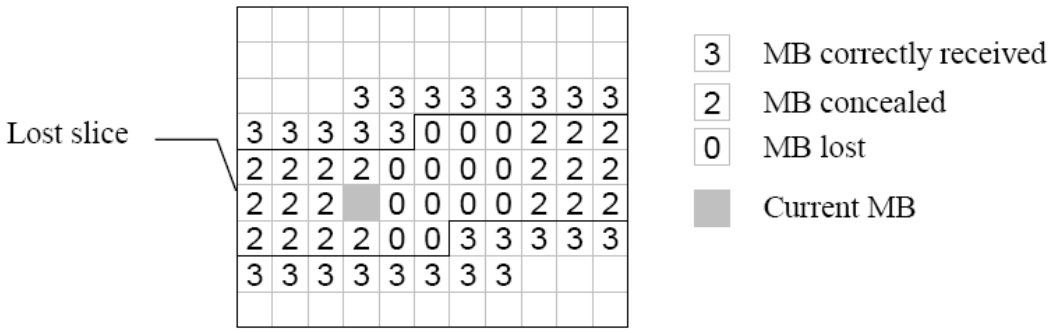

Figure 2.6: snapshot of the status map during the concealment phase where already concealed MBs have the status of "Concealed", and the currently processed (concealed) MB is marked as "Current MB".

as "concealed". The conceal processing order is of great importance, since there is no assurance that for each macroblock there is a set of "correctly received" neighbor macroblocks to be used during concealment process. In fact it may happen that also "concealed" macroblock are used during the process when no "correctly received" immediate neighbors of a "Lost" MB exist. It is possible that a wrong concealment procedure can lead to an error propagation to neighbor not already concealed macroblocks. "Lost" macroblock are processed starting from the edge of the frame first and then move inwards column-by-column, trying to avoid error propagation from the usually "difficult" center area to the "easy" side parts of the frame. Usually center areas are characterized by discontinuous motion and large coded prediction errors, since scenes usually take action in this portion of the sequence. On the other hand the side parts of the frame represent the background of the scene with continuous motion areas and similar motion over several frames.

First of all we focus on concealment procedure adopted in I frame by the reference software. The "lost" macroblocks in intra frame are spatially concealed since there is no assurance that previous frame areas may resemble the lost macroblocks. All the spatial concealment process is based on weighted sample averaging as described in Katsaggelos and Galatsanos (1998). The lost macroblock is restored as a weighted sum of the nearest samples belonging to the neighbor macroblocks. Each boundary samples is so weighted wrt the inverse of distance

between the boundary sample itself and the sample that need concealment, as described in the following formula:

$$SampleValue = \frac{(\sum a_i(B-d_i))}{\sum(B-d_i)} \tag{2.1}$$

where a_i is the boundary sample value belonging to the adjacent macroblock, B is the macroblock size and d_i is the distance between a_i and the sample to be concealed. As an example the missing sample in Figure 2.7 is calculated as follows:

$$SampleValue = \frac{(15(16-3)+21(16-12)+32(16-7)+7(16-8))}{(13+4+9+8)}$$

The conceal tries to use only "Correctly received" neighbor macroblocks if at least two of them are available, otherwise also "Concealed" macroblocks are involved in concealment process.

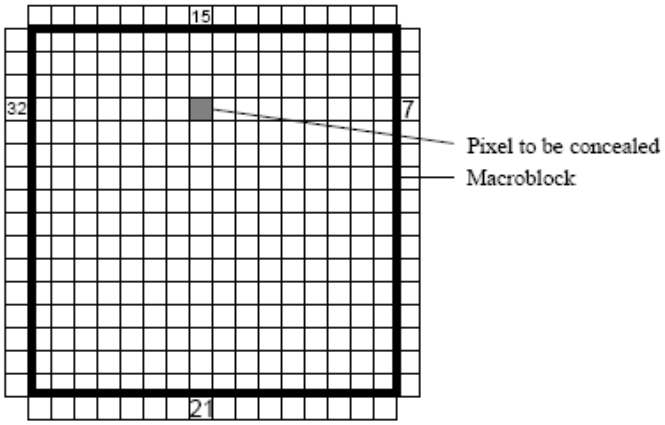

Figure 2.7: Spatial concealment based on weighted sample averaging.

The conceal procedure adopted in P frame tries to guess the possible motion vector of the lost macroblock on the basis of the motion vectors belonging to neighbors. Once the motion vector is found the concealed macroblock is simply recreated by motion copy from the correct reference frame. The algorithm implemented in the reference software is based on Lam et al. (1993). First of all motion activity of the picture under exam is investigated, if the average motion vector is smaller than a pre-defined threshold all the lost slices are concealed by copying from co-located positions in the reference frame. If this condition is not

achieved motion-compensated error concealment starts. Since motion vectors belonging to spatially near macroblocks are highly correlated, the motion vector of a "Lost" macroblock is predicted from motion vectors of neighbor macroblocks. This procedure is particulary suitable for continuous motion vector fields, such as a frame area covered by a moving foreground scene object. "Lost" macroblocks is so reconstructed by choosing the motion vector of the neighbor that select the macroblock which minimizes the Side Match Distortion (SMD). The minimum partition for neighbor macroblocks is 8x8, if subpartitions are present the 8x8 motion vector is obtained by averaging the subpartitions motion vectors. The SMD is calculated as:

$$SMD = \sum |i_{in} - i_{out}| \tag{2.2}$$

where i_{in} is the luminance of the samples at boundaries of the macroblock selected by motion vector under exam, while i_{out} is the luminance of samples at boundaries of the neighbor macroblocks, as shown in Figure 2.8. The winning predictor is so the one which minimizes the luminance change across boundaries.

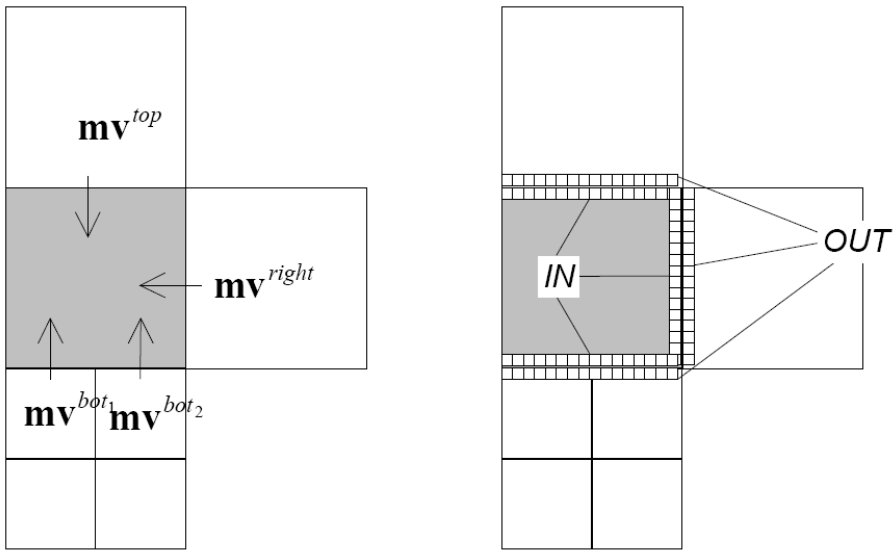

Figure 2.8: Motion Concealment

All the motion concealment process can be so summarized in the following

formula:

$$Prediction = \arg\min_{mv}(\sum |i_{in}(mv) - i_{out}|) \tag{2.3}$$

When multiple references are used, the reference frame of the candidate motion vector is used as the reference frame for the current macroblock. That is, when calculating SMD , the $i_{in}(mv)$ samples are from the reference frame of the candidate motion vector.

2.3 NORM

NORM was proposed in Naccari et al. (2009) as a No-Reference video quality Monitoring for the H.264/AVC standard.

NORM receives in input the decoded frame, the received/concealed motion vectors, prediction residuals and coding modes, giving as output an estimate of the channel induced distortion $\hat{D}_n^i$ for the ith macroblock in frame n. The accuracy of the estimate with respect to the distortion computed in full reference mode can be evaluated at macroblock level. In general channel distortion can be written as:

$$\hat{D}_n^i = \frac{1}{B^2}\sum_{x=1}^{B}\sum_{y=1}^{B}(E_n^i(x,y))^2 = \frac{1}{B^2}\sum_{x=1}^{B}\sum_{y=1}^{B}(\hat{M}_n^i(x,y) - \tilde{M}_n^i(x,y))^2 \tag{2.4}$$

where N is the number of macroblocks, $\hat{M}_n^i(x,y)$ is the BxB macroblock reconstructed at the decoder side with no channel losses, and $\tilde{M}_n^i(x,y)$ is the BxB macroblock reconstructed at the decoder side when channel losses occurred. The algorithm specifically considers four different types of distortion:

- $D_n^{i,SP}$ distortion due to spatial propagation
- $D_n^{i,TP}$ distortion due to temporal propagation
- $D_n^{i,SC}$ distortion due to spatial concealment
- $D_n^{i,TC}$ distortion due to temporal concealment

It is possible to demonstrate that these contributions could be used to calculate the induced channel distortion of each different type of macroblock:

- $D_n^i(intra-ok) = D_n^{i,SP}$: the distortion of a correctly received intra macroblock is due to spatial error propagation.
- $D_n^i(inter-ok) = D_n^{i,TP}$: the distortion of a correctly received inter macroblock is due to temporal error propagation.
- $D_n^i(intra-ko) = D_n^{i,SC} + D_n^{i,SP}$: the distortion of a lost intra macroblock is due to spatial error propagation and spatial concealment.
- $D_n^i(inter-ko) = D_n^{i,TC} + D_n^{i,SP}$: the distortion of a lost inter macroblock is due to temporal error propagation and temporal concealment.

So NORM is able to evaluate the distortion of each macroblock by simply estimating the four different types of distortion $D_n^{i,SC}$, $D_n^{i,TC}$, $D_n^{i,SP}$, $D_n^{i,TP}$.

The spatial error propagation $D_n^{i,SC}$ accounts for a negligible fraction of the overall macroblock distortion in intra-frames, so the estimated spatial error propagation $\hat{D}_n^{i,SC}$ is set to zero:

$$\hat{D}_n^{i,SP} = 0 \tag{2.5}$$

$D_n^{i,SP}$ can be so erased by $D_n^i(intra-ok)$ and $D_n^i(intra-ko)$ definition.

The distortion due to the temporal propagation of errors $D_n^{i,TP}$ is modeled as a weighted sum of the distortions already found for the macroblocks used as predictors:

$$\hat{D}_n^{i,TP} = \frac{1}{16}\sum_{q=1}^{16}\Big(\sum_{p=1}^{N_o(q)} \eta_p \hat{D}_{n-r(q)}^{(p)}\Big) \tag{2.6}$$

where $\hat{D}_{n-r(q)}$ is the distortion of one of the $N_o(q)$ $(1 \leq N_o(q) \leq 4)$ $4x4$ subpartition that predicts the macroblock under consideration and η_p is proportional to the number of pixels of subpartition involved into prediction. This distortion is used to calculate $D_n^i(inter-ok)$ and $D_n^i(inter-ko)$

The distortion due to the action of the spatial concealment is related to the loss of high frequency content of the lost macroblock, caused by the spatial interpolation performed during concealment. NORM estimates this loss by comparing the interpolated block with the one obtained with a simple zero-motion temporal concealment, which typically preserves the high frequency content of the original

block:

$$\hat{D}_n^{i,SC} = \frac{1}{B^2}\sum_{x=1}^{B}\sum_{y=1}^{B}(\tilde{M}_n^i(x,y) - \tilde{M}_n^{i,0}(x,y))^2 \tag{2.7}$$

where $\tilde{M}_n^i(x,y)$ is the macroblock reconstructed by the spatial concealment, while $\tilde{M}_n^{i,0}(x,y)$ is the macroblock obtained by a simple zero motion concealment. This distortion is involved into $D_n^i(intra-ko)$ calculation.

Temporal concealment, distortion is due to the loss of the original motion vectors and to the lack of prediction residuals, both terms are explicitly considered by NORM. Temporal concealment distortion $D_n^{i,TC}$ estimates the difference between the predictor provided by temporal concealment and the one corresponding to the original motion vectors, and can be written as:

$$D_n^{i,TC} = D_n^{i,MV} + D_n^{i,PR} \tag{2.8}$$

where $D_n^{i,MV}$ and $D_n^{i,PR}$ represent respectively distortion due to lack of real motion vector and prediction residuals. The distortion induced by lack of motion vectors is estimated as:

$$\hat{D}_n^{i,MV} = \frac{1}{B^4}\sum_{j=0}^{B-1}\sum_{k=0}^{B-1}\Phi_n^i(\omega_j,\omega_k)(1-\cos(\omega_j\delta_x+\omega_k\delta_y)) \tag{2.9}$$

where $\Phi_n^i(\omega_j,\omega_k)$ is the estimated power spectral density of the predictor obtained with the temporal concealment and δ_x , δ_y are the differences between the real and concealed motion vector along the two axis. The estimation $\hat{\delta}_x$ and $\hat{\delta}_y$ are calculated as the standard deviation of the $8x8$ neighbor motion vectors wrt the concealed one:

$$\hat{\delta}_n^i = \sqrt{\frac{1}{L}\sum_{l=1}^{L_\varphi}(\tilde{v}_n^i - v_{8x8}^l)^2} \tag{2.10}$$

where v_{8x8}^l is the lth candidate motion vector related to the $8x8$ neighbor macroblocks used by temporal concealment. The distortion induced by lack of pre-

diction residuals is estimated as:

$$\hat{D}_n^{i,PR} = \frac{1}{B^2} \sum_{x=1}^{B} \sum_{y=1}^{B} \Theta_{n-r}^{i}(x+\tilde{v}_x^i, y+\tilde{v}_y^i)^2 \tag{2.11}$$

where $\tilde{v}_x^i$, $\tilde{v}_y^i$ are the concealed motion vector, and $\Theta_{n-r}^{i}(x+\tilde{v}_x^i, y+\tilde{v}_y^i)$ are the prediction residuals of the macroblock used to create the concealment prediction. The total distortion $D_n^{i,TC}$ is so used to calculate $D_n^i(inter-ko)$.

So the distortion for each macroblock can be readily computed simply applying the following algorithm:

for $n = 0$ to N **do**
  for $i = 0$ to M **do**
    if macroblock i is lost then **then**
      if if macroblock i ε I frame **then**
        $\hat{D}_n^i = \hat{D}_n^i(intra-ko) = \hat{D}_n^{i,SC}$
      else
        $\hat{D}_n^i = \hat{D}_n^i(inter-ko) = \hat{D}_n^{i,TC} + \hat{D}_n^{i,TP}$
      end if
    else
      if macroblock i ε P or B frame then **then**
        $\hat{D}_n^i = \hat{D}_n^i(intra-ok) = 0$
      else
        $\hat{D}_n^i = \hat{D}_n^i(inter-ok) = \hat{D}_n^{i,TP}$
      end if
    end if
  end for
end for

CHAPTER 3

STUDY OF NR-P NORM PERFORMANCES

One of the objective of this work is to run NORM without the help of bitstream information, NORM inputs must be so estimated from the decoded video. As a reminder NORM inputs that must be estimated are:

- Motion Vectors (Real and Concealed)
- Residuals
- Structure of the Group Of Pictures, (GOP)
- Coding mode of each Macroblock
- Map of Lost Macroblocks

In the following sections we separately study the impact of each estimation over NORM performances.

3.1 Motion Vector and Residuals Estimation

Real and concealed motion vectors are used by NORM to search for distortion created by temporal concealment, and distortion due to temporal propagation. Motion vectors are estimated by performing motion estimation on the decoded sequence. Any motion estimation algorithm can be used for this purpose. We

set a number of reference frames k on which the search is carried out, as it is not known which is the exact number of reference frames used by the encoder. Larger values of k provide a better estimation, but they clearly entail a larger computational cost. We use $k = 5$ in our experiments.

Prediction residuals can be readily computed once MVs have been found. Together with the prediction residuals, for each frame of $M \times N$ pixels we build a $(M/B) \times (N/B)$ map of prediction residual energies, whose ith entry gives the MSE distortion between the ith $B \times B$ macroblock in the current frame and its respective predictor in the reference frame.

Our tests are executed over *Foreman*, *News*, *Mobile* CIF resolution video sequences. The video sequence has been coded with a fixed quantization parameter for I and P slices (QP = 36), with a frame rate of 30 Hz, using the H.264/AVC reference software encoder (version JM12.3 (JVT)) with the main profile.

Each coded frame is partitioned into slices, where each slice contains a horizontal row of macroblocks. Each coded slice is then packetized according to the real-time transfer protocol (RTP) specifications Wenger (2003). The simulated error-prone channel drops coded packets according to a packet loss rate (PLR) in the range [0.1 10]. The error patterns have been generated using a two-state Gilbert's model Gilbert et al. (1960) with average burst length of three packets. We simulated the transmission of the test sequences over 15 channel realizations for each considered PLR value [0.1 0.4 1 3 5 10].

In tabs. 3.1, 3.2, 3.3 are presented the obtained correlation coefficients between the estimated channel distortions and the real ones with three different granularity levels (macroblock, frame, sequence). Four different estimations are presented. In NORM pure estimation all bitstream information are used, while in the other three cases NORM is ran using different motion vectors and residuals estimations. In particular motion vectors are estimated using H.264/AVC algorithm for block motion estimation changing the used motion estimation modes and rate distortion optimization (RDO). Motion estimation modes specify the search pattern used to find the best matching block. A full search it may for example be used in the sense that all the candidates are tested. Obviously this approach is quite

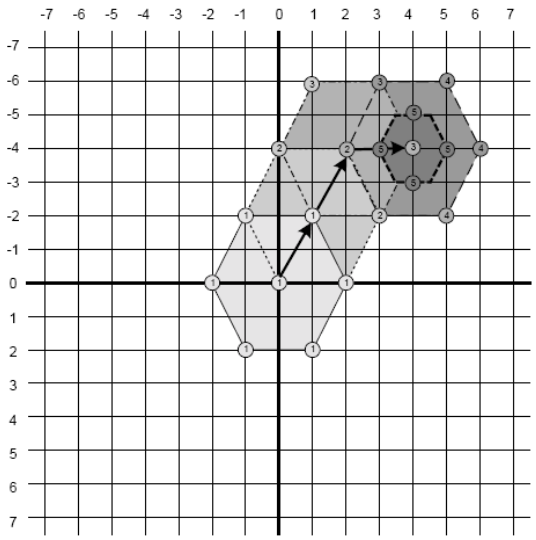

Figure 3.1: Hexagon search pattern

computationally intensive. Generally only a subset of all the possible candidates are taken into account. The subsets are generated thanks to search patterns.

We chose to run Hexagon search and the Simplified Hexagon search algorithms. Hexagon search pattern starts from the candidate selected by the motion copy vector (mvx=0, mvy=0) and tries out all the possible blocks centered on the hexagon vertices till the best prediction is found as depicted in Figure 3.1. Simplified Hexagon has a similar pattern search but changes the decision mode to select the best matching block, achieving faster but poorer performances.

On the other hand the RDO acts over the choice of the best predictor taking into account its distortion and the amount of data required to encode it. Two settings are used during our tests. In one case no RDO optimization is used and the predictor is chosen wrt its distortion. In the second case RDO optimization is active and so the best predictor is also selected taking into account its rate.

It is possible to notice in tabs. 3.1, 3.2, 3.3 how the obtained correlation coefficients at frame level are above 0.94 in all cases. In particular the best predictions are achieved with Hexagon search pattern and active RDO. This is an expected result since the tested sequences are coded with active RDO optimization. It is also possible to notice how the change of the particular search pattern do not affect too much the channel distortion estimation, since the results obtained with

	Hex - RDO Off			SHex - RDO Off			Hex - RDO ON			NORM		
PLR [%]	MB	Frm	Seq	MB	Frm	Seq	MB	Frm	Seq	MB	Frm	Seq
0.1	0.91	0.99	0.99	0.87	0.99	0.99	0.92	0.99	0.99	0.92	0.99	0.99
0.4	0.86	0.97	0.97	0.86	0.97	0.97	0.87	0.97	0.97	0.87	0.97	0.97
1	0.91	0.98	0.98	0.90	0.98	0.98	0.91	0.98	0.98	0.91	0.98	0.98
3	0.90	0.98	0.99	0.89	0.98	0.99	0.91	0.99	0.99	0.92	0.99	0.99
5	0.91	0.99	0.99	0.91	0.99	0.99	0.92	0.99	0.99	0.92	0.99	0.99
10	0.89	0.97	0.97	0.88	0.96	0.97	0.90	0.97	0.98	0.90	0.97	0.98

Table 3.1: Correlation coefficients with different motion vectors estimations between real and estimated channel distortion for Mobile Sequence

	Hex - RDO Off			SHex - RDO Off			Hex - RDO ON			NORM		
PLR [%]	MB	Frm	Seq	MB	Frm	Seq	MB	Frm	Seq	MB	Frm	Seq
0.1	0.69	0.96	0.97	0.66	0.96	0.97	0.76	0.96	0.97	0.77	0.97	0.97
0.4	0.71	0.94	0.94	0.73	0.94	0.94	0.71	0.94	0.94	0.82	0.94	0.96
1	0.84	0.96	0.96	0.84	0.96	0.96	0.84	0.96	0.96	0.84	0.96	0.96
3	0.79	0.96	0.96	0.79	0.96	0.96	0.80	0.96	0.96	0.81	0.96	0.97
5	0.82	0.95	0.96	0.82	0.95	0.96	0.81	0.95	0.96	0.84	0.96	0.97
10	0.82	0.93	0.93	0.81	0.93	0.93	0.82	0.93	0.93	0.85	0.94	0.93

Table 3.2: Correlation coefficients with different motion vectors estimations between real and estimated channel distortion for Foreman Sequence

	Hex - RDO Off			SHex - RDO Off			Hex - RDO ON			NORM		
PLR [%]	MB	Frm	Seq	MB	Frm	Seq	MB	Frm	Seq	MB	Frm	Seq
0.1	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
0.4	0.87	0.97	0.97	0.88	0.98	0.98	0.87	0.97	0.97	0.89	0.98	0.98
1	0.94	0.99	0.99	0.93	0.99	0.99	0.93	0.99	0.99	0.94	0.99	0.99
3	0.93	0.98	0.99	0.93	0.98	0.99	0.94	0.98	0.99	0.94	0.98	0.99
5	0.87	0.96	0.96	0.87	0.96	0.96	0.89	0.96	0.96	0.88	0.97	0.97
10	0.92	0.97	0.98	0.92	0.97	0.98	0.92	0.97	0.98	0.92	0.97	0.98

Table 3.3: Correlation coefficients with different motion vectors estimations between real and estimated channel distortion for News Sequence

	Hex - RDO Off			SHex - RDO Off			Hex - RDO ON		
PLR [%]	MB	Frm	Seq	MB	Frm	Seq	MB	Frm	Seq
0.1	0.95	0.99	0.99	0.92	0.99	0.99	0.97	0.99	0.99
0.4	0.92	0.99	0.99	0.92	0.99	0.99	0.92	0.98	0.98
1	0.95	0.98	0.98	0.95	0.98	0.98	0.95	0.98	0.98
3	0.95	0.99	0.99	0.95	0.99	0.99	0.97	0.99	0.99
5	0.96	0.99	0.99	0.96	0.99	0.99	0.96	0.99	0.99
10	0.96	0.99	0.99	0.96	0.99	0.99	0.97	0.99	0.99

Table 3.4: Corr. coeff. between estimated channel distortion with bitstream-NORM and NORM feed with different motion vectors estimations for Mobile

	Hex - RDO Off			SHex - RDO Off			Hex - RDO ON		
PLR [%]	MB	Frm	Seq	MB	Frm	Seq	MB	Frm	Seq
0.1	0.85	0.97	0.98	0.82	0.97	0.98	0.89	0.97	0.98
0.4	0.86	0.94	0.94	0.87	0.95	0.95	0.84	0.97	0.97
1	0.94	0.98	0.98	0.94	0.98	0.99	0.93	0.98	0.99
3	0.94	0.98	0.98	0.94	0.98	0.98	0.93	0.98	0.98
5	0.94	0.98	0.99	0.94	0.98	0.99	0.91	0.98	0.99
10	0.94	0.98	0.98	0.93	0.98	0.98	0.93	0.98	0.98

Table 3.5: Corr. coeff. between estimated channel distortion with bitstream-NORM and NORM feed with different motion vectors estimations for Foreman

	Hex - RDO Off			SHex - RDO Off			Hex - RDO ON		
PLR [%]	MB	Frm	Seq	MB	Frm	Seq	MB	Frm	Seq
0.1	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
0.4	0.95	0.98	0.98	0.96	0.99	0.99	0.95	0.98	0.98
1	0.98	0.99	0.99	0.98	0.99	0.99	0.99	0.99	0.99
3	0.98	0.99	0.99	0.98	0.99	0.99	0.99	0.99	0.99
5	0.98	0.99	0.99	0.98	0.99	0.99	0.98	0.99	0.99
10	0.98	0.99	0.99	0.98	0.99	0.99	0.99	0.99	0.99

Table 3.6: Corr. coeff. between estimated channel distortion with bitstream-NORM and NORM feed with different motion vectors estimations for News

Hexagon and simplified Hexagon search are similar. Finally in tabs. 3.4, 3.5, 3.6 all the channel distortion estimations obtained feeding NORM with estimated motion vectors and residuals are correlated to the results obtained by NORM fed with bitstream information. In all cases the correlation coefficients at frame level are always above 0.97, which means that the motion vectors and residuals estimations do not affect significantly NORM performances.

Since all performed motion vectors and residuals estimation have high correlation wrt NORM we decided to use the ones obtained with Hexagon search pattern and no RDO optimization, to avoid a loss of generality.

3.2 Structure of Group Of Pictures

Structure of GOP defines the order of the different types of frames (I, P, B). In our tests for example the defined structure of GOP is (I BB P BB P BB P BB P BB I). So I frames substitutes the P frames every 5 frames, while each P frame two B frames are inserted.

First of all the periodicity of B frame is estimated thanks to a QP estimator described as defined in Tagliasacchi and Tubaro (n.d.). Then the periodicity of I frame is computed.

The ran motion estimation algorithm produces also motion vectors for I frames with the associated residuals. Since the I frames are intra coded the number of macroblocks with a residuals' energy larger then a given threshold, is bigger wrt P frames.

For each frame prediction residuals' energy E_{res} at macroblock 16x16 level is computed. For each obtained map the number of macroblocks with an E_{res} higher then a given threshold is then computed. In Figure 3.2 is possible to notice that I frames produce a peak in the calculated function. The I periodicity is so computed searching for the frequency of the greatest peak in the autocorrelation spectrum of the given function, (Figure 3.3). The experimental results confirm that the estimated GOP structure is always as expected.

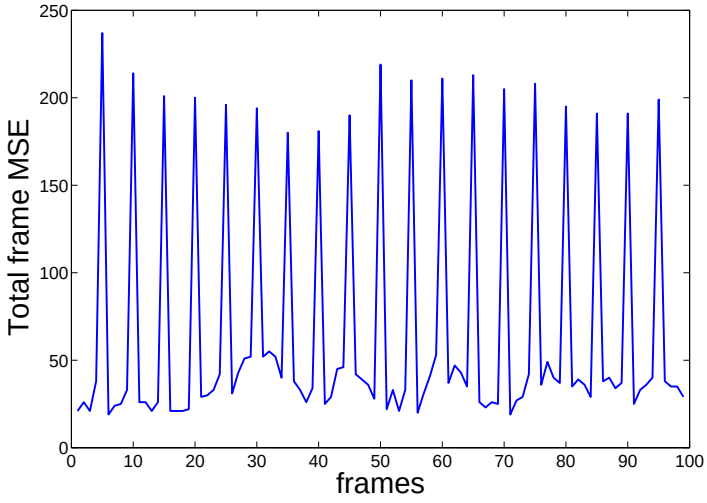

Figure 3.2: Total MSE of inter prediction residuals over each frame

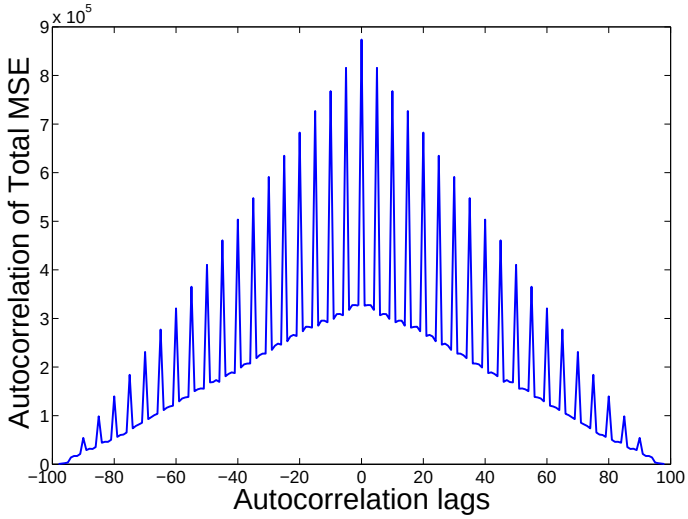

Figure 3.3: Autocorrelation of the function in Figure 3.2

3.3 Coding mode of each Macroblock

Coding mode of each macroblock are used by NORM since in P frame it is possible to have intra predicted macroblock. While there is no problem when intra macroblocks in P frame are lost, since the temporal concealment is used, a correctly received intra macroblock in a P frame does not account for temporal drift, since it is not motion predicted.

However as stated in Naccari et al. (2009) the percentage of intra predicted macroblocks in P frames is around the 4% . It is so possible to approximate this behavior considering all macroblocks in P frames as inter predicted. The obtained experimental results confirms that this approximation leads to high correlation wrt NORM performed using real coding modes extracted form the bitstream.

3.4 Map of Lost Macroblocks

In this section we describe the impact of map of lost macroblocks estimation, demonstrating that errors done during this estimate have an higher impact over channel distortion estimation wrt errors done in motion vectors and residuals estimation.

Channel distortion for each macroblock can be modeled as:

$$D^i(t) = \alpha^i(t)D^i_C(t) + \beta^i(t)D^i_{MC_t}(t-1) \tag{3.1}$$

where $D^i(t)$ is the channel induced distortion at frame t, $D^i_C(t)$ is the distortion introduced at time t by a channel loss, $\alpha(t)^i$ is the map of lost macroblock at time t, $\beta^i(t)$ is the transmission coefficient of distortion from frame $t-1$ to frame t due to predictive nature of coding and finally $D^i_{MC}(t-1)$ is the motion compensated distortion of previous frame at time $t-1$ that flows in t due to drift propagation. It must be noticed that $D^i_{MC}(t-1)$ can be rewritten as:

$$D^i_{MC}(t-1) = \alpha^i(t-1)D^i_C(t-1) + \beta^i(t-1)D^i_{MC_{t-1}}(t-2) \tag{3.2}$$

and so the eq. 3.1 can be redefined as:

$$D^i(t) = \alpha^i(t)D^i_C(t) + \alpha^i(t-1)\beta^i D^i_{MC_t}(t-1) + \beta^i\beta^i(t-1)D^i_{MC_tMC_{t-1}}(t-2) \tag{3.3}$$

that can be generalized as:

$$D^i(t) = \sum_{f=1}^{t} \alpha(f)^i\beta(f)^{i(t-f)}D^i_{MC_{t-f}}(f) \tag{3.4}$$

roughly speaking this equation states that the channel induced distortion $D^i(t)$ of a i^{th} macroblock at time t is the sum of all the distortions due to concealment that are forward propagated due to the predictive nature of the coding tool.

The estimated map $\alpha(t)^i$ can be affected by false positives, macroblocks tagged as lost but correctly received, and false negatives, lost macroblocks tagged as received. A false positive at time t introduces an over estimation of the channel distortion at time t equal to $D^i_C(t)$, so all the distortion $D^i(t+n)$ are overestimated of a factor equal to $\beta(f)^{i(n)}D^i_{MC_n}(t)$. Trivially a false negative produces an underestimation with the same values. It must be noticed that errors in map of lost macroblocks estimation at certain time t leads not only to wrong channel distortion estimation at time t, but also to a wrong estimation of the drift tail generated by the loss under exam at time $\tau > t$.

We want now to study the impact on distortion estimation of errors in map of lost macroblocks estimation wrt errors in motion vectors and residuals estimation. First of all we analyze a macroblock i^{th} which does not suffer from distortion due to drift:

$$D^i(t) = \alpha^i(t)D^i_C(t) \tag{3.5}$$

Algorithm like NORM give an estimation $\hat{D}^i(t)$ of the distortion that can be expressed as:

$$\hat{D}^i(t) = \alpha^i(t)\hat{D}^i_C(t, MV, Res) \tag{3.6}$$

$\hat{D}^i_C(t, MV, Res)$ represents the estimation of $D^i_C(t)$ obtained using motion vectors and residuals information. Equation 3.6 can be rewritten as follows when motion

vectors and residuals are estimated:

$$\hat{D}^i_{\hat{MV}-\hat{R}}(t) = \alpha^i(t)\hat{D}^i_C(t,\hat{MV},\hat{Res}) \tag{3.7}$$

this is the case of results obtained in Section 3.1, where estimated motion vectors and residuals are used to ran NORM and obtain an estimated distortion $\hat{D}^i_{\hat{MV}-\hat{R}}(t)$. It is important to notice that $\hat{D}^i_C(t,\hat{MV},\hat{Res})$ can also be rewritten as:

$$\hat{D}^i_C(t,\hat{MV},\hat{Res}) = \hat{D}^i_C(t,MV,Res) + e_{\hat{MV}-\hat{R}}(t) \tag{3.8}$$

where $e_{\hat{MV-R}}(t)$ represents the error committed in distortion estimation due to motion vectors and residuals estimate. Since also the map of lost macroblocks need to be estimated is always possible to write:

$$\hat{D}^i_{\hat{\alpha}}(t) = \hat{\alpha}^i(t)\hat{D}^i_C(t,MV,Res) \tag{3.9}$$

which represents the estimated distortion with an estimated map of lost macroblocks.

It is so now possible to show how distortion errors due to map of lost macroblock estimation are always bigger wrt the ones due to motion vectors and residuals estimate for a chosen macroblock *ith*. In particular two cases must be handled, $\hat{\alpha}^i$ representing a false positive (FP) or $\hat{\alpha}^i$ representing a false negative (FN).

In the first case (FP) $\alpha^i(t) = 0$ but $\hat{\alpha}^i(t) = 1$ so the real distortion is correctly estimated from $D^i_C(t,\hat{MV},\hat{Res})$ and no errors are introduced. On the other hand $\hat{D}^i_{\hat{\alpha}}(t) = \hat{D}^i_C(t,MV,Res)$, which always introduces an estimation error greater than the one introduced by motion vectors and residuals.

In the second case (FN) $\alpha^i(t) = 1$ but $\hat{\alpha}^i(t) = 0$, the estimated motion vectors and residuals introduce an error equal to $e_{\hat{MV}-\hat{R}}(t)$, while the FN in $\alpha^i(t)$ leads to an error in absolute value equal to $\hat{D}^i_C(t,MV,Res)$. Since from results in 3.1 we can assure that $\hat{D}^i_C(t,\hat{MV},\hat{Res})$ is a good predictor of $\hat{D}^i_C(t,MV,Res)$ is also possible to write that $\hat{D}^i_C(t,MV,Res) >> e_{\hat{MV}-\hat{R}}(t)$. Which finally demonstrates that the

estimation of map of lost macroblocks is the most sensible part among all the estimations.

Moreover it must be remembered that while estimation errors in motion vectors and residuals estimation are localized in time and do not affect future results, errors in map of lost macroblocks estimation always lead to a drift of the error that affects a certain number of frames after the burst, giving birth to an higher error energy.

Obliviously different FPs and FNs lead to different estimation errors. We want to understand which FPs and FNs could be negligible wrt NORM estimation. We know that NORM calculates $\hat{D}^i_C(t, MV, Res)$ in two different ways depending from the type of used concealment, we will analyze them separately.

3.4.1 Temporal Concealment

We first analyze the weight of FPs and FNs when $\hat{D}^i_C(t, MV, Res)$ is expressed as in eq. 2.8. In this case $\hat{D}^i_C = \hat{D}^i_{TC}$ and represents the distortion due to temporal concealment.

We now want to quantify the specific $\hat{D}^{i,TC}_n$ estimation error when a FN or FP is present, wrt the particular features of the macroblock under consideration.

As already said $\hat{D}^{i,TC}_n$ is computed as the sum of two different contributes eq. 2.8 The first term models the distortion due to motion vectors lack as written in eq. 2.9. In particular $\Phi^i_n(w_j, w_k)$ denotes the power spectral density of the spatial predictor $\hat{P}^i_n(w)$ and it is computed as:

$$\Phi^i_n(w_j, w_k) = |\frac{1}{B^2} \sum_{x=0}^{B-1} \sum_{y=0}^{B-1} \hat{P}^i_n(w) e^{-j(w_j x + w_k y)}|^2 \tag{3.10}$$

while δ_x, δ_y are the differences between the real motion vector and the concealed one:

$$\delta_{x|y} = |\vec{v}^i_{n,x|y} - \tilde{v}^i_{n,x|y}| \tag{3.11}$$

We now want to understand for which features $\hat{D}^{i,MV}_n$ is near to zero or small enough (ε) wrt the general distortion. First of all we analyze the second term of

the expression:

$$(1 - cos(w_j\delta_x + w_k\delta_y)) \rightarrow \varepsilon$$

$$cos(w_j\delta_x + w_k\delta_y) \rightarrow 1$$

$$(w_j\delta_x + w_k\delta_y) \rightarrow 0$$

this term tends to zero if and only if $w_j\delta_x \rightarrow 0$ and $w_k\delta_y \rightarrow 0$. We can so study two different situations:

- $\delta_x \rightarrow 0$ and $\delta_y \rightarrow 0$
- $w_j \rightarrow 0$ and $w_k \rightarrow 0$

In the first case we know that $\hat{\delta}_n^i$ is estimated as in eq. 2.10 which can be rewritten as the expected value of the mean square error between the concealed motion vector $\tilde{v}_n^i$ and all the neighbors motion vectors v_{SxS}^l:

$$\hat{\delta}_n^{i2} = E[(\tilde{v}_n^i - v_{SxS}^l)] \tag{3.12}$$

So when $\delta_x \rightarrow 0$ and $\delta_y \rightarrow 0$ also $E[(\tilde{v}_n^i - v_{SxS}^l)] \rightarrow 0$.

We can so conclude that, if a macroblock is placed in a zone with a simple uniform motion, its $E[(\tilde{v}_n^i - v_{SxS}^l)] \rightarrow 0$ *and also its estimated distortion* $D_n^{i,MV} \rightarrow 0$.

The second case of interest is the one related to the spatial frequency $w_j \rightarrow 0$ and $w_k \rightarrow 0$. As already said for low frequencies $(1 - cos(w_j\delta_x + w_k\delta_y)) \rightarrow \varepsilon$ while it grows up as the frequencies become higher. Since the PSD $\Phi_n^i(w_j, w_k)$ works as a weight for $(1 - cos(w_j\delta_x + w_k\delta_y))$, we can argue that, if the PSD lays in the low frequency range:

$$\Phi_n^i(w_j, w_k)(1 - cos(w_j\delta_x + w_k\delta_y)) \rightarrow \varepsilon$$

Summarizing, for macroblock with a simple texture whose PSD $\Phi_n^i(w_j, w_k)$ *lays in the low frequency range, the estimated distortion* $D_n^{i,MV} \rightarrow 0$.

We now focus our attention over the second term of $\hat{D}_n^{i,TC} = \hat{D}_n^{i,MV} + \hat{D}_n^{i,PR}$ which models the distortion due to the lack of real prediction residuals computed as in eq.2.11 where Θ_{n-r}^i represents the prediction residuals of the macroblock in

the reference frame pointed by the motion vector. We can also rewrite the estimated distortion as:

$$\hat{D}_n^{i,PR} = E[(\Theta_{n-r}^i)^2] \tag{3.13}$$

so it is trivial to write that $\hat{D}_n^{i,PR} \to \varepsilon$ when $E[(\Theta_{n-r}^i)^2] \to \varepsilon$. The dimension of the prediction residuals variance depends from the features of the particular coded macroblock. If the variance is small the macroblock has simple texture and lays in a zone with a uniform simple motion so $E[(\Theta_{n-r}^i)^2] \to \varepsilon$, since it is easier to create a predictor which is a good approximation of the real macroblock. On the other hand, a big variance is linked to macroblock with high texture that lays in a zone with a chaotic motion. In this case much more information is contained in the prediction residuals, and so $E[(\Theta_{n-r}^i)^2]$ grows higher.

In conclusion, if the coded macroblock has simple texture and lays in a zone with a uniform simple motion, its $E[(\Theta_{n-r}^i)^2] \to \varepsilon$ *and so the estimated distortion* $D_n^{i,PR} \to 0$.

Matching up all the results we can argue that zones with a simple texture and simple and uniform motion produce a $\hat{D}_n^{i,TC}$ which is smaller wrt the ones computed in different zones. These results are coherent wrt the concealment algorithm, which works obviously better in this kind of regions. So if an FP or FN falls in these zones its impact over the total estimated distortion is smaller wrt FPs or FNs fallen in different areas.

3.4.2 Spatial Concealment

We now analyze the weight of FPs and FNs when $\hat{D}_C^i(t, MV, Res)$ is expressed as in eq. 2.7. In this case $\hat{D}_C^i = \hat{D}_{SC}^i$ and represents the distortion due to spatial concealment.

From eq. 2.7 we can argue that $D_{SC}^i \to \varepsilon$ when $(\tilde{M}_n^i(x,y) - \tilde{M}_n^{i,0}(x,y)) \to \varepsilon$, which can be rewritten as $\tilde{M}_n^i(x,y) \to \tilde{M}_n^{i,0}(x,y)$. So the spatial concealment distortion is negligible only when the lost macroblock is similar to the co-located macroblock in the previous frame, which may happens when there is no motion and the texture of the macroblock is easy predictable from the spatial conceal.

Also in this case we can finally argue that FPs and FNs produce a negligible

distortion only if they fall in areas with no motion and a very simple texture that can be easily reconstructed by the spatial concealment algorithm.

It must be noticed that in the modern network PLR never exceeds the 10%, so the number of true negatives is much more bigger than the one of true positives ($TN >> TP$). If we suppose to have the same probability for FPs or FNs, we have much more FPs than FNs, which means a greater contribution in total estimated distortion by the FPs. *So it's more likely to set up the estimation algorithm for the map of lost macroblocks with a greater probability for FNs than for FPs in regions with a simple texture and simple and uniform motion.*

CHAPTER 4

MAP OF LOST MACROBLOCKS ESTIMATION

This chapter is dedicated to the map of lost macroblocks (MLM) estimation problem. As already described the decoder recovers the lost macroblocks running a particular concealment algorithm. In our scenario we suppose to not know which algorithm is ran. However in Section 1.2 we identified several mild assumptions that are valid for a large class of concealment algorithms. Our objective is to use these assumptions to find macroblocks properties whose values can be due to a reconstruction process, e.g. residuals' energy, motion vectors, boundaries discontinuity. Roughly speaking we want to extract for each macroblock belonging to a corrupted sequence a set of features that are indexes of the macroblock status, lost or correctly received. For example, in our scenario, the temporal concealment recovers a lost block by coping a macroblock from a previous reference frame. So for lost temporal concealed macroblocks we can argue that the temporal prediction residuals' energy will be zero.

Once this set of features has been found we want to estimate the posterior probability that a i^{th} macroblock is lost observing the identified set of features:

$$P^i(L|\mathbf{f}) \tag{4.1}$$

where L indicates that the macroblock is lost and $\overline{L}$ that the block is correctly

received. On the other hand $\mathbf{f}$ is a vector of features, where each $\mathbf{f}_j$ feature is extracted from the corrupted video and is linked to the particular status of the macroblock. So each i^{th} macroblock will be classified thanks to the estimated posterior $P^i(L|\mathbf{f})$. To estimate the posterior $P^i(L|\mathbf{f})$ we rewrite it using the Bayes theorem:

$$P^i(L|\mathbf{f}) = \frac{P^i(\mathbf{f}|L) \cdot P^i(L)}{P^i(\mathbf{f})} \tag{4.2}$$

where the likelihood $P^i(\mathbf{f}|L)$ is the conditional probability to have a certain set of features knowing that the macroblock is lost, the prior $P^i(L)$ is the probability to have a lost macroblock and finally $P^i(\mathbf{f})$ is a normalizing constant that indicates the probability of certain set of features $\mathbf{f}$. So to solve the prior estimation problem we decide to estimate the likelihood and the prior from the corrupted video:

- Estimated likelihood: $\hat{P}^i(\mathbf{f}|L)$
- Estimated prior: $\hat{P}^i(L)$

it must be noticed that we do not estimate $P^i(\mathbf{f})$ since it works as a normalizing term and can be neglected for our purposes. With the estimated prior and likelihood we will be so able to create a probability map that gives for each macroblock the probability to be lost observing the chosen features, solving the MLM estimation problem.

However in the following section we will see that it is not possible to solve the described estimation problem since it is not unusual that the concealment perfectly restores the lost macroblocks. We will so redefine the MLM estimation problem with a new posterior $P^i(BC|\mathbf{f})$, which is the probability for the i^{th} macroblock to be badly concealed observing the chosen set of features. Then we will estimate the likelihood and the prior for the new defined posterior, obtaining an estimation of $P^i(BC|\mathbf{f})$. Finally we will introduce the Markov random fields model to take advantage from the spatial relationship between badly concealed macroblocks and we solve the related maximum a posterior (MAP) problem to obtain the boolean map of lost macroblocks using a Min-Cut/Max-Flow algorithm.

4.1 Concealment Effectiveness

The objective of concealment is to restore as much as possible lost information. Two different types of concealment are used in I and P frames, spatial and temporal. In both cases the algorithms try to recreate the lost macroblocks using the information available at the decoder side. It must be noticed that, since the concealment algorithm takes advantage of the received or already restored information of the neighborhood, the quality of the reconstruction strictly depends from the characteristics of the neighborhood and of the macroblock itself.

It may happen that concealment is able to perfectly restore the lost macroblock:

$$\mathbf{M}^i(x,y) = \tilde{\mathbf{M}}^i(x,y) \tag{4.3}$$

where $\mathbf{M}^i(x,y)$ is the *BxB* macroblock reconstructed at the decoder side with no channel losses and $\tilde{\mathbf{M}}^i(x,y)$ is the *BxB* macroblock reconstructed by concealment at the decoder side when channel losses occurred. The induced channel distortion of the i^{th} macroblock is so perfectly zero:

$$D^i = \frac{1}{B^2}\sum_{x=1}^{B}\sum_{y=1}^{B}(\mathbf{M}^i(x,y) - \tilde{\mathbf{M}}^i(x,y))^2 = 0 \tag{4.4}$$

In this case likelihood $P^i(\mathbf{f}|L)$ estimation problem is ill posed, since there is no way to distinguish the lost macroblocks from the reconstructed ones. As an example we study the particular concealment algorithms implemented in the H.264/AVC reference software, which are the ones performed during our tests.

First we analyze the performances of the temporal concealment applied in P frames, searching for cases in which the concealment is able to reconstruct lost macroblocks perfectly:

$$\mathbf{M}^i(x,y) = \tilde{\mathbf{M}}^i_T(x,y) \tag{4.5}$$

where $\tilde{M}^i_T(x,y)$ is the *BxB* macroblock reconstructed by temporal concealment. Intuitively features that affect the concealment performances are the motion and texture of the area in which $\tilde{M}^i_T(x,y)$ lays.

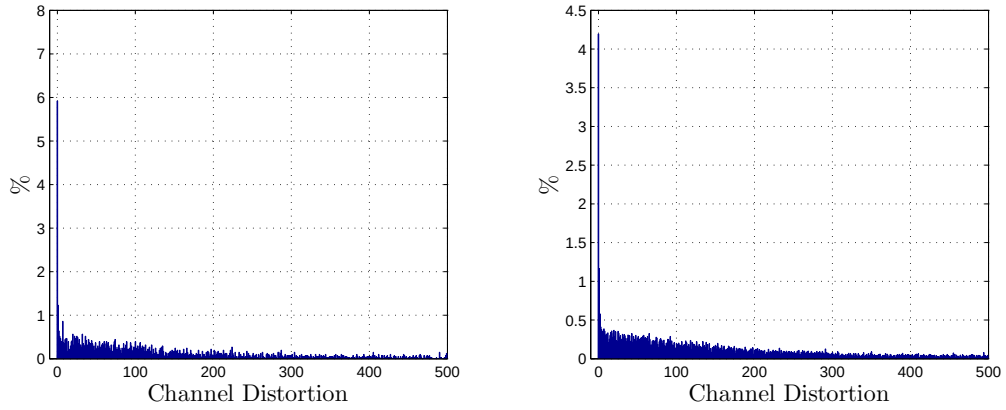

Figure 4.1: Distribution of channel distortion of lost macroblocks reconstructed with temporal concealment for the *Mobile* sequence.

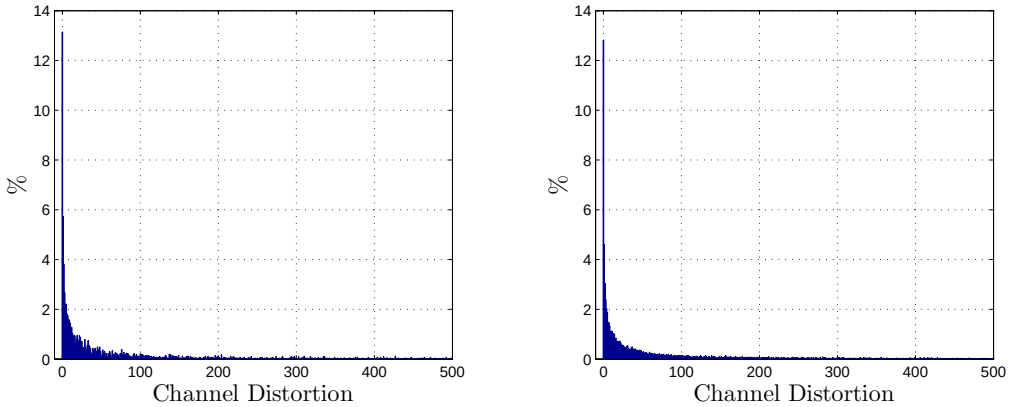

Figure 4.2: Distribution of channel distortion of lost macroblocks reconstructed with temporal concealment for the *Foreman* sequence.

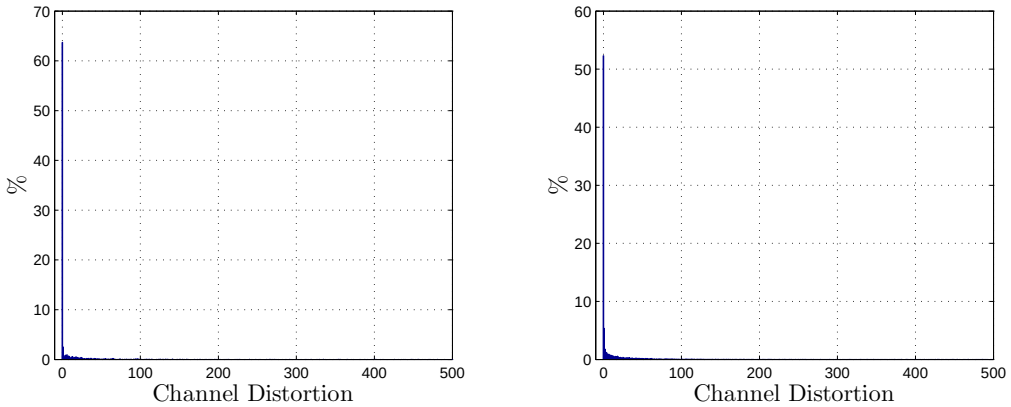

Figure 4.3: Distribution of channel distortion of lost macroblocks reconstructed with temporal concealment for the *News* sequence.

Figure 4.4: On the left the original noiseless frame with the lost slices in pink. On the right the corrupted and concealed frame with blue macroblocks perfectly restored. The perfectly restored zones have uniform zero motion. (Induced channel distortion equals to zero)

Figure 4.5: On the left the original noiseless frame with the lost slices in pink. On the right the corrupted and concealed frame with blue macroblocks perfectly restored. (Induced channel distortion equals to zero). The perfectly restored zones have uniform motion.

Figure 4.6: On the left the original noiseless frame with the lost slices in pink. On the right the corrupted and concealed frame with blue macroblocks perfectly restored. (Induced channel distortion equals to zero). The perfectly restored zones are "Flat".

Two cases in which is more probable that nearly perfect reconstruction may be reached can be identified:

- The motion of the area is uniform: all the motion vectors point to the same direction.
- The texture of the area is simple: its spectrum is localized in low frequencies range.

The first case is shown in Figure 4.4, it is possible to notice that a portion of the lost slice is perfectly reconstructed. In this area the motion in nearly uniform and near to zero. The concealment algorithm reconstructs the macroblock copying a motion vector from the neighbors, so there is an high probability that the correct motion vector is restored, since all neighbors share the same motion vector (area with uniform motion). In Figure 4.5 is depicted another example of uniform motion, with motion vectors different from zero.

Nearly perfect reconstruction is also possible in areas where the spectrum is localized in low frequency range, Figure 4.6. If the lost area is "flat", choosing an incorrect motion vector during concealment can however lead to perfect reconstruction, since all neighbor macroblocks are similar.

Histograms in Figures 4.1, 4.2, 4.3 represent the distortion D_T^i produced by temporal concealed macroblocks of three sequences, *Mobile Foreman* and *News*, at two different PLRs *(1, 5)* over 15 realizations. The percentage of macroblocks nearly perfectly reconstructed is higher in *News* wrt *Mobile* and *Foreman* sequences, due to uniform motion areas and "flat" macroblocks.

Performances for spatial concealment are then evaluated. Also in this case we are searching for perfectly reconstructed macroblocks:

$$\mathbf{M}^i(x,y) = \tilde{\mathbf{M}}_S^i(x,y) \tag{4.6}$$

where $\tilde{M}_S^i(x,y)$ is the *BxB* macroblock reconstructed by spatial concealment. Intuitively also in this case zones without high spatial frequencies are the ones that can reach lower channel distortion.

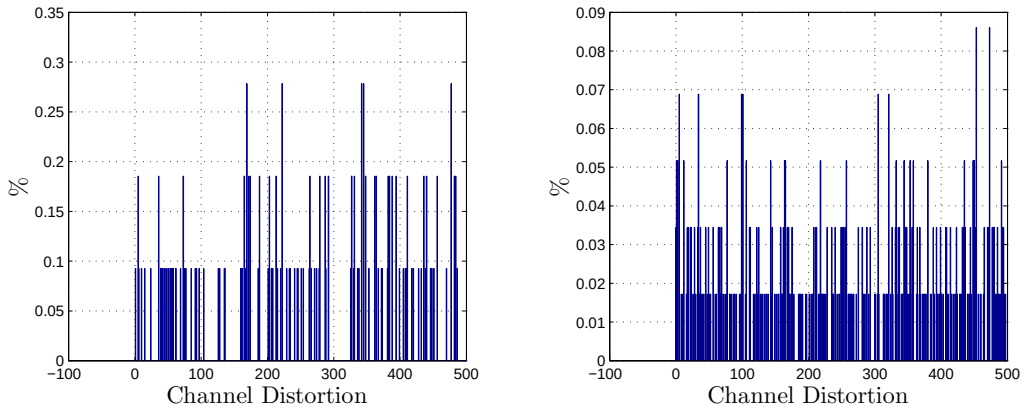

Figure 4.7: Distribution of channel distortion of lost macroblocks reconstructed with spatial concealment for the *Mobile* sequence.

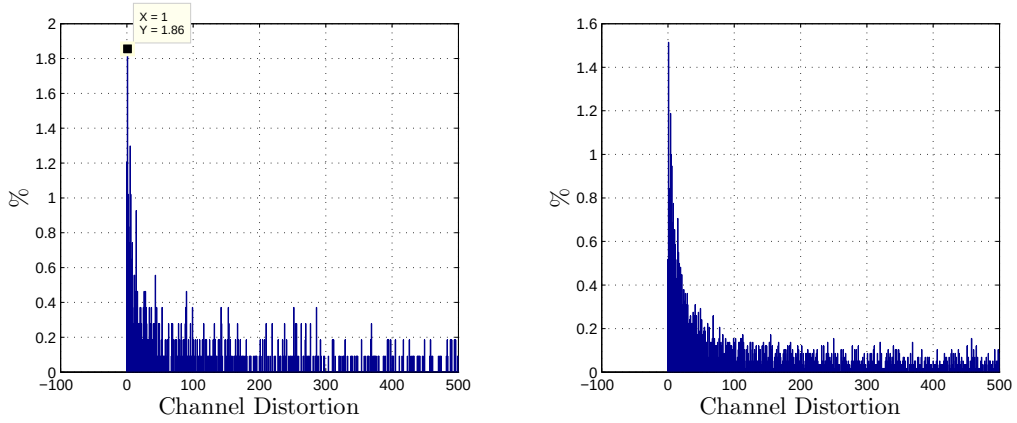

Figure 4.8: Distribution of channel distortion of lost macroblocks reconstructed with spatial concealment for the *Foreman* sequence.

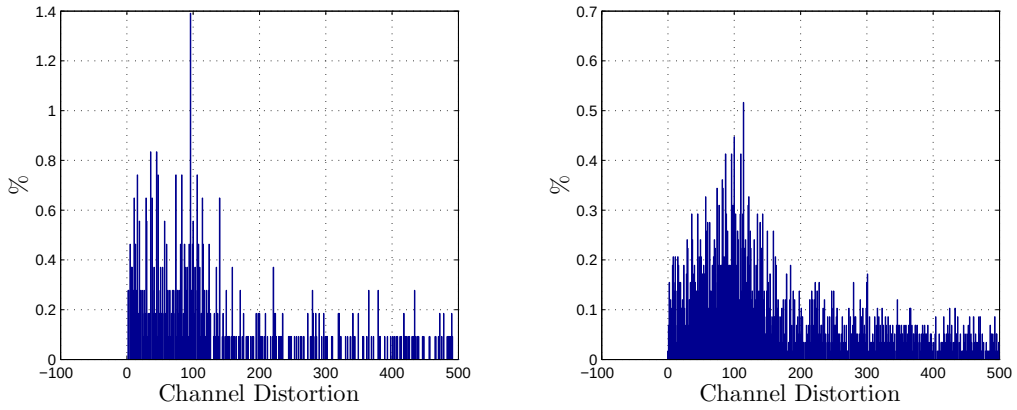

Figure 4.9: Distribution of channel distortion of lost macroblocks reconstructed with spatial concealment for the *News* sequence.

Figure 4.10: On the left the original noiseless frame with the lost slices in pink. On the right the corrupted and concealed frame with blue macroblocks nearly perfectly restored. The nearly perfectly restored zone is "Flat".

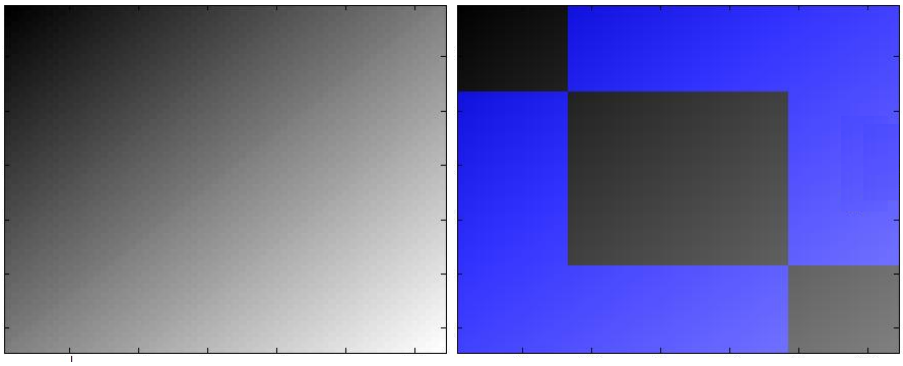

Figure 4.11: On the left the original noiseless frame with the lost slices in pink. On the right the corrupted and concealed frame with blue macroblocks perfectly restored. (Induced channel distortion equals to zero)

In Figure 4.10 is possible to appreciate that "flat" areas exhibit a better reconstruction, since lost macroblocks are more predictable from boundaries of their neighbors using the interpolating function in eq. 2.1. Moreover in Figure 4.11 it is possible to appreciate an artificial example in which perfect spatial reconstruction is achieved due to the particular texture of the image.

Histograms in Figures 4.7, 4.8, 4.9 represent the distortion D_S^i of spatially concealed macroblocks for the same dataset used for temporal concealment. In general the temporal conceal performs better than the spatial one. Moreover in *Foreman* sequence spatial concealment reaches higher performances, since its background has a texture easily spatially predictable.

From above considerations we understand that likelihood $P^i(\mathbf{f}|L)$ estimation problem is ill-posed. For cases in which concealed and real macroblocks are indistinguishable $\mathbf{M}^i(x,y) = \tilde{\mathbf{M}}^i(x,y)$, it is not possible to find features $\mathbf{f}$ able to discriminate between $P^i(\mathbf{f}|L)$ and $P^i(\mathbf{f}|\overline{L})$ distributions. From this consideration we are forced to redefine the likelihood as the probability of the i^{th} macroblock to have a certain set of features $\mathbf{f}$ observing that the macroblock is badly concealed $P^i(\mathbf{f}|BC)$. Roughly speaking we search for a set of features $\mathbf{f}$ discriminating between poorly concealed macroblocks, that produce an appreciable channel distortion, and all the other blocks. Fortunately these limitations do not affect our final objective since we are interested in channel distortion estimation and not detectable cases are the ones with channel distortion D^i equal to zero.

4.2 New Posterior Estimate

In previous section a new likelihood was introduced to overcome the problem of perfectly or nearly perfectly concealed macroblocks. The prior estimation problem must be so redefined to fit the new likelihood. In particular we are interested in estimating the probability of the i^{th} macroblock to be badly concealed observing a set of features $\mathbf{f}$:

$$P^i(BC|\mathbf{f}) \tag{4.7}$$

where BC indicates that the macroblock is badly concealed and $\overline{BC}$ that the block

is correctly received or concealed without visual impairments. Roughly speaking this probability states that we are interested in searching concealed macroblocks that introduce visual impairments in the decoded video.

Thanks to Bayes theorem we can rewrite posterior $P^i(BC|\mathbf{f})$) as:

$$P^i(BC|\mathbf{f}) = \frac{P^i(\mathbf{f}|BC) \cdot P^i(BC)}{P^i(\mathbf{f})} \tag{4.8}$$

where the likelihood $P^i(\mathbf{f}|BC)$ is the conditional probability to have a certain set of features knowing that the macroblock is badly concealed, the prior $P^i(BC)$ is the probability to have a badly concealed macroblock and finally $P^i(\mathbf{f})$ is a normalizing constant that indicates the probability of certain set of features $\mathbf{f}$.

We have so to estimate the likelihood and the prior from the corrupted video sequence:

- Estimated likelihood: $\hat{P}^i(\mathbf{f}|BC)$
- Estimated prior: $\hat{P}^i(BC)$

we neglect $P^i(\mathbf{f})$ since works as a normalizing term and it is not necessary for our purposes.

Since a temporal badly concealed macroblock differs from a spatial one two different posteriors, with their own likelihoods and priors, must be estimated for temporal and spatial concealments:

- $\hat{P}^i(BC_T|\mathbf{f}_T) \propto \hat{P}^i(\mathbf{f}|BC_T) \cdot \hat{P}^i(BC_T)$
- $\hat{P}^i(BC_S|\mathbf{f}_S) \propto \hat{P}^i(\mathbf{f}|BC_S) \cdot \hat{P}^i(BC_S)$

also different set of features $\mathbf{f}$ must be identified for temporal ($\mathbf{f}_T$) and spatial ($\mathbf{f}_S$) concealments.

In the following sections we analyze how to estimate likelihoods and priors, searching also for the best features $\mathbf{f}$ that discriminate between badly concealed and not lost or correctly recovered macroblocks.

4.3 Ground Truth

The performances of our estimation must be measured wrt a ground truth. Since we are interested in searching for badly concealed blocks *BC*, map of lost macroblocks (MLM) may be not a good ground truth for our tests, since it does not take into account the concealment effectiveness.

Our ground truth *Gt* could be so created weighting the MLM wrt the map of concealment effectiveness *Ce*, which measures the concealment capability in recreating the i^{th} macroblock with no visual impairments:

$$Gt^i = Ce^i \cdot MLM^i \tag{4.9}$$

in particular *Ce* has high values for macroblock concealed with visual impairments and low otherwise.

The concealment effectiveness map *Ce* is a function of the particular slicing and adopted concealment algorithm $Ce((Slicing),(Conc.Alg.))$. Supposing to know the real slicing pattern and the used concealment algorithm it is possible to search for the real *Ce*. However in our scenario we cannot assume to know the slicing pattern or the concealment and we have to estimate *Ce* starting from our mild assumptions for temporal and spatial concealments.

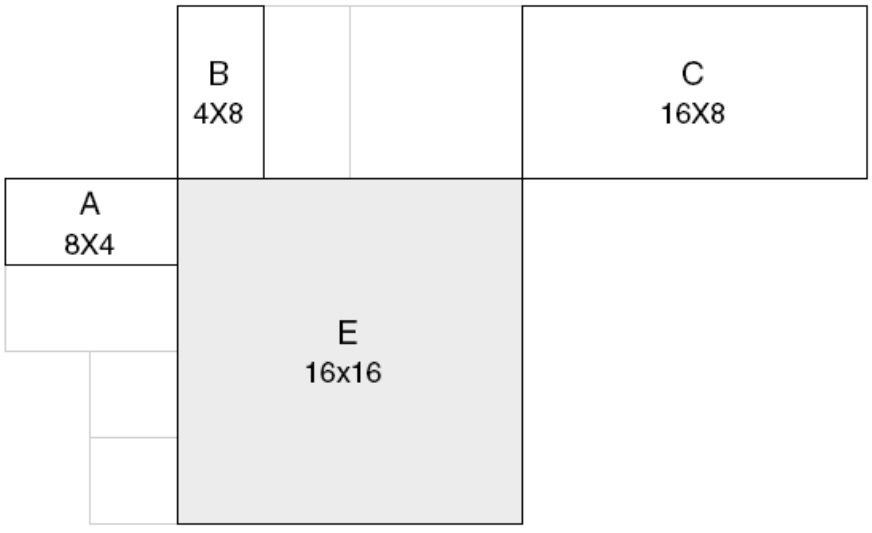

Figure 4.12: Partitions used to compute *MVP*.

For temporal concealment we decided to model Ce_T as follows:

$$\hat{C}e_T^i = D_T^i \cdot MVPD^i \tag{4.10}$$

where:

- D_T^i is the induced channel distortion of the i^{th} macroblock due to temporal concealment as described in equation 2.4
- $MVPD^i$ is the motion vector predictor difference and it is defined as the difference between the motion vector predictor MVP^i and the motion vector MV^i of the i^{th} macroblock of the noiseless sequence:

$$MVPD^i = |MVP^i - MV^i| \tag{4.11}$$

  notice that the MVP is defined as the median of the motion vectors for partitions A, B and C of Figure 4.12.

So the concealment effectiveness map Ce_T will have low values for temporal concealed macroblocks $\tilde{\mathbf{M}}_T^i(x,y)$ with no visual impairments. In particular the first term D^i accounts for cases in which a nearly perfect reconstruction is achieved. On the other hand the $MVPD^i$ models cases in which concealment restores the i^{th} macroblock with no visual impairments but with a not negligible channel distortion D^i. This may happen when the reconstructed vector preserves the relationship wrt its neighborhood but the whole lost slice suffers from a slight shift wrt the original.

Since our ground truth must be a binary map, $\hat{C}e_T^i$ must be thresholded. In particular a binary version of D_T^i and $MVPD^i$ must be defined:

$$\mathrm{B}D_T^i \begin{cases} 1 & \text{if } D_T^i > Th_D \\ 0 & \text{if } D_T^i \leq Th_D \end{cases} \tag{4.12}$$

$$\mathrm{B}MVPD^i \begin{cases} 1 & \text{if } MVPD^i > Th_{MVPD} \\ 0 & \text{if } MVPD^i \leq Th_{MVPD} \end{cases} \tag{4.13}$$

where Th_{MVPD} and Th_D are respectively thresholds for $MVPD$ and D_i. It is now possible to define a binary version of $\hat{C}e_T^i$:

$$\mathrm{B}\hat{C}e_T^i = (\mathrm{B}D_T^i) \text{ AND } (\mathrm{B}MVPD^i) \tag{4.14}$$

So the obtained binary version of concealment effectiveness map $\mathrm{B}\hat{C}e_T^i$ has values equal to zero for reconstructed macroblocks with no visual impairments and zero otherwise. In particular the two thresholds Th_{MVPD} and Th_D are experimentally chosen looking for the visually best estimation of $\mathrm{B}\hat{C}e_T^i$, setting both of them to zero. So the first thresholded term select all macroblocks where the concealment restoration works perfectly, and the second one all the macroblocks coded as skip. All the process is depicted in Figure 4.13.

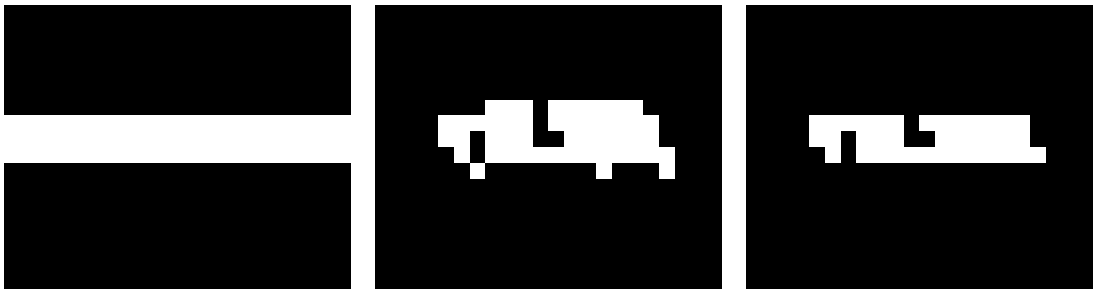

Figure 4.13: Starting form the left we have the map of lost macroblock, the estimated concealment effectiveness map $\mathrm{B}\hat{C}e_T^i$, and the obtained ground truth.

Also for spatial concealment a ground truth must be defined. However in this case a spatial concealment effectiveness map $\hat{C}e_S^i$ is not necessary, since from results obtained in Section 4.1, we know that the number of cases in which spatial concealment is able to achieve a reconstruction with no visual impairments is negligible. So the adopted ground truth for spatial concealment is the map of lost macroblock (MLM).

4.4 Likelihood Estimation

In this section we focus on the estimate of likelihoods for temporal and spatial concealments, starting from mild assumptions made in Section 1.2.

4.4.1 Temporal Likelihood Estimation

First of all we identify a feature $\mathbf{f}_{Tj}$ which is able to discriminate between temporally badly concealed and not lost or correctly recovered macroblocks, starting from the mild assumption made over the temporal concealment, that can be summarized as follows:

- Temporally concealed macroblocks must have a predictor, in a previous reference frame, with prediction residuals' energy near to zero.

First of all we define the following quantities:

- $\tilde{M}_T^i(x,y,t)$ is the temporally concealed BxB i^{th} macroblock belonging to the frame at time t.
- $v_x^i(t)$ and $v_y^i(t)$ are respectively x and y coordinates of the concealed motion vector used to temporally predict the i^{th} macroblock from a previous reference frame at time $t-n$.
- $M_T^i(x-v_x^i(t), y-v_y^i(t), t-n)$ is the temporal prediction of the macroblock $\tilde{M}_T^i(x,y,t)$

Our mild assumption over temporal concealment can be so written as:

$$\tilde{M}_T^i(x,y,t) = M_T^i(x-v_x^i(t), y-v_y^i(t), t-n) \tag{4.15}$$

which means that the residuals' energy MSE_T^i computed between the temporally concealed i^{th} macroblock and its prediction at time $t-n$ will be equal to zero:

$$MSE_T^i(t) = \frac{1}{B^2}\sum_{x=1}^{B}\sum_{y=1}^{B}(\tilde{M}_T^i(x,y,t) - M_T^i(x-v_x^i(t), y-v_y^i(t), t-n))^2 = 0 \tag{4.16}$$

So the temporal concealment simply restores a lost macroblock copying a block from a previous reference frame at time $t-n$. Moreover we know from the standard that lost macroblocks have always dimension of 16x16 pixels which means $B = 16$. Supposing so to compute temporal prediction residuals $MSE_T^i(t)$ for all the macroblocks belonging to the frame at time t, the concealed macroblocks $\tilde{M}_T^i(x,y,t)$ will have $MSE_T^i(t) = 0$. Since temporal prediction residuals' energy $MSE_T^i(t)$ is a discriminant feature wrt badly concealed macroblocks and other blocks, it can be used in our temporal likelihood estimation, setting the vector of features $\mathbf{f}_T = MSE_T^i(t)$ with size 1x1. We can so write temporal likelihood $P^i(\mathbf{f}_T|BC_T)$ as:

$$P^i(MSE_T^i|BC_T) \tag{4.17}$$

that can be so modeled as an impulse centered in MSE_T^i equal to zero.

All these considerations perfectly fit if real motion vectors $v_x^i(t)$ and $v_y^i(t)$ are available and no deblocking filter is applied. Unfortunately in our scenario the motion is estimated on the reconstructed video ($\hat{v}_x^i(t)$ and $\hat{v}_y^i(t)$) and it is possible that the predictor with the minimum residuals' energy $MSE_T^i(t)$ will not be found. Moreover with an active deblocking filter the concealed macroblocks may differ from their predictors:

$$\tilde{M}_T^i(x,y,t) \neq M_T^i(x - v_x^i(t), y - v_y^i(t), t-n) \tag{4.18}$$

and the calculated MSE_T^i may be different from zero as well. The likelihood distribution $P^i(MSE_T^i|BC_T)$ can not be so modeled as an impulse. Moreover is not possible to find a general distribution model, since it depends from the particular used motion estimation and to the used deblocking filter. However, from our experimental results, we can suppose that a good approximation of the distribution is an exponential function. As expected, adopting this function, for small MSE_T^i values we have high probabilities, while for big MSE_T^i values we have low probabilities. Roughly speaking it is more probable that a lost, and so also badly concealed, macroblock will have a small MSE_T^i value.

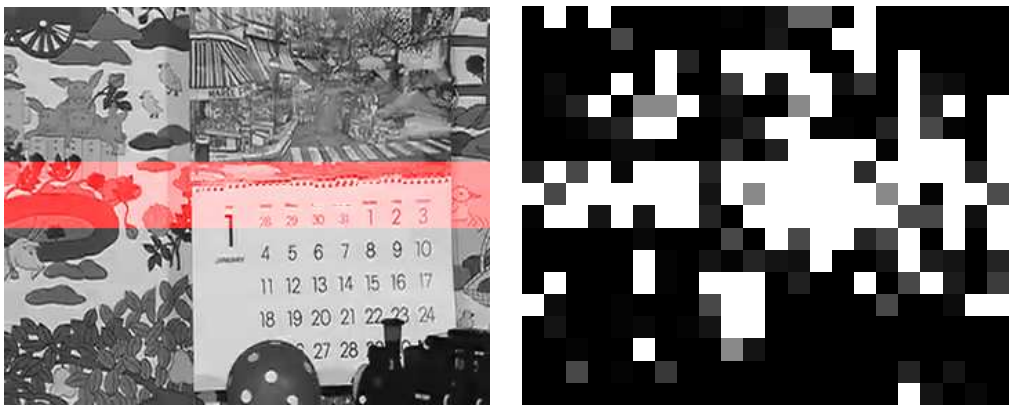

Figure 4.14: On the left the corrupted video, with the lost slices in red, on the right the estimated likelihood map

For each i^{th} macroblock belonging to a P frame of the corrupted sequence MSE_T^i is calculated. The obtained values are so mapped to probabilities using the following function:

$$\hat{P}^i(MSE_T^i|BC_T) \propto e^{\frac{MSE_T^i}{\alpha_{TL}}} \tag{4.19}$$

we so obtain an estimation of the ith macroblock temporal likelihood $P^i(MSE_T^i|BC_T)$. As already said, since we do not know the real likelihood distribution we experimental chose the decay factor α_{TL}. In Figure 4.14 is depicted the estimated likelihood for a given corrupted frame.

We now test our estimated likelihood as a binary classifier, understanding how much the chosen feature MSE_T^i is able to discriminate between temporally badly concealed macroblocks and correctly concealed or received ones wrt the constructed ground truth. We so measure the separability capabilities of MSE_T^i wrt the two defined classes, badly concealed macroblocks (Positive Class) and the correctly concealed and received macroblocks (Negative Class). To achieve this task we use the Receiving Operating Characteristics (ROC) curves.

Figure 4.15 illustrates an example of two overlapping probability density functions (PDF) describing the distribution of a feature in two classes. Suppose that the blue curve on the right is the pdf of positive class, while the red one on the left is the pdf of the negative class. We set a threshold T as a region boundary to discriminate among the two classes: in particular for $x < T$ we decide the class to

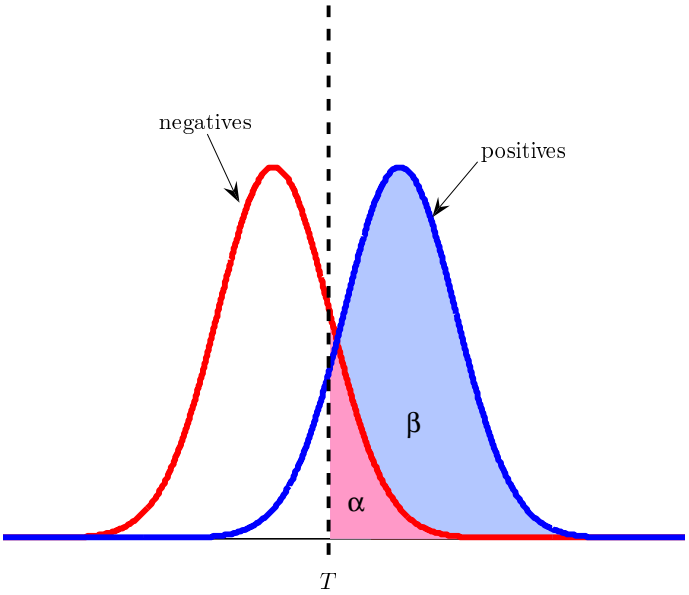

Figure 4.15: Example of overlapping pdf's of the same feature in two classes.

be negative, while for $x \geq T$ we decide the class to be positive. The shaded area indicated by β is the probability of getting a true positive given the threshold T, while the shaded area marked by the letter α is the probability of deciding that a feature value x is positive while it is actually negative (this is called false positive). Varying T over all the possible values of x, one obtains a plot of the *true positive rate* versus the *false positive rate*. This kind of plot is known as *Receiving Operating Characteristic* (ROC) curve. ROC curves have been widely used in pattern recognition and classification to evaluate and visualize the performance of a classifier. Used in the context of feature selection, ROC curves evaluate the power of discrimination of a single feature between two classes when a simple decision boundary (a threshold) is used as classifier. A more detailed discussion on ROC curves in classification can be found in Fawcett (2004).

In Figures 4.16, 4.17, 4.18 are presented the ROCs of the estimated likelihood $\hat{P}^i(MSE_T^i|BC_T)$ wrt our defined ground truth for *Mobile Foreman* and *News* sequences at different PLRS (1,5) over 15 realizations. It is possible to notice that for *News* sequence the obtained areas under curve (AUCs) are near 0.56, which means that our curves are near to the line of no-discrimination, the worst possible classification (a random guess). On the other hand the obtained AUCs for

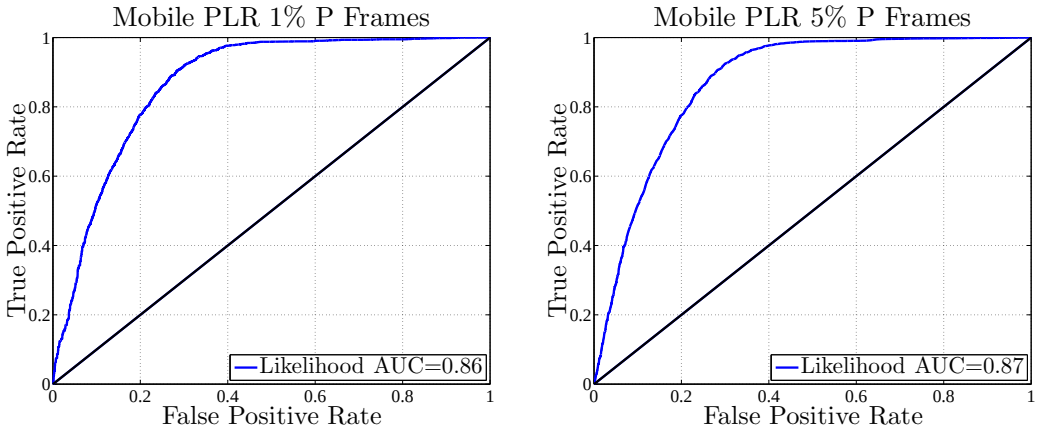

Figure 4.16: Temporal likelihood ROC curves for *Mobile* sequence at two different PLRs, 1% on the left 5%, on the right, over 15 realizations

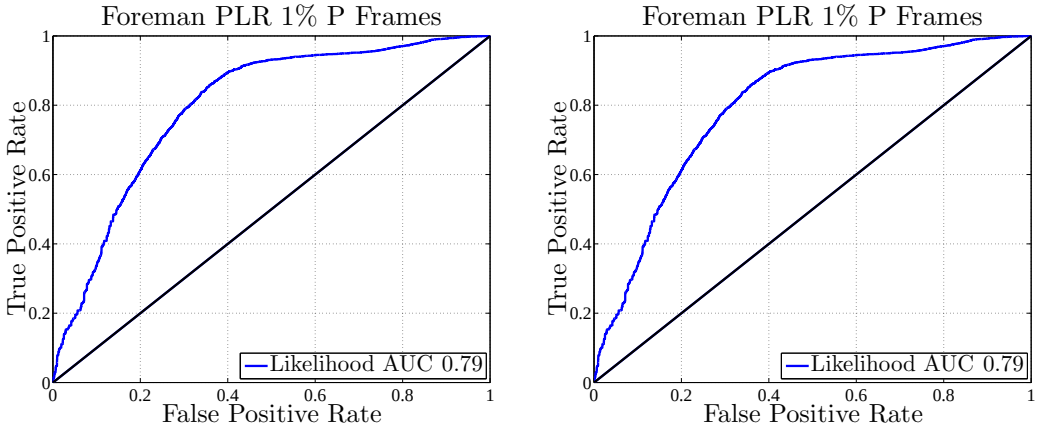

Figure 4.17: Temporal likelihood ROC curves for *Foreman* sequence at two different PLRs, 1% on the left 5%, on the right, over 15 realizations

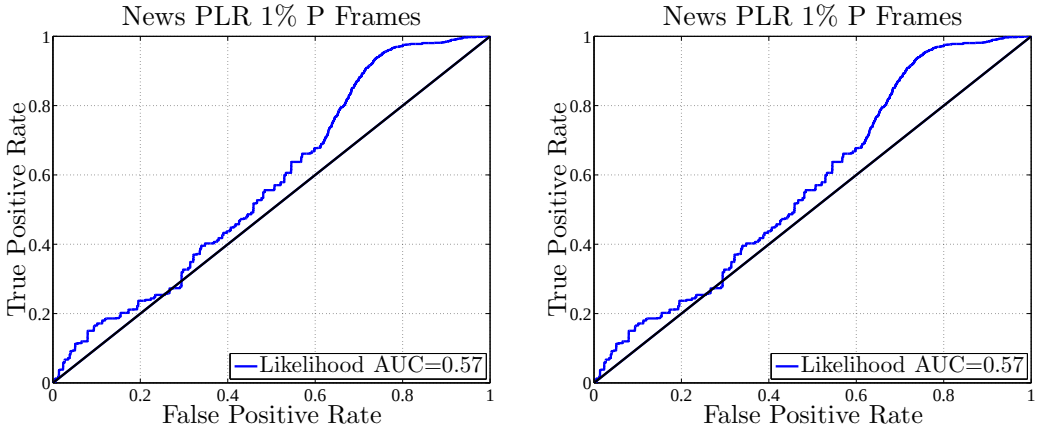

Figure 4.18: Temporal likelihood ROC curves for *News* sequence at two different PLRs, 1% on the left 5%, on the right, over 15 realizations

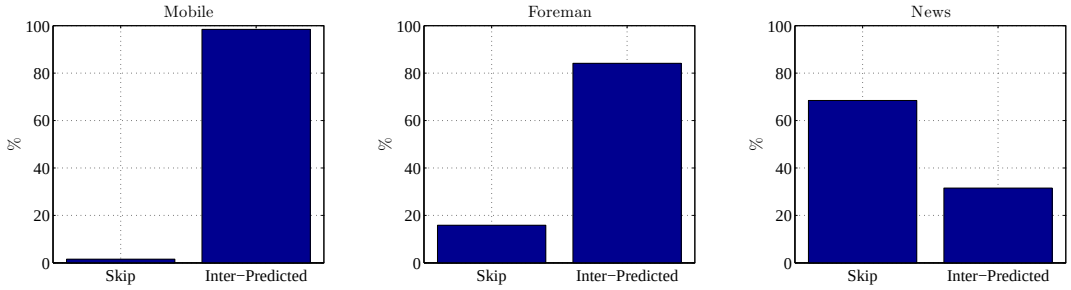

Figure 4.19: Macroblocks classified wrt their coding modes for three sequences.

Foreman and *Mobile* sequences have higher values above 0.80, it must be noticed that the perfect classifier has an AUC equal to one.

For *News* sequence in particular, it is trivial to see that MSE_T^i is not sufficient to discriminate between badly concealed macroblocks and not lost or correctly concealed ones, since their distributions are not sufficiently separable. This behavior can be explained looking to the histogram in Figure 4.19 where macroblocks are classified wrt their coding modes. It is possible to appreciate that for *News* sequence the percentage of skipped macroblocks ($MSE_T^i = 0$) is substantially higher wrt *Mobile* and *Foreman*. This means that all these macroblocks are tagged as positives even if they are negatives (False Positives), lowering the ROCs *News* curves wrt *Mobile* and *Foreman*.

We can so conclude that particulary for static sequences like *News* the estimated likelihood is not sufficient to achieve a class separation. The problem must be so regularized introducing a prior able to lowering the weight of false positives without changing the true ones.

4.4.2 Spatial Likelihood Estimation

As for temporal concealment we search for a feature $\mathbf{f}_j$ to be useful in spatial likelihood $\hat{P}^i(\mathbf{f}_S|BC_S)$ estimation starting from the mild assumption made upon spatial concealment:

- Spatially concealed macroblocks are a combination of neighborhood with prediction residuals' energy near to zero.

We now define the following quantities:

- $\tilde{M}^i_S(x,y,t)$ is the spatial concealed BxB i^{th} macroblock belonging to the frame at time t.
- $M^i_{REC}(x,y,t)$ is the i^{th} macroblock reconstructed by neighbor macroblocks boundaries

Our mild assumption over spatial concealment can be so written as:

$$\tilde{M}^i_S(x,y,t) = M^i_{REC}(x,y,t) \tag{4.20}$$

which means that the residuals' energy MSE^i_S computed between the spatially concealed i^{th} macroblock and $M^i_{REC}(x,y,t)$ will be equal to zero:

$$MSE^i_S(t) = \frac{1}{B^2}\sum_{x=1}^{B}\sum_{y=1}^{B}(\tilde{M}^i_S(x,y,t) - M^i_{REC}(x,y,t))^2 = 0 \tag{4.21}$$

$M^i_{REC}(x,y,t)$ depends from the adopted spatial concealment algorithm, but in our scenario we do not know the concealment algorithm process. We so decide to model $M^i_{REC}(x,y,t)$ as a bilinear function described as follows:

$$\hat{M}^i_{REC}(x,y,t) = \alpha^i W(Y+y)a^i(x) + \beta^i W(17-X-x)b^i(y) + \gamma^i W(17-Y-y)c^i(x) + \delta^i W(X+x)d^i(y) \tag{4.22}$$

where a^i, b^i, c^i, d^i, are the boundaries of the neighbors of the i^{th} macroblock used to spatially reconstruct the macroblock (Figure 4.20), $W(dist)$ is a function that weights the pixel values wrt its distance from the pixel under reconstruction, X and Y are the coordinates of the upper left pixel of the i^{th} macrolbock and $\alpha^i, \beta^i, \gamma^i, \delta^i$ are the unknown coefficients of the i^{th} macroblock. We moreover suppose to know that $W(dist)$ is expressed as follows:

$$W(x) = \frac{x}{x+y+(17-x)+(17-y)} \tag{4.23}$$

obviously it is always possible to redefine this particular functional form to fits other assumptions.

As already said $\alpha^i, \beta^i, \gamma^i, \delta^i$ are unknown and we cannot directly compute for a given macroblock its estimated spatial reconstruction $\hat{M}^i_{REC}(x,y,t)$. We so decide to search for $\alpha^i, \beta^i, \gamma^i, \delta^i$ that minimize $MSE^i_S(t)$ in eq. 4.21, this problem can be solved by least squares method.

Our ovedetermined system can be defined for the i^{th} macroblock belonging to a frame at time t as:

$$\tilde{M}^i_S(x,y,t) = \alpha^i W(Y+y)a^i(x) + \beta^i W(17-X-x)b^i(y) + \gamma^i W(17-Y-y)c^i(x) + \delta^i W(X+x)d^i(y) \tag{4.24}$$

with $1 \leqslant x \leqslant 16$ and $1 \leqslant y \leqslant 16$. Rewriting the system in matrix form we obtain:

$$\mathbf{mb} = \mathbf{A} \cdot \phi \tag{4.25}$$

where **mb**, **A**, ϕ are matrices defined as:

$$\tilde{M}^i_S(x,y,t) = \begin{pmatrix} \tilde{M}^i_S(1,1,t) \\ \tilde{M}^i_S(1,2,t) \\ . \\ . \\ . \\ \tilde{M}^i_S(16,16,t) \end{pmatrix} [256x1] \tag{4.26}$$

$$\mathbf{A} = \begin{pmatrix} W(1+Y)a^i(1) & W(16-X-1)b^i(1) & W(16-Y-1)c^i(1) & W(x)d^i(1+X) \\ W(2+Y)a^i(1) & W(16-X-1)b^i(2) & W(16-Y-2)c^i(1) & W(1)d^i(2+X) \\ . & . & . & . \\ . & . & . & . \\ . & . & . & . \\ W(16+Y)a^i(16) & W(17-X-16)b^i(y) & W(17-Y-16)c^i(16) & W(16)d^i(16+X) \end{pmatrix} [256x4] \tag{4.27}$$

$$\phi = \begin{pmatrix} \alpha^i \\ \beta^i \\ \gamma^i \\ \delta^i \end{pmatrix} [4x1] \tag{4.28}$$

thanks to the pseudoinverse we can so finally calculate ϕ that minimizes $MSE_S^i(t)$:

$$\phi = (\mathbf{A}^T \cdot \mathbf{A})^{-1} \cdot \mathbf{A} \cdot \mathbf{mb} \tag{4.29}$$

With the obtained ϕ we can so finally calculate $\hat{M}_{REC}^i(x,y,t)$.

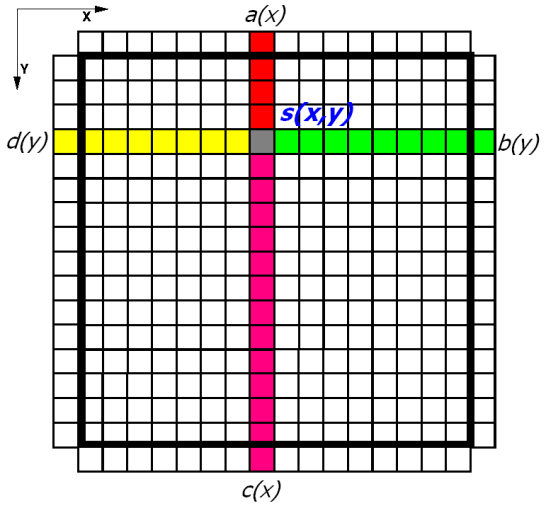

Figure 4.20: Spatial reconstruction $\hat{M}_{REC}^i(x,y,t)$ of the ith macroblock.

Since spatial prediction residuals' energy $MSE_S^i(t)$ is a discriminant feature wrt spatial badly concealed macroblocks, it can be so used in our spatial likelihood estimation, setting the vector of features $\mathbf{f}_S = MSE_S^i(t)$ with size 1x1. We can so write spatial likelihood $P^i(\mathbf{f}_S|BC_S)$ as:

$$P^i(MSE_S^i|BC_S) \tag{4.30}$$

that can be so modeled as an impulse centered in MSE_S^i equal to zero.

However since we do not know the real $M_{REC}^i(x,y,t)$, our estimation $\hat{M}_{REC}^i(x,y,t)$ may lead to a $MSE_S^i(t)$ which is different from zero so also in this case we decide to model $P^i(MSE_S^i|BC_S)$ distribution with an exponential function; high probabilities

value are so linked to low MSE_S^i values.

We so map the chosen feature MSE_S^i to obtain an estimate of the spatial likelihood $P^i(MSE_S^i|BC_S)$:

$$\hat{P}^i(MSE_S^i|BC_S) \propto e^{\frac{MSE_S^i}{\alpha_{LS}}} \tag{4.31}$$

In Figure 4.21 is represented the estimated likelihood for a chosen corrupted I frame.

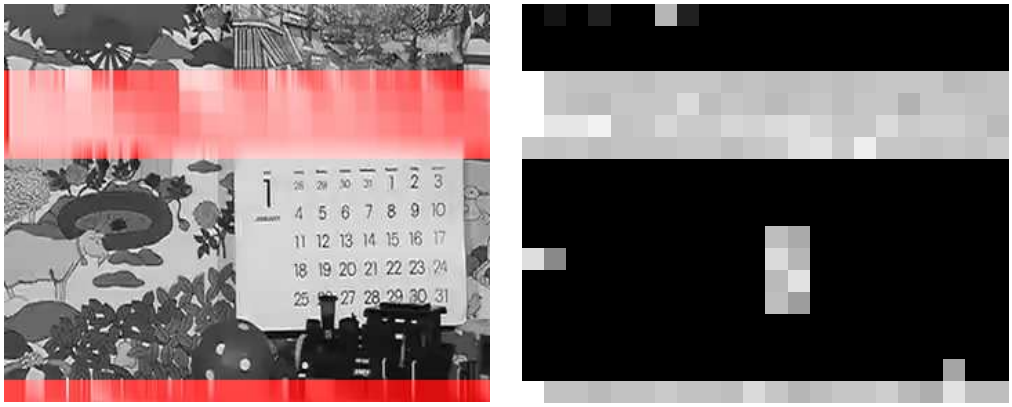

Figure 4.21: On the left the corrupted video, with lost slices in red, on the right the estimated likelihood map

In Figures 4.22, 4.23, 4.24 are presented the ROCs of the estimated spatial likelihood $\hat{P}^i(MSE_S^i|BC_S)$ wrt our defined spatial ground truth for *Mobile Foreman* and *News* sequences at different PLRS (1,5) over 15 realizations. In general the obtained AUCs are above 0.80, and in particular AUCs related to *Mobile* achieve a nearly perfect classification, since its content has a texture that is not spatial predictable and it is more difficult that false positives arise. On the other hand for *News* sequence there is an high number of false positives. This is possible since some areas of the sequence are flat and so spatially predictable, giving birth to false positives.

Also in this case the introduction of a prior may help in reaching higher AUCs, working as confidence wrt our estimate, lowering false positives weight.

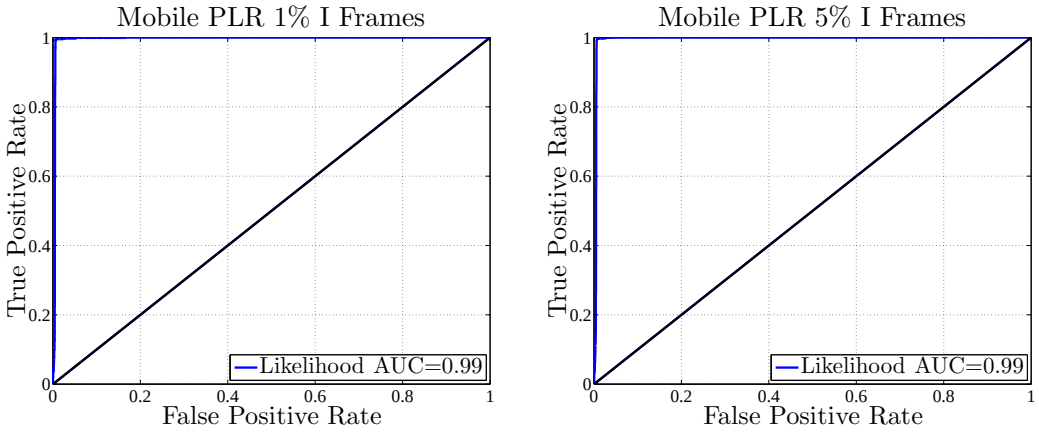

Figure 4.22: Spatial likelihood ROC curves for *Mobile* sequence at two different PLRs, 1% on the left 5%, on the right, over 15 realizations

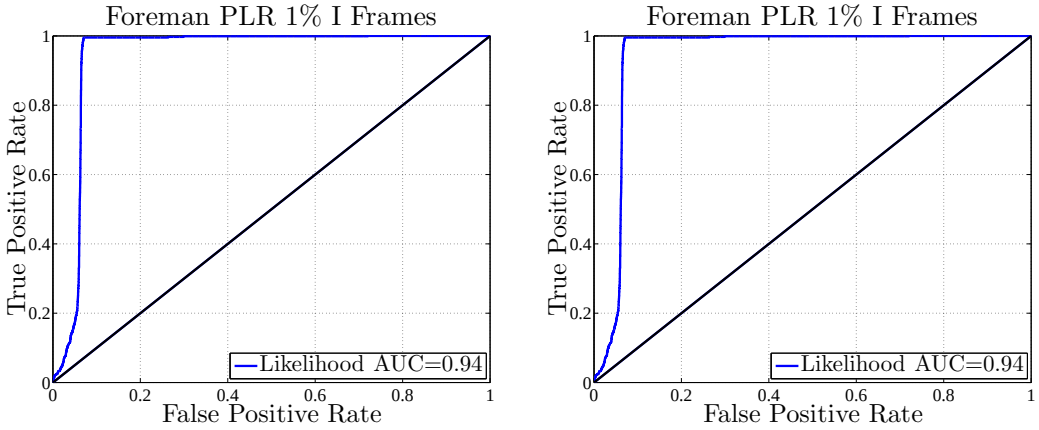

Figure 4.23: Spatial likelihood ROC curves for *Foreman* sequence at two different PLRs, 1% on the left 5%, on the right, over 15 realizations

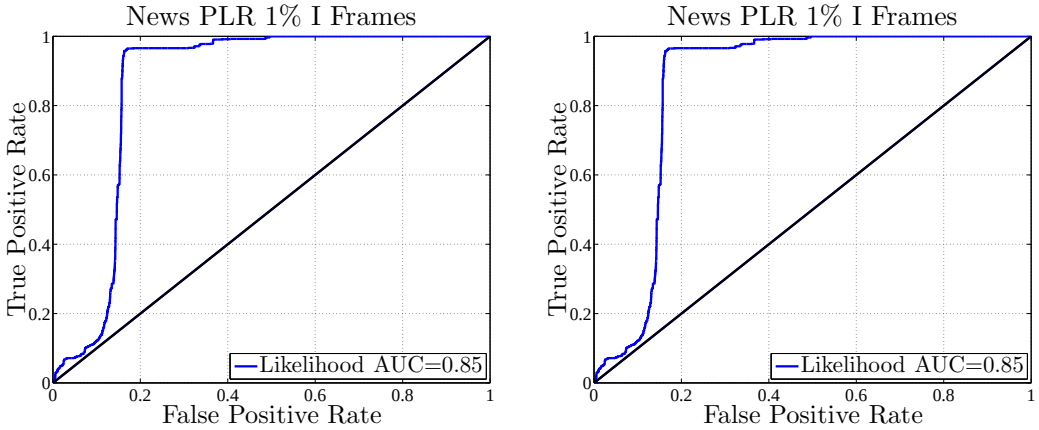

Figure 4.24: Spatial likelihood ROC curves for *News* sequence at two different PLRs, 1% on the left 5%, on the right, over 15 realizations

4.5 Prior Estimation

As seen till now classifications achieved with likelihoods are not always sufficient. In fact we have not an a priori knowledge that features MSE_S^i and MSE_T^i used to classify between correctly and badly concealed blocks are not also linked to some intrinsic characteristics of the video. For example MSE_S^i false positives are not lost macroblocks that are easily spatially predictable due to their own content. Also for MSE_T^i the same consideration holds. In this case false positives are not lost macroblocks that are temporally predicated with a low residuals' energy (e.g. skip coded macroblocks). We want so to create a map which is able to filter out estimated likelihood $\hat{P}^i(MSE^i_{T\text{or}S}|BC)$ false positives due to intrinsic characteristics of the sequence under consideration.

Prior probability $P^i(BC)$, calculated over the noiseless frame, gives the probability for the i^{th} macroblock to be badly concealed. $P^i(BC)$ satisfies the previous constrain, in general in fact estimated likelihoods $\hat{P}^i(MSE^i_{T|S}|BC)$ false positives have a low $P^i(BC)$ values since their content is intrinsically easily predictable and so easily concealable. On the contrary estimated likelihoods $\hat{P}^i(MSE^i_{T|S}|BC)$ true positives will have high values of priors $P^i(BC)$ since their content is not easily predictable and so not easily concealable. Roughly speaking confidence weights how much the likelihoods features MSE_S^i, MSE_T^i values are linked to a badly concealment or to a content property.

Obviously in our scenario the noiseless frame is unavailable. We so decide to compute the likelihood estimate $\hat{P}^i(BC)$ using the previous motion compensated P frame. We will see that this approximation is not always correct, due to the differences between current and motion compensated previous frame. We however chose to use it since it is the one that fits better our mild assumptions.

4.5.1 Temporal Prior Estimation

$\hat{P}^i(BC_T)$ models the probability that the temporal concealment is not able to perform a restoration without visual impairments. From our mild assumptions we know that temporal concealment simply copies a macroblock from a previous

reference frame trying to preserve the motion field around it.

We choose to model the temporal prior $\hat{P}^i(BC_T)$ using the $MVPD^i(t-1)$ of the previous motion compensated P frame, which is the difference between the motion vector predictor and the motion vector under exam, as defined in equation 4.11. This feature is able to discriminate between zones that are restorable with no visual impairments from zones where the concealment is not able to achieve these performances. The MVPD is in fact a measure of the predictability of the motion vector under exam wrt its neighbor motion vectors, typically a temporal concealment works better in these zones since is able to restore the macroblocks looking at the neighbors.

We can so define the MVPD for the i^{th} macroblock calculated over the previous motion compensated P frame at time $t-1$:

$$MVPD^i(t-1) = |MVP(x-v_x, y-v_y, t-1) - MV(x-v_x, y-v_y, t-1)| \tag{4.32}$$

where v_x and v_y are the motion vectors related to the i^{th} macroblock belonging to the frame at time t, and x and y are the coordinates of the upper left pixel of the same block.

It must be noticed that it may happen that the predictor described by v_x and v_y is a composition of pixels belonging to more than one macroblock (max 4) of the frame at time $t-1$. This means that our predictor can have more than one MVP and MV. Among all the possible MPVs and MVs we decided to chose the ones which belong to the macroblock at time $t-1$ that shares the biggest number of pixels with our predictor.

We can so finally define the estimated temporal concealment prior as follows:

$$\hat{P}^i(BC_T) \propto 1 - e^{\frac{MVPD^i(t-1)}{\alpha_{PT}}} \tag{4.33}$$

as for likelihood we map the MVPD calculated over the motion compensated frame at time $t-1$ between 0 and 1 to obtain a probability map. We can so finally

state that $\hat{P}^i(BC_T|MSE_T^i)$ can be calculated as:

$$\hat{P}^i(BC_T|MSE_T^i) \propto (1 - e^{\frac{MVPD^i(t-1)}{\alpha_{PT}}}) \cdot e^{\frac{MSE_T^i}{\alpha_{LT}}}; \tag{4.34}$$

In Figure 4.25 are represented the obtained posterior probability map for a given corrupted frame.

Figure 4.25: Starting form the left the corrupted video the likelihood map and the posterior map

Differently from likelihood ROC shape and AUC depends from the chosen α_{PT}, α_{LT}, the two decay factors. In fact different mapping leads to a different relationship between the assigned probabilities of each macroblock. In Figure 4.26 are reported the surfaces of AUCs wrt the two decay factors for two sequences (*Foreman, News*). It is possible to notice that the AUC strictly depends from α_{LT} and α_{PT}. We search for the α_{PT}, α_{LT} which maximize the related ROC's AUC. However we noticed that the obtained decay values are quite sequence indepen-

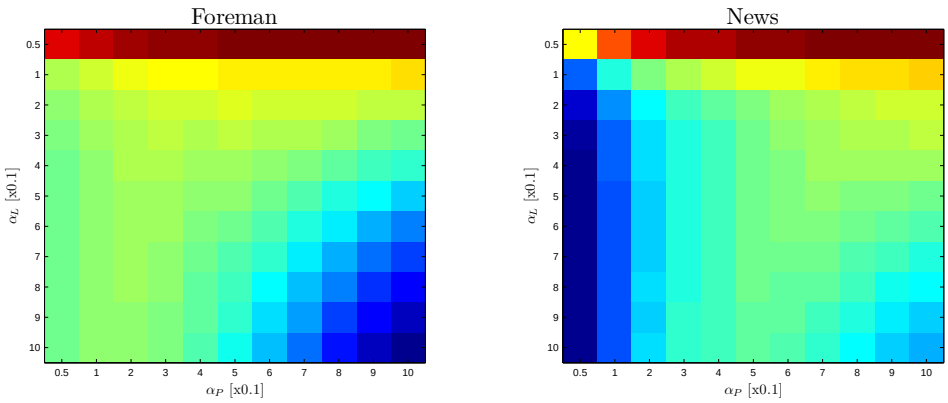

Figure 4.26: AUC surfaces for *Foreman, News* sequences at PLR 5%

dent, we so decide to set up $\alpha_{PT} = 0.5$, $\alpha_{LT} = 0.05$ for all the three videos.

In Figures 4.27, 4.28, 4.29 is possible to appreciate the ROC curves obtained using $\hat{P}^i(BC_T|MSE_T^i)$ as classification feature with the same dataset used in Section 4.4. For *News* sequence the new feature obtains better results wrt the likelihood used in Section 4.4. On the contrary for *Mobile* and *Foreman* the AUCs obtained with the likelihood $\hat{P}^i(MSE_T^i|BC_T)$ are greater.

Since the prior $P^i(BC_T)$ is estimated over the motion compensated previous P frame motion vectors, it is trivial to understand that for high motion sequences, like *Foreman* and *Mobile*, this estimate is not sufficiently precise. In fact the motion vectors change fast from frame to frame, and previous motion compensated motion vectors $MV(x-v_x, y-v_y, t-1)$ can substantially differ from the ones of the noiseless frame at time t.

The adopted solution is to neglect the prior for sequence whose general motion differs too much from frame to frame. The total motion difference (TMD) is calculated for each frame as follows:

$$TMD(t) = \sum_{x=1}^{N} \sum_{y=1}^{M} |MV(x,y,t) - MV(x,y,t-1)| \tag{4.35}$$

where $MV(x,y,t)$ is the motion vector of the macroblock at the x,y position belonging to the frame at time t, while $MV(x,y,t-1)$ is the motion vectors of the co-located macroblock belonging to the frame at time $t-1$. In Figure 4.30 is depicted TMD function wrt time for two different sequences. It is possible to notice that *Foreman* TMD values are in general greater wrt *News* ones. This consideration suggests that a thresholded TMD value may be used as a switch for prior usage. In particular choosing a certain threshold th_{TMD}, if $TMD \leqslant th_{TMD}$, the prior will be used. Otherwise if $TMD > th_{TMD}$, the prior map will be neglected.

So the algorithm for $P^i(BC_T|MSE_T^i)$ estimation can be summarized as follows:

1: **for** each macroblock *ith* **do**
2: **if** $TMD^i < th_{TMD}$ **then**
3: $\hat{P}^i(BC_T|MSE_T^i) \propto \hat{P}^i(MSE_T^i|BC_T) \cdot \hat{P}^i(BC_T)$

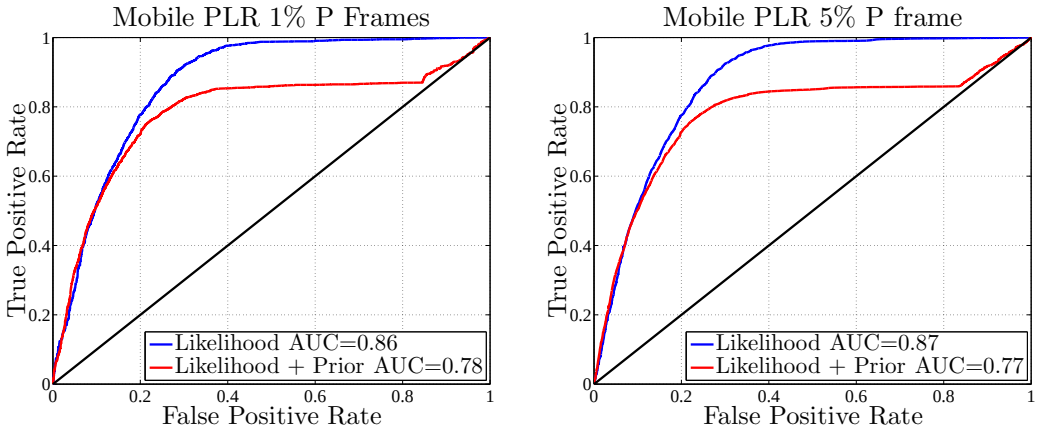

Figure 4.27: Temporal likelihood and prior ROC curves for *Mobile* sequence at two different PLRs, 1% on the left 5%, on the right, over 15 realizations

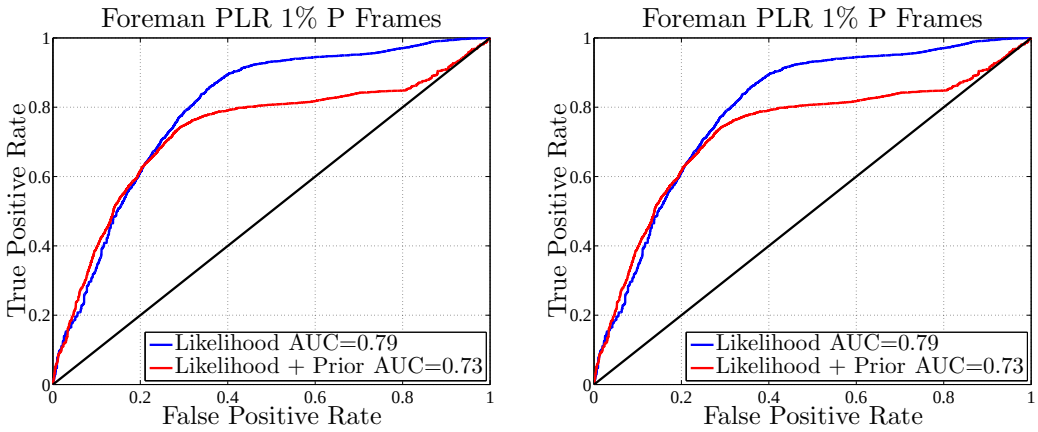

Figure 4.28: Temporal likelihood and prior ROC curves for *Foreman* sequence at two different PLRs, 1% on the left 5%, on the right, over 15 realizations

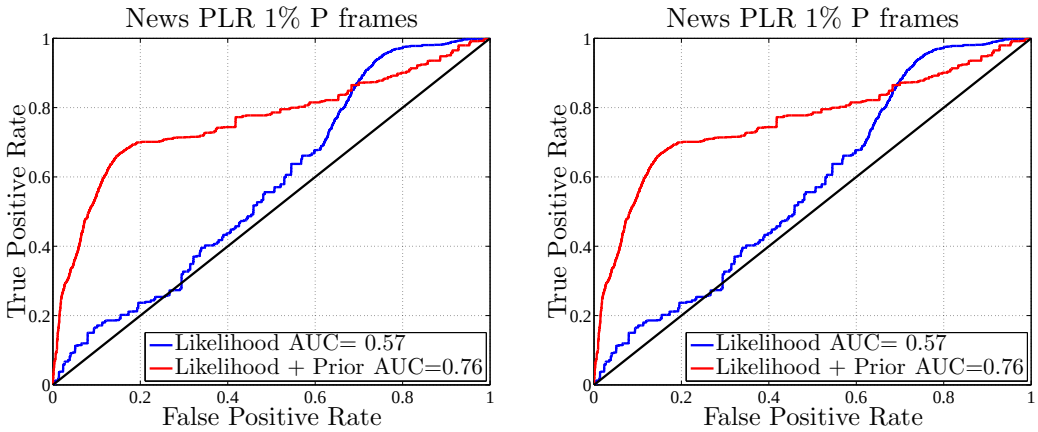

Figure 4.29: Temporal likelihood and prior ROC curves for *News* sequence at two different PLRs, 1% on the left 5%, on the right, over 15 realizations

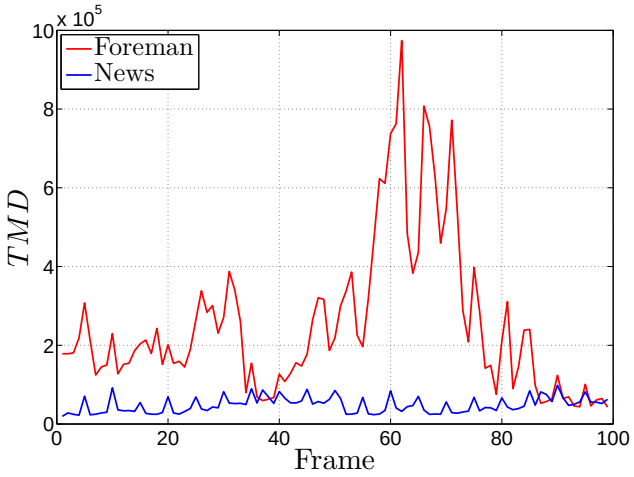

Figure 4.30: TMD for each frame of *Foreman* and *News* sequences.

4: **else**

5: $\hat{P}^i(BC_T|MSE^i_T) \propto \hat{P}^i(MSE^i_T|BC_T)$

6: **end if**

7: **end for**

4.5.2 Spatial Prior Estimation

As for temporal prior, $\hat{P}^i(BC_S)$ models the probability that the spatial concealment is not able to perform a restoration without visual impairments, an index of spatial concealment effectiveness.

We choose to model the spatial prior $\hat{P}^i(BC_S)$ using the $MSE^i_S(t-1)$ which is the MSE^i_S computed over the previous motion compensated P frame. This feature is able to discriminate between zones that are spatially restorable with no visual impairments from zones where the spatial concealment is not able to achieve these performances. In fact false positives macroblock, wrt the estimated likelihood $\hat{P}^i(MSE^i_S|BC_S)$, will have low values of $MSE^i_S(t-1)$ since their spatial predictability depends from a content property.

We can so finally define the estimated spatial concealment prior as:

$$\hat{P}^i(BC_S) \propto 1 - e^{\frac{MSE^i_S(t-1)}{\alpha PS}} \tag{4.36}$$

and the posterior $\hat{P}^i(BC_S|MSE_S^i)$ can be computed as:

$$\hat{P}^i(BC_S|MSE_S^i) \propto (1 - e^{\frac{MSE_S^i(t-1)}{\alpha_{PS}}}) \cdot e^{\frac{MSE_T^i}{\alpha_{LS}}} \tag{4.37}$$

In Figure 4.31 is represented the obtained posterior probability map for a given corrupted frame.

As done for temporal prior we select α_{PS}, α_{LS} which maximize the related ROC's AUC. In Figure 4.32 are depicted the corresponding surfaces. In this case we set $\alpha_{PS} = 0.9$, $\alpha_{LS} = 0.1$ for all the three sequences.

ROC curves obtained using prior $\hat{P}^i(BC_S|MSE_S^i)$ are reported in Figures 4.34, 4.35, 4.36. The obtained AUCs are always bigger than the ones obtained using only the likelihood, achieving our objective. However it is possible to notice that the ROC curves obtained with posterior $\hat{P}^i(BC_S|MSE_S^i)$ are not always above the curves obtained by likelihood $\hat{P}^i(MSE_S^i|BC_S)$.

This behavior is caused by the particular concealment algorithm. The confidence is in fact computed over a dataset that is an estimate of the noiseless frame. In particular we suppose to compute the spatial concealment effectiveness over a particular macroblock when all its neighbors are correctly received. This consideration does not hold in our case, since the concealment also uses already restored blocks during the recovering process. In particular it may happen that a macroblock with low $\hat{P}^i(BC_S)$, will be reconstructed with visual impairments since the used boundaries belong to already badly concealed macroblocks. So false negatives number grows up lowering true positives rate for curves obtained

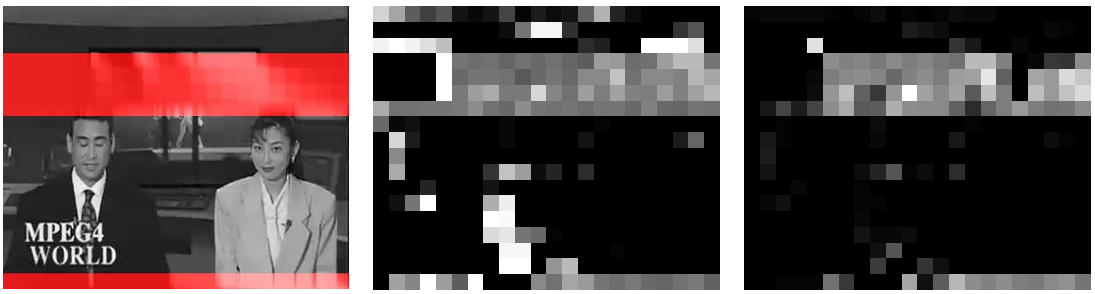

Figure 4.31: Starting form the left the corrupted video the likelihood map and the posterior map

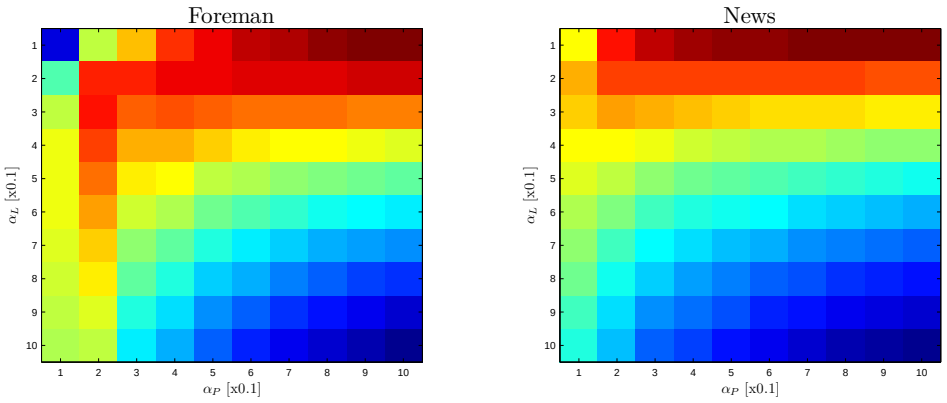

Figure 4.32: AUC surfaces for *Foreman*, *News* sequences at PLR 5%

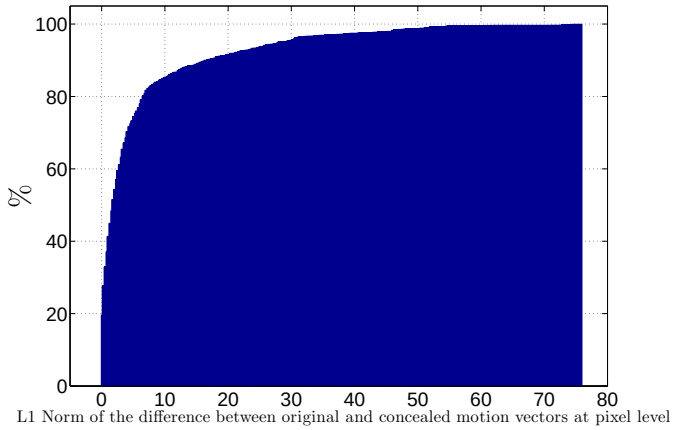

Figure 4.33: Cumulative distribution for the L1 norm of the difference between motion vectors belonging to the noiseless and corrupted frame.

with $\hat{P}^i(BC_S|MSE^i_S)$.

This problem can obviously only be solved knowing the slice structure and the concealment process, but in our scenario these assumptions can not be made.

It must be noticed that the same considerations can also be made upon temporal prior estimation. In particular for temporal concealment it may happen that a lost macroblock may be restored using motion vectors belonging to badly concealed blocks. We can so easily argue that wrong temporal confidence estimations are due to the difference of motion vectors used by concealment process and the one of the prediction of the noiseless frame used to compute the confidence $P^i(BC_T)$. In fact from the cumulative distribution function in Figure 4.33, which is computed for the L1 norm of the difference between motion vectors belonging

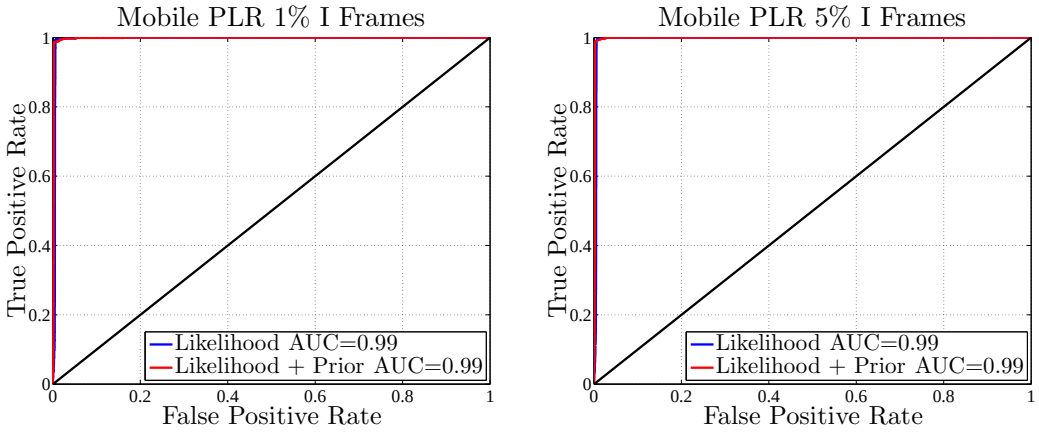

Figure 4.34: Spatial likelihood and prior ROC curves for *Mobile* sequence at two different PLRs, 1% on the left 5%, on the right, over 15 realizations

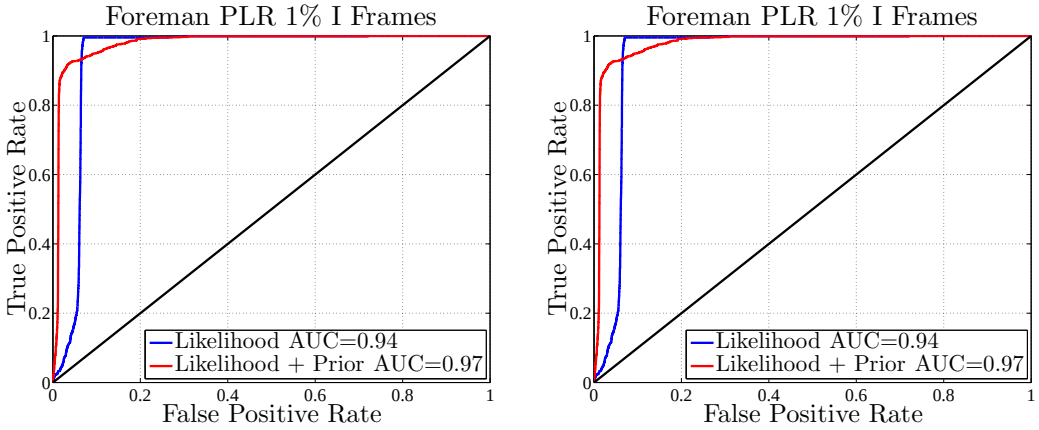

Figure 4.35: Spatial likelihood and prior ROC curves for *Foreman* sequence at two different PLRs, 1% on the left 5%, on the right, over 15 realizations

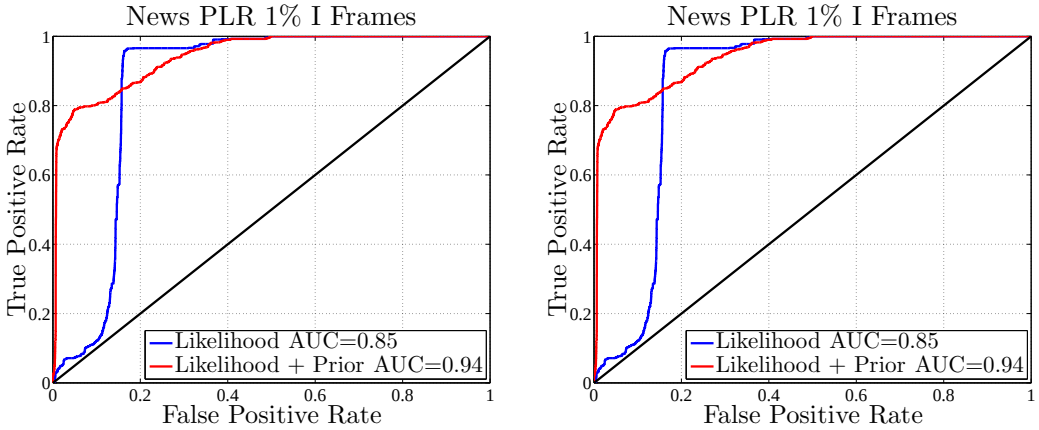

Figure 4.36: Spatial likelihood and prior ROC curves for *News* sequence at two different PLRs, 1% on the left 5%, on the right, over 15 realizations

to the noiseless and corrupted frame over lost slices, it is possible to notice that an high percentage of vectors have a difference higher than 4 pixels.

4.6 Markov Fields

Till now we did not take into account spatial relationship between lost macroblocks. In fact knowing the particular slice structure it is possible to deduce also the particular topology of map of lost macroblocks, that may help during its estimation.

Unfortunately we do not have this a priori knowledge in our scenario, but we can however argue that there is a spatial relationship between lost macroblocks, since we know that they are packetized into slices. This spatial relationship can be modeled using a Markov Random Field.

Markov Random Fields (MRF) are probabilistic undirected graphical models that help in the analysis of probability distributions. In particular they provide a simple way to visualize the structure of a probabilistic model including the conditional dependence between variables. A graph comprises *nodes* and *links*. Each node represents a random variable (or a group of random variables), while the link express the probabilistic relationship between them. Roughly speaking this models are able to capture the *causal* process by which the observed data are generated.

In Figure 4.37 is depicted the adopted undirect graphical model (MRF) representing a corrupted frame, in which each node x_i is a boolean variable denoting the state of the i^{th} macroblock to be restored with or without visual impairments, and y_i denotes the corresponding estimated $\hat{P}^i(BC|MSE^i_{T|S})$. Note that to distinguish between *observed* and *hidden* variables the nodes are shaded or empty.

It is trivial to understand that in this model we suppose that the state of a macroblock is linked to the ones of its four neighbors. But this assumption can be changed creating the lattice that performs better wrt unknown slicing structure. We want so to solve the defined MRF searching the maximum a posteriori for $\hat{P}^i(BC, \text{Neighborhood}|MSE^i_{T|S})$ where a new spatial prior was inserted thanks

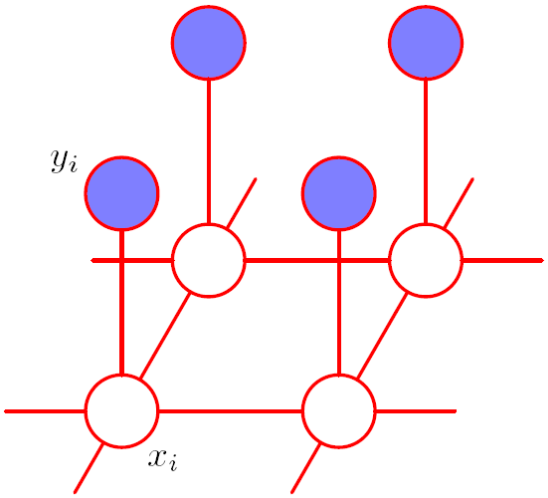

Figure 4.37: The undirect graph used to model the estimated map of badly concealed macroblock.

to Markov field. It was demonstrated by Boykov et al. (1998) and Greig et al. (1989) that this maximization problem can also be solved by Min-Cut/Max-Flow graph cut technique for a particular class of functions. We so redefine our frame model as the two terminal graph addressed by Boykov et al. (1998) and Greig et al. (1989), in particular we decided to model our problem as already done for image segmentation in Boykov and Funka-Lea (2006), since it can be similarly described.

In general a graph $G = \langle V, E \rangle$ is composed by a set of nodes V and a set of links or edges E. In our specific case the nodes are the macroblocks. The graph contains also two additional special nodes called terminals which correspond to the two labels that can be assigned to our macroblocks (Badly Concealed and Correctly received or concealed). These two terminals are usually called *source*, s and *sink*, t.

We are so interested in minimize the energy function associated to the new defined graph depicted in Figure 4.38:

$$E(L) = \sum_{p \in P} D_p(L_p) + \sum_{(p,q) \in N} V_{p,q}(L_p, L_q) \tag{4.38}$$

where $L = L_p | p \in P$ is the labeling for each macroblock of our frame P, $D_p(.)$ is a

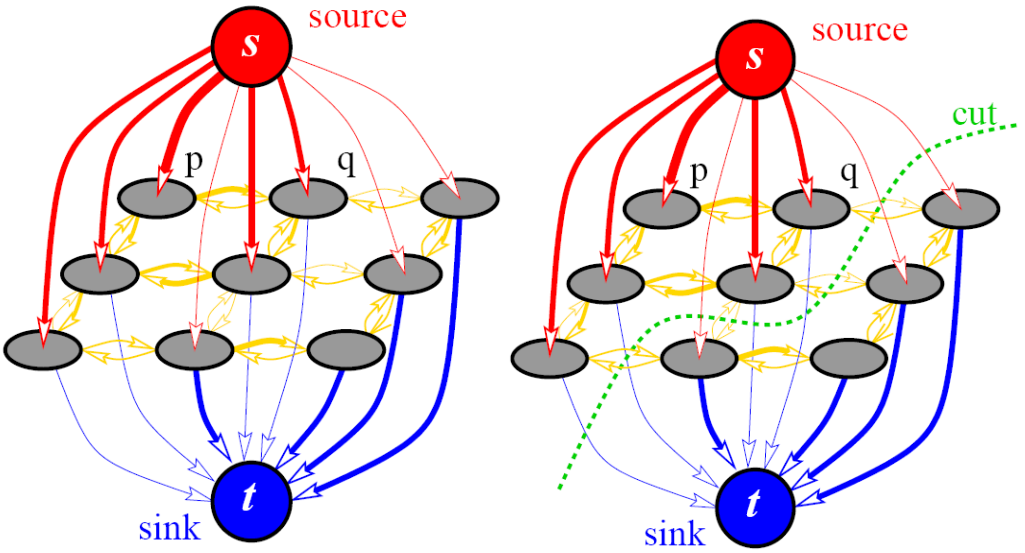

Figure 4.38: The adopted directed capacitated graph. Edge costs are reflected by their thickness.

data penalty function, $V_{p,q}$ is an interaction potential and N is a set of all pairs of neighboring macroblocks.

There are so two types of edges in a graph: n-links and t-links. N-links connect a pair of neighbor macroblocks and their costs are derived from the macroblocks interaction term $V_{p,q}$ in 4.38, which represents a penalty for discontinuity between the blocks. On the other hand t-links connect macroblocks with terminals (labels). In this case their costs are associated to a penalty for assigning the macroblock to the corresponding label. This cost is derived from the D_p term in 4.38.

In particular in our scenario n-links weights are computed as follows:

$$N-links(p,q) = |\hat{P}^p(BC|MSE^i_{T|S}) - \hat{P}^q(BC|MSE^i_{T|S})| \tag{4.39}$$

where p and q are two adjacent macroblocks. On the other hand t-links weights are computed as:

$$T-links(p,L) = |L - \hat{P}^p(BC|MSE^p_{T|S})| \tag{4.40}$$

where the p^{th} macroblock is connected to the L^{th} label, in our case to the sink ($L=0$ correctly concealed or received macroblocks) or source ($L=1$ badly concealed

macroblocks)

To solve $E(L)$ minimization problem we use a Min-Cut/Max-Flow algorithm. In particular s/t cut C on a graph with two terminals is a partitioning of the nodes in the graph into two disjoint subsets S and T such that the source s is in S and the sink t is in T.

In combinatorial optimization the cost of a cut $C = (S,T)$ is defined as the sum of the costs of boundary edges (p,q) where $p \in S$ and $q \in T$. The minimum cut problem on a graph is to find a cut that has the minimum cost among all cuts. One of the fundamental results in combinatorial optimization is that the minimum s/t cut problem can be solved by finding the maximum flow from the source s to the sink t. Loosely speaking, maximum flow is the maximum amount of water that can be sent from the source to the sink by interpreting graph edges as directed pipes with capacities equal to edge weights. The theorem of Ford and Fulkerson states that a maximum flow from s to t saturates a set of edges in the graph dividing the nodes into two disjoint parts S,T corresponding to a minimum cut. To solve this minimization problem we use the algorithm presented by Boykov and Kolmogorov (2001).

Summarizing, for each frame the posterior probability is estimated $\hat{P}^p(BC|MSE^p_{T|S})$, and the n-link and t-link are computed as described in 4.39 and 4.40. Kolmogorov and Boykov's Min-Cut/Max-Flow algorithm is then ran over the constructed graph obtaining a classification for each macroblock.

Our *Source* models macroblock concealed with visual impairments, on the other hand *Sink* models received macroblocks and correctly concealed ones. We label the first class with one (Positive) and the second one with zero (Negative).

As seen till now our a posterior probability estimations $\hat{P}^i(BC|MSE^p_{T|S})$ are created using an exponential mapping function whose decay depends from the particular chosen α_P and α_L. It is clear that changing these parameters is possible to obtain a different estimation of the posterior probability, and specifically different weights for t-links and n-links.

In Figure 4.39 are reported the TPR and FPR for the map obtained using Min-Cut/Max-Flow algorithm wrt different likelihood's α_L and prior's α_P. From these

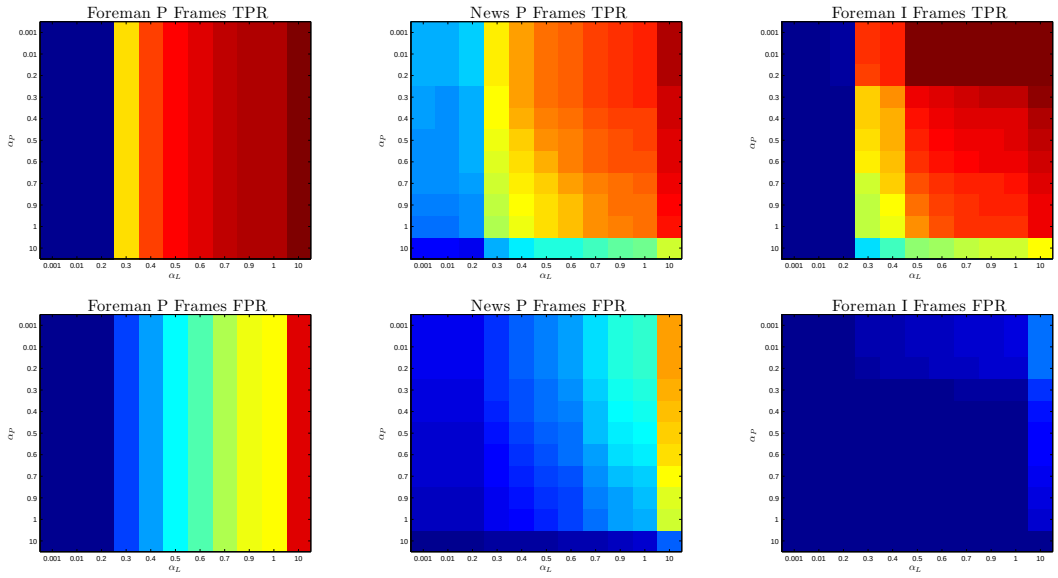

Figure 4.39: Starting form the left the corrupted video the likelihood map and the posterior map

surfaces we can so choose the α_P and α_L to obtain the desired ratio between TPR and FPR. In particular for spatial concealment we chose $\alpha_{LS} = 0.4$ and $\alpha_{PS} = 0.001$ for all the three sequences. On the other hand for temporal concealment we set $\alpha_{LT} = 0.3$ for *Foreman* and *Mobile* sequences, while for static sequence like *News* $\alpha_{LT} = 1$ and $\alpha_{PT} = 0.02$.

In Figure 4.46 it is possible to appreciate the results obtained by Min-Cut/Max-Flow algorithm in estimating the map of lost macroblocks . The Max-Flow/Min-Cut obtained TPRs and FPRs for the different sequences are shown in Figures 4.40,4.41,4.42,4.43,4.44 and 4.45 together with the ROCs previously obtained for I and P frames using likelihood and posterior. It is possible to notice that estimations obtained with Min-Cut/Max-Flow have always a TPR, FPR ratio higher wrt the one achieved by prior and posterior at the same FPR. However for temporal concealment in *News* sequence the obtained results are comparable to the ones obtained with posterior. However it must be noticed that the ROC curves does not take into account the specific weight of the recognized true positives in terms of channel induced distortion D^i. It may so happen that even if we have the same TPR, FPR ratio the true positives recognized by the Min-Cut/Max-Flow algorithm may be different from the ones found by likelihood or prior and in partic-

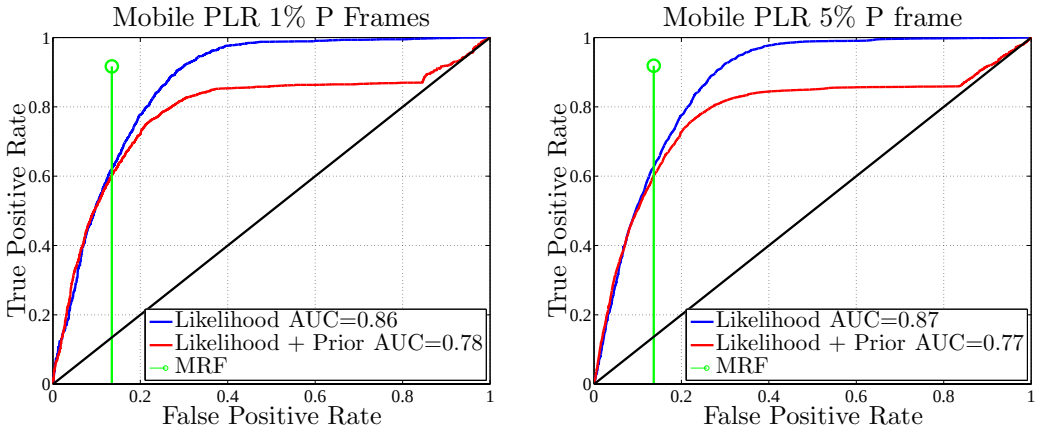

Figure 4.40: Temporal likelihood and prior ROC curves with TPR and FPR obtained by Min-Cut/Max-Flow algorithm for *Mobile* sequence

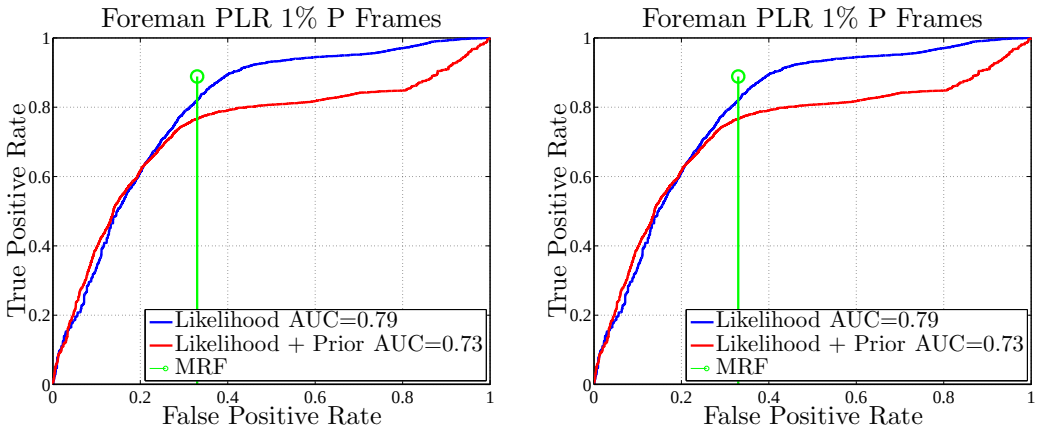

Figure 4.41: Temporal likelihood and prior ROC curves with TPR and FPR obtained by Min-Cut/Max-Flow algorithm for *Foreman* sequence

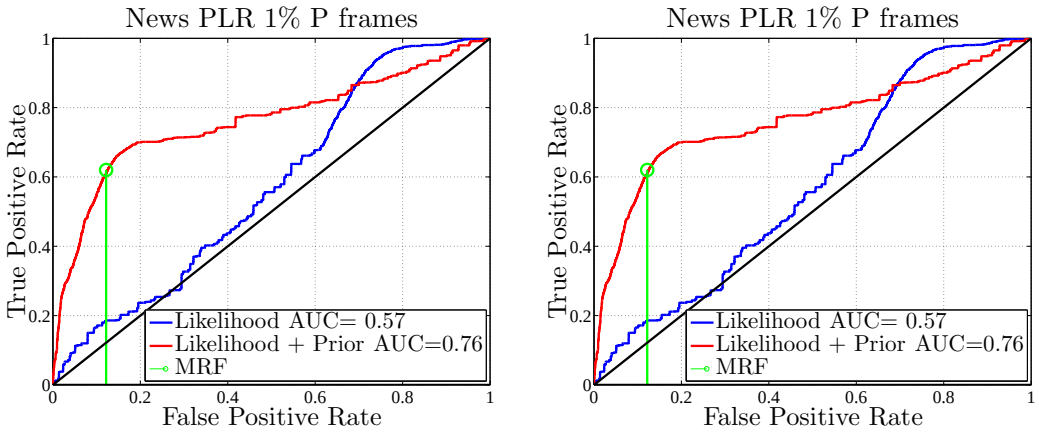

Figure 4.42: Temporal likelihood and prior ROC curves with TPR and FPR obtained by Min-Cut/Max-Flow algorithm for *News* sequence

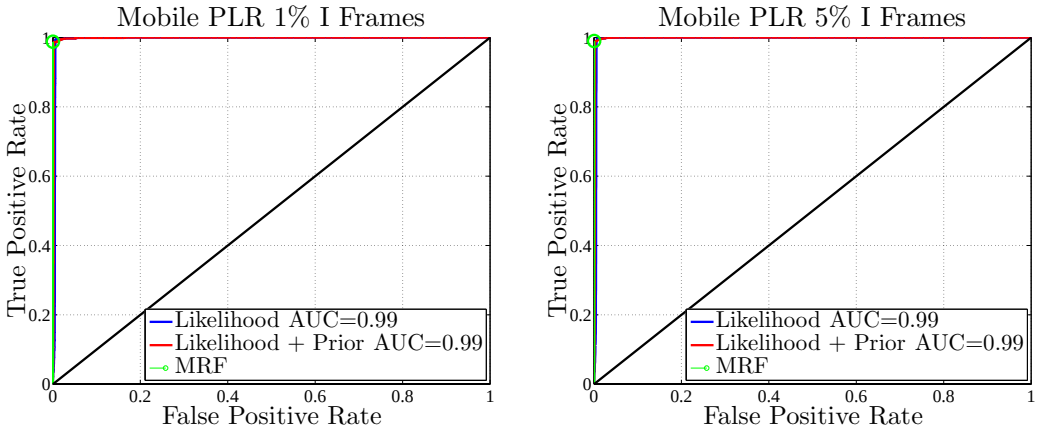

Figure 4.43: Spatial likelihood and prior ROC curves with TPR and FPR obtained by Min-Cut/Max-Flow algorithm for *Mobile* sequence

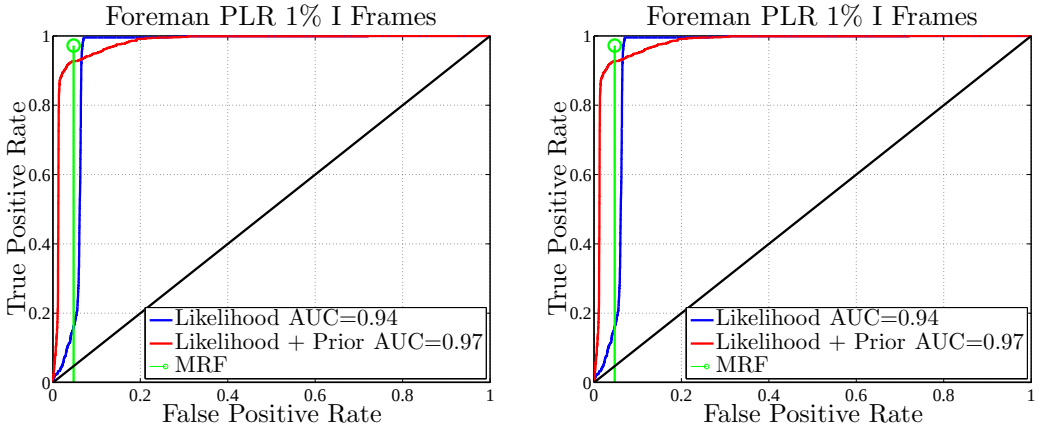

Figure 4.44: Spatial likelihood and prior ROC curves with TPR and FPR obtained by Min-Cut/Max-Flow algorithm for *Foreman* sequence

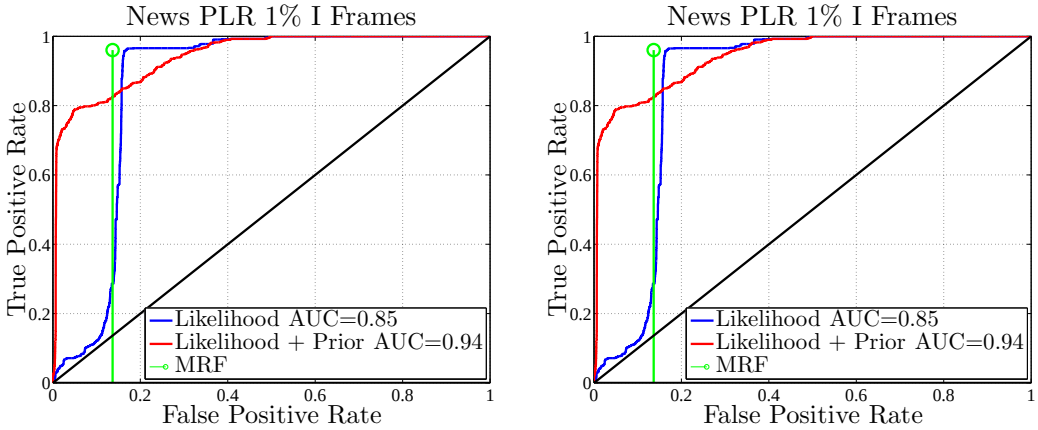

Figure 4.45: Spatial likelihood and prior ROC curves with TPR and FPR obtained by Min-Cut/Max-Flow algorithm for *News* sequence

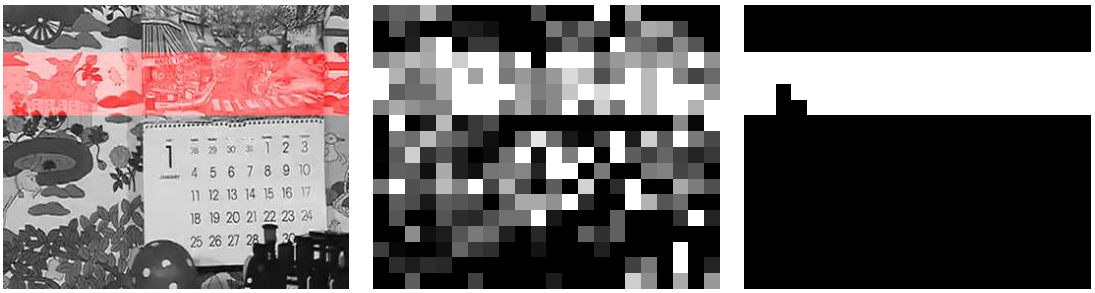

Figure 4.46: Starting form the left the corrupted video the likelihood map and the posterior map

ular the ones recognized by Min-Cut/Max-Flow algorithm may have higher D^i. This means that with Min-Cut/Max-Flow algorithm we are able to achieve better channel distortion estimation. These considerations are confirmed by results in Section 5.2.

4.7 System Overview

In this chapter we described a method to estimate the map of lost badly concealed macroblocks able to feed NORM. For each I and P frames the likelihoods $P^i(MSE_S^i)|BC_S)$ and $P^i(MSE_T^i|BC_T)$ are estimated at macroblock level as described in Section 4.4. Then, using the motion compensated previous P frame, the priors $P^i(BC_S)$ and $P^i(BC_T)$ are estimated as in Section 4.5. With the estimated priors and likelihoods we are able to compute the posterior probabilities $P^i(BC_S|MSE_S^i)$ and $P^i(BC_T|MSE_T^i)$, useful to define the undirected graph of Section 4.6, which takes into account the spatial relationship between lost macroblocks. Solving the defined graph with the Min-Cut/Max-Flow algorithm described in Boykov and Kolmogorov (2001) we finally obtain the estimated map of lost badly concealed macroblocks. An overview of the system is depicted in Figure 4.47.

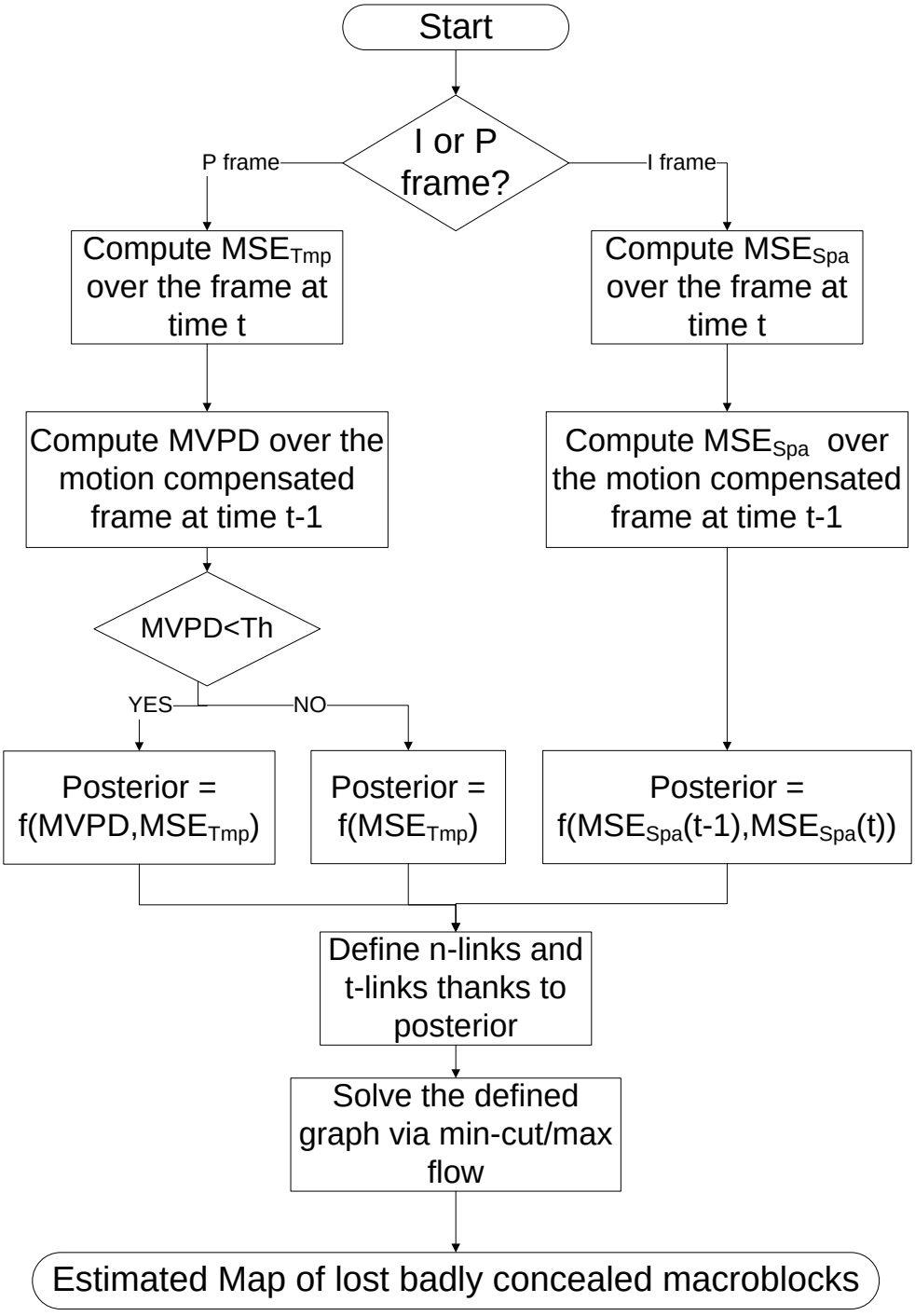

Figure 4.47: System overview for lost badly concealed macroblock

CHAPTER 5

EXPERIMENTAL RESULTS AND COMPARISON

This chapter is dedicated to the analysis of NR-P NORM performances. NR-P NORM algorithm is obtained feeding NORM with the estimated bitstream input described in Chapter 3, and the estimated map of lost macroblocks obtained in Chapter 4. In particular we are interested in evaluating NR-P NORM performances using three different map of lost macroblocks (MLM) estimations:

- MLM estimated as the estimated thresholded likelihoods $\hat{P}^i(MSE_S^i|BC_S)$ and $\hat{P}^i(MSE_T^i|BC_T)$. The obtained NR-P method is called NR-P$_L$ NORM
- MLM estimated as the estimated thresholded posteriors $\hat{P}^i(BC_S|MSE_S^i)$ and $\hat{P}^i(BC_T|MSE_T^i)$. The obtained NR-P method is called NR-P$_P$ NORM
- MLM estimated as the output of Min-Cut/Max-Flow algorithm. The obtained NR-P method is called NR-P$_{MRF}$ NORM

We then carried out several experiments on real video sequences simulating the error prone channel and running the described NR-P NORM methods to estimate the induced channel distortion. In the next section we will present the test dataset, and in the following one the obtained estimated channel distortions with our three NR-P methods.

5.1 Source Coding Conditions

We consider a typical scenario in video coding applications, the internet protocol television (IPTV). H.264/AVC reference software (version JM12.3 (JVT) has been used with main profile. Three CIF video sequences, *Mobile, Foreman, News* have been coded at 256 kbps and 30 fps with a fixed quantization parameter for I and P slices (QP = 36). The number of reference frames used during prediction is fixed to 5 and rate distortion optimization (RDO) is enabled. Finally the used motion estimation algorithm for inter prediction is the simplified UMHexagon Search.

Each coded frame is partitioned into slices, where each slice contains a horizontal row of macroblocks. Each coded slice is then packetized according to the real-time transfer protocol (RTP) specifications Wenger (2003). The simulated error-prone channel drops coded packets according to a packet loss rate (PLR) in the range [0.1 10]. The error patterns have been generated using a two-state Gilbert's model Gilbert et al. (1960) with average burst length of three packets. We simulated the transmission of the test sequences over 15 channel realizations for each considered PLR value [0.1 0.4 1 3 5 10]. This dataset was already used in Section 3.1 to analyze the impact of motion vector estimation.

The three chosen sequences have different types of contents. In particular *Mobile* is a sequence with high motion wrt *News*. On the other hand *Foreman* is a mixture of high motion frames followed by a static scene. Since our NR-P methods are dependent from the particular video contents this dataset permits to analyze the behavior of our system in these different contexts.

5.2 Experimental Result and Discussion

In order to evaluate the accuracy of our NR-P methods we measured the Pearson's correlation coefficients between the estimated and real MSE distortions. Two granularity levels are taken into account frame and sequence (Seq.). Finally also the NR-P/NR-B NORM results are presented as an upper bound for our estimations. From tabs. 5.1, 5.2 and 5.3 it is possible to appreciate that the NR-P/NR-B NORM approach is clearly the one that performs better, because it has access to the original motion vectors and prediction residuals and, specifically, to the true map of channel errors.

As expected the NR-P$_P$ method achieves higher correlation coefficients wrt the NR-P$_L$ one, in particular for *News* sequence. It must be remembered that the NR-P$_P$ method takes advantage of the confidence map that is able to lower the number of false positives. This is particulary evident looking at the scatter plots at frame level depicted in Figure 5.3. It is possible to appreciate how the estimations obtained by NR-P$_P$ are nearer to the 45° line, in particular at low PLR (0.1% 0.4%), wrt the ones obtained by NR-P$_L$.

However, as stated in Section 4.5, for high motion sequences like *Mobile* and *Foreman* the confidence map is not used in P frames due to its poorer results. It is so trivial to understand that are static sequences like *News* the ones that take greater advantages from NR-P$_P$ method. Moreover as shown in Figures 4.34, 4.35, 4.36 the impact of confidence usage in I frames, is greater for static sequences. Also in this case, these considerations are confirmed by scatter plots at frame level presented in Figures 5.2, 5.1. It is in fact possible to see that results obtained by NR-P$_L$ and NR-P$_P$ are comparable for high motion sequences like *Mobile* and *Foreman*.

The NR-P$_{MRF}$ method archives higher correlation coefficient wrt NR-P$_P$ and NR-P$_L$ methods (tabs. 5.1, 5.2 and 5.3). In particular from scatter plots in Figures 5.4, 5.5, 5.6 it is possible to appreciate that the obtained dispersions,with NR-P$_{MRF}$ method, are much more concentrated around the 45° line.

However it must be noticed that for *Foreman* sequence, the correlation coefficients obtained at frame level by the NR-P$_{MRF}$ method are lower wrt the ones obtained by the other two NR-P methods. This behavior is due to a to wrong estimation of the spatial prior during changes of scene, in fact it must be remembered that spatial priors are computed over the previous motion compensated P frame. When a change of scene occurs, the previous frame is no more a good predictor of the noiseless frame at time t and false positives may arise in the estimated map of lost macroblocks. Since the MRF takes advantage from spatial relationship between adjacent blocks, it may happen that false positives induce a wrong relationship wrt their neighbors, creating a wrong MLM estimation.

Finally it is possible to appreciate from scatter plots at sequence level in Figure 5.7 that all the three proposed methods achieve high correlation coefficients (always above 0.96) at Seq. level. However the dispersions related to the NR-P$_{MRF}$ method are nearer to the 45° line, which means that this method achieves a more precise estimation of the real channel induced distortion.

	NR-P$_L$		NR-P$_P$		NR-P$_{MRF}$		NORM	
PLR [%]	Frm	Seq	Frm	Seq	Frm	Seq	Frm	Seq
0.1	0.84	0.99	0.84	0.99	0.86	0.98	0.99	0.99
0.4	0.81	0.98	0.81	0.98	0.84	0.98	0.98	0.98
1	0.82	0.94	0.82	0.94	0.93	0.98	0.98	0.98
3	0.95	0.96	0.95	0.96	0.95	0.97	0.99	0.99
5	0.92	0.92	0.92	0.92	0.94	0.95	0.99	0.99
10	0.87	0.87	0.87	0.87	0.89	0.87	0.97	0.98
Tot	0.94	0.98	0.94	0.98	0.94	0.98	0.99	0.99

Table 5.1: Correlation coefficients with different NR-P and NR-P/B methods wrt real distortion for *Mobile* sequence

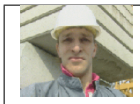	NR-P$_L$		NR-P$_P$		NR-P$_{MRF}$		NORM	
PLR [%]	Frm	Seq	Frm	Seq	Frm	Seq	Frm	Seq
0.1	0.67	0.96	0.67	0.96	0.57	0.98	0.98	0.98
0.4	0.66	0.75	0.66	0.76	0.57	0.78	0.94	0.96
1	0.87	0.91	0.87	0.91	0.86	0.96	0.96	0.96
3	0.86	0.87	0.87	0.88	0.84	0.87	0.96	0.97
5	0.88	0.92	0.89	0.92	0.87	0.92	0.96	0.97
10	0.84	0.91	0.83	0.92	0.85	0.94	0.93	0.93
Tot	0.88	0.98	0.88	0.98	0.88	0.98	0.96	0.99

Table 5.2: Correlation coefficients with different NR-P and NR-P/B methods wrt real distortion for *Foreman* sequence

	NR-P$_L$		NR-P$_P$		NR-P$_{MRF}$		NORM	
PLR [%]	Frm	Seq	Frm	Seq	Frm	Seq	Frm	Seq
0.1	0.51	0.98	0.98	0.99	0.97	0.99	0.99	0.99
0.4	0.67	0.88	0.87	0.93	0.83	0.95	0.98	0.98
1	0.87	0.92	0.90	0.94	0.91	0.97	0.99	0.99
3	0.72	0.89	0.79	0.92	0.92	0.95	0.98	0.99
5	0.70	0.76	0.80	0.81	0.89	0.91	0.96	0.96
10	0.66	0.90	0.74	0.91	0.88	0.93	0.97	0.98
Tot	0.71	0.96	0.80	0.97	0.91	0.98	0.98	0.99

Table 5.3: Correlation coefficients with different NR-P and NR-P/B methods wrt real distortion for *News* sequence

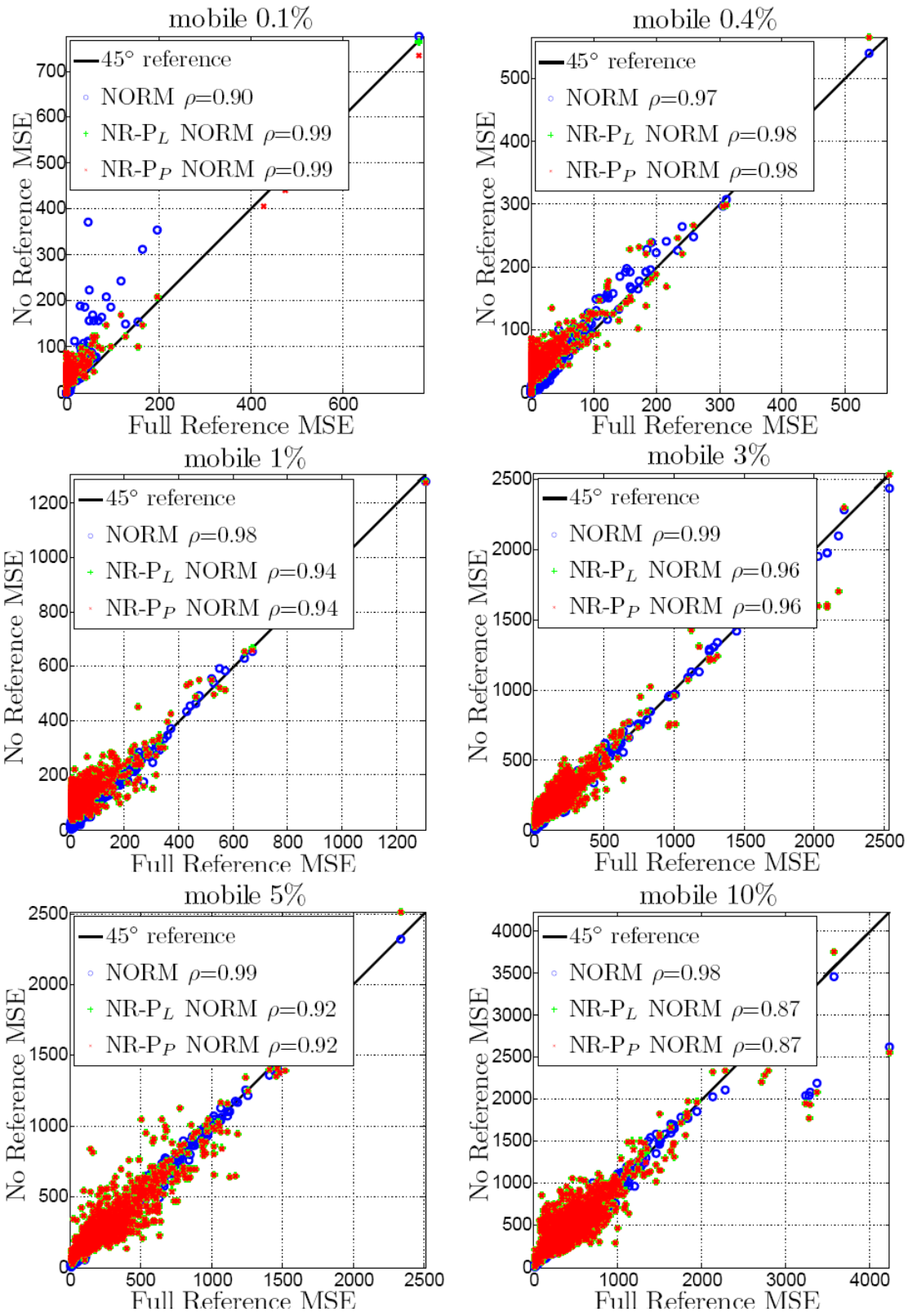

Figure 5.1: NR-P$_P$ and NR-P$_L$ scatter plots at frame level for *Mobile* sequence at different PLRs

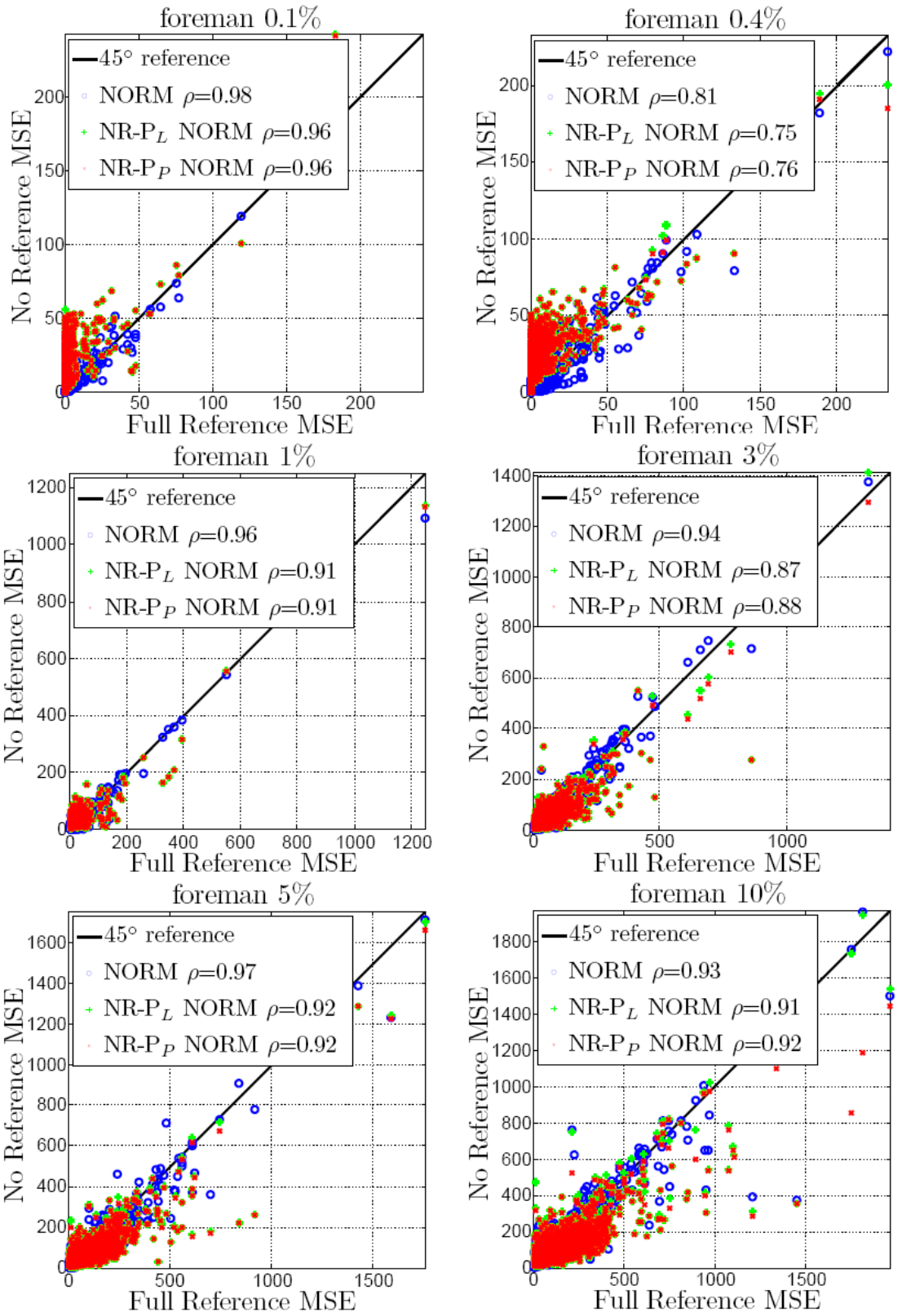

Figure 5.2: NR-P$_P$ and NR-P$_L$ scatter plots at frame level for *Foreman* at different PLRs

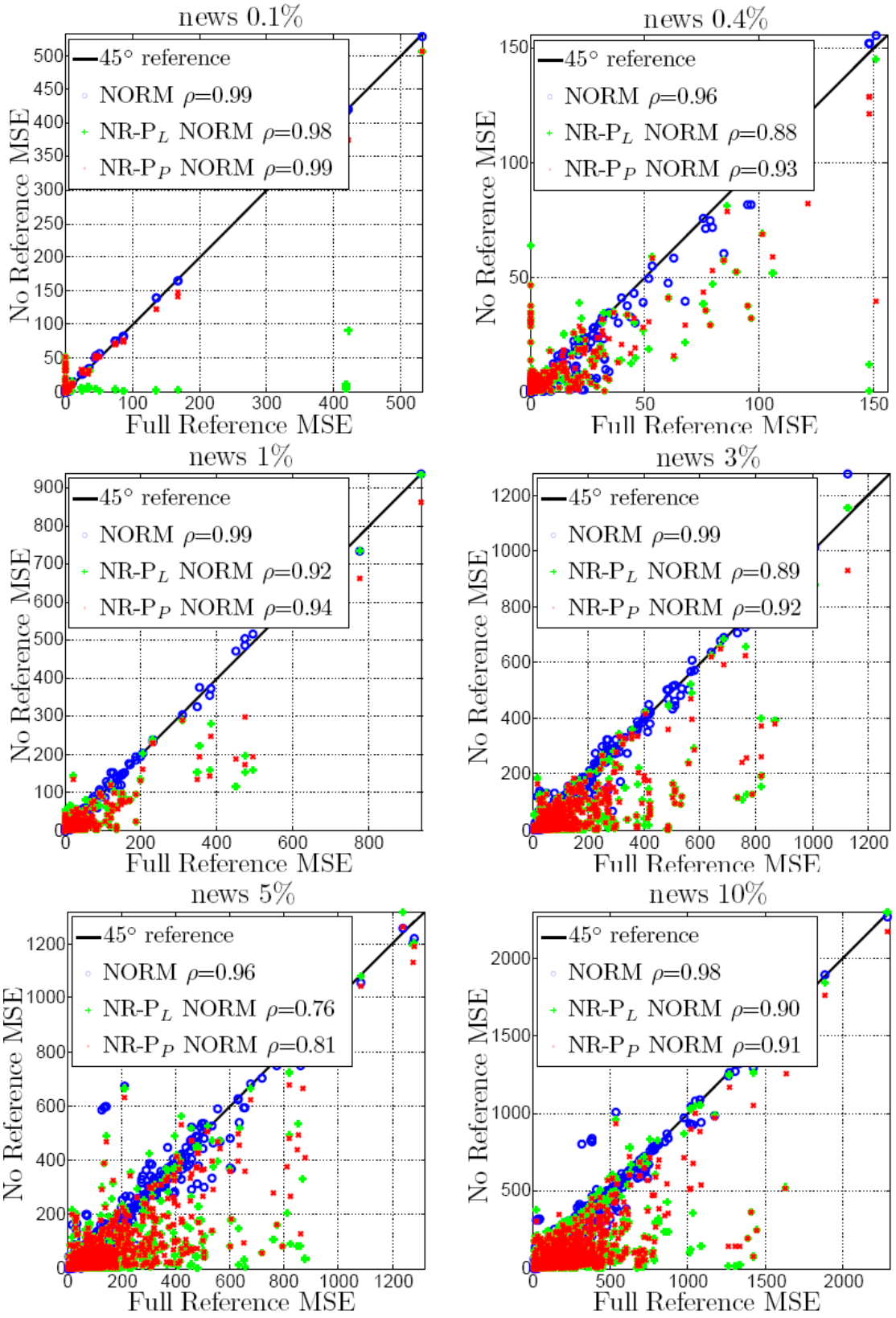

Figure 5.3: NR-P_P and NR-P_L scatter plots at frame level for *News* at different PLRs

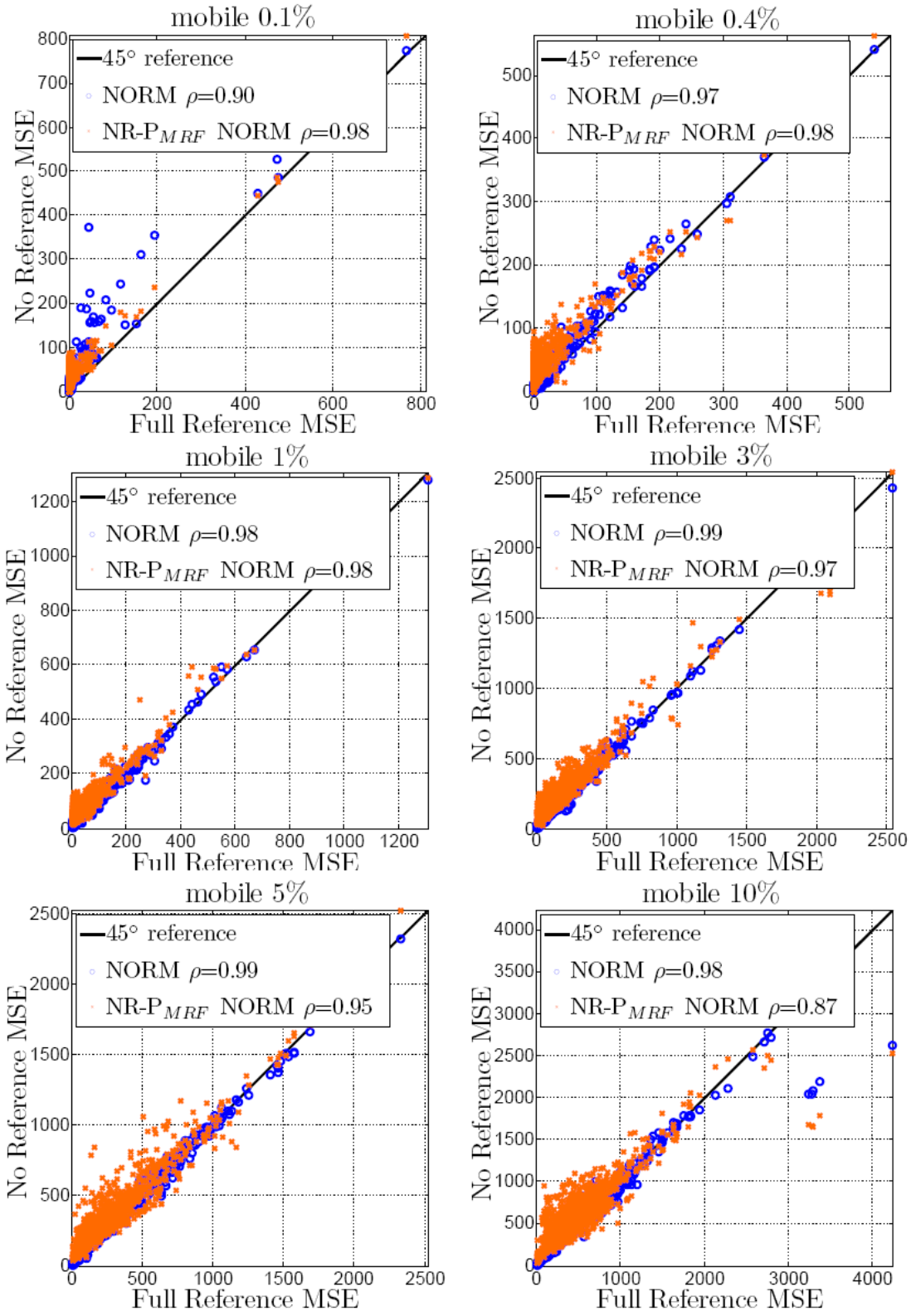

Figure 5.4: NR-P$_{MRF}$ scatter plots at frame level for *Mobile* at different PLRs

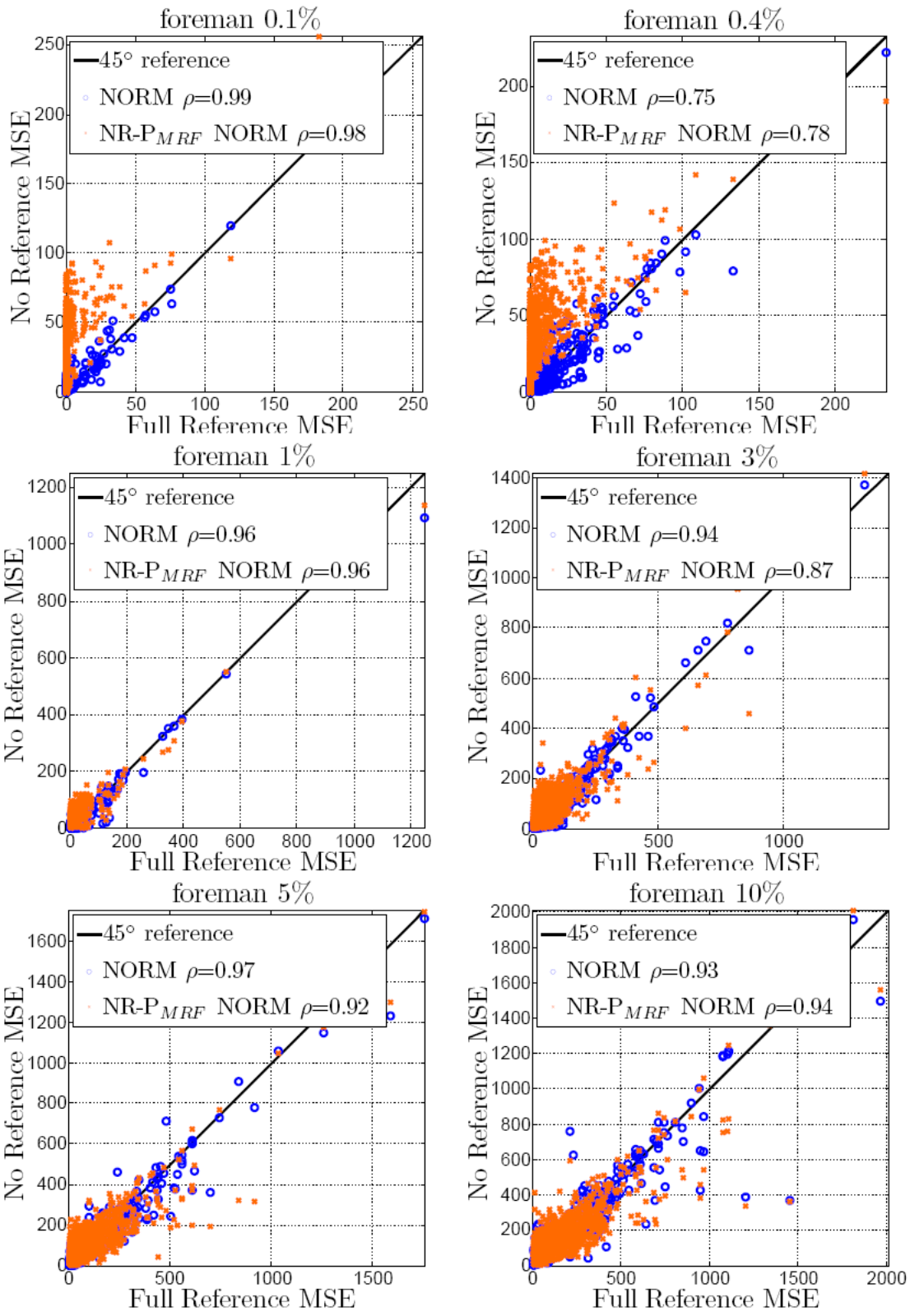

Figure 5.5: NR-P$_{MRF}$ scatter plots at frame level for *Foreman* at different PLRs

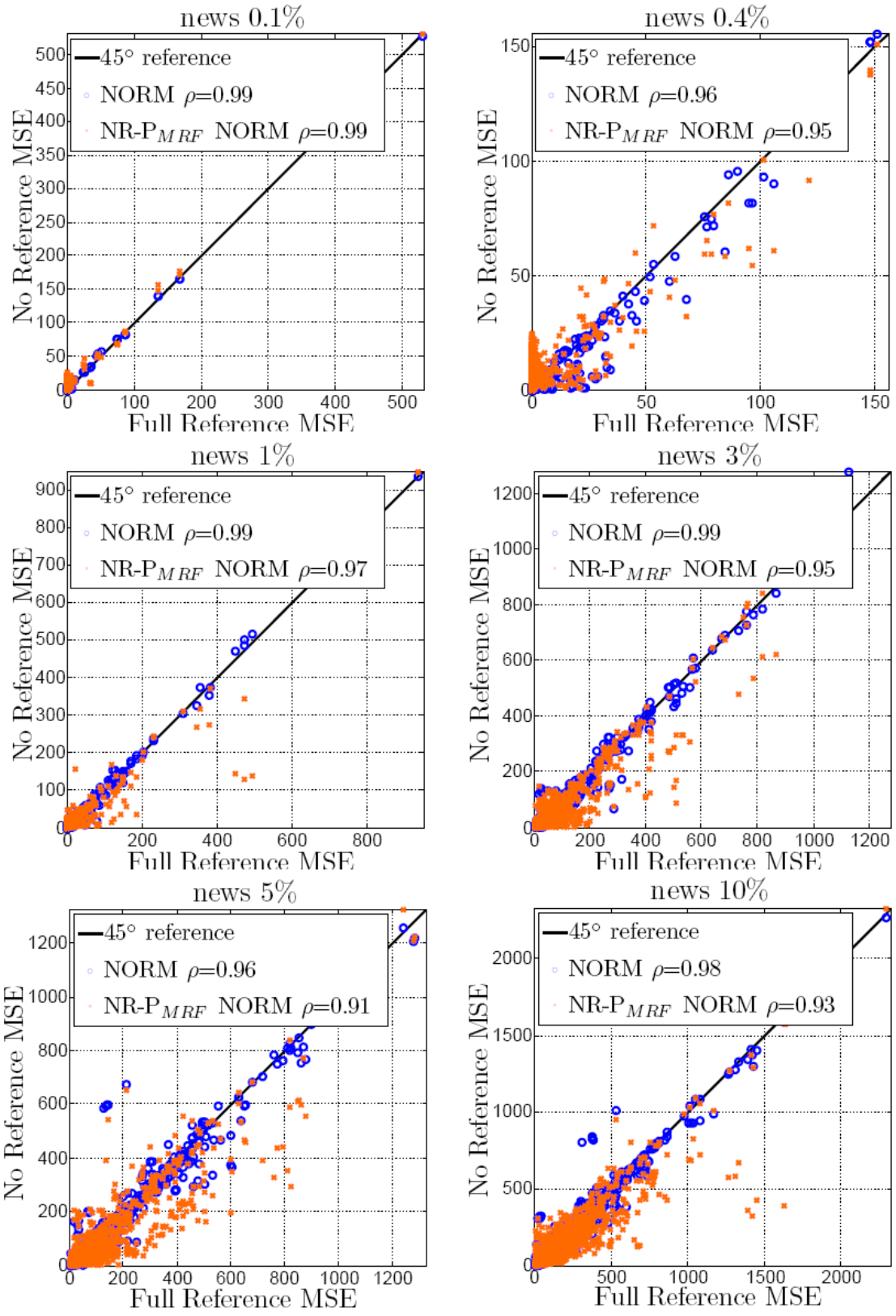

Figure 5.6: NR-P$_{MRF}$ scatter plots at frame level for *News* at different PLRs

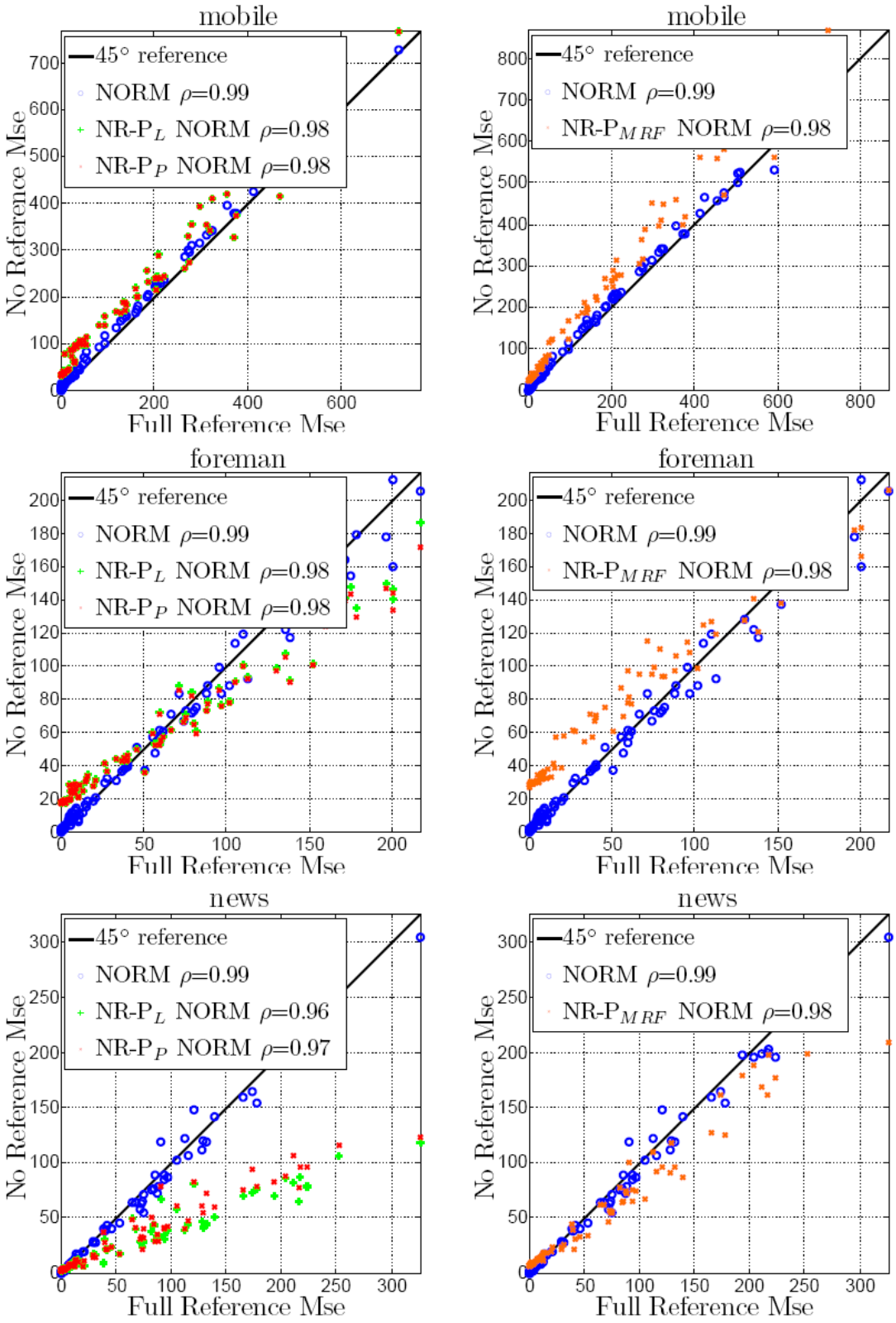

Figure 5.7: Scatter plots at sequence level for all the sequences under exam

CHAPTER 6

CONCLUSIONS

In this chapter we summarize the presented work, briefly describing the different steps to reach the NR-P video quality monitoring. Finally we discuss some future works on the theme of no reference pixel based video quality monitoring.

6.1 NR-P Video Quality Monitoring

In this section we will briefly review the created system composed by all the parts described in the previous chapters. The main objective of our work is to create a NR-P version of NORM. To achieve this objective several bitstream information must be estimated:

- Motion Vectors
- Residuals
- Structure of the Group Of Pictures (GOP)
- Map of Lost Macroblocks (MLM)

Motion vectors and residuals are computed performing a motion block estimation algorithm over the corrupted decoded video. GOP structure is estimated using a *QP* estimator for B frames, while I frames are recognized as frames with an high number of macroblocks having a temporal prediction residuals' energy greater of a given threshold.

Map of lost macroblocks points out, for each frame, which macroblocks have been lost during transmission. This information is crucial for our objective since its impact over the final estimated distortion is greater wrt other NORM inputs estimations. We so decide to focus our attention over this particular problem.

Since it may happen that the concealment algorithms are able to perfectly restore lost macroblocks, the MLM estimation problem is ill posed. Our new objective is so to recognize temporally and spatially badly concealed macroblocks without knowing the particular adopted concealment algorithm. Steps for map of badly concealed macroblock estimation can be summarized as follows:

- Likelihoods $P^i(MSE_S^i|BC_S)$ and $P^i(MSE_T^i|BC_T)$ are estimated for each i^{th} macroblock:
  - MSE_S^i is the residuals' energy computed between the spatially concealed i^{th} macroblock and i^{th} macroblock reconstructed by neighbor macroblocks boundaries.
  - MSE_T^i is the residuals' energy computed between the temporally concealed i^{th} macroblock and the temporal prediction of the i^{th} macroblock.
- Priors $P^i(BC_S)$ and $P^i(BC_T)$ are computed over the previous motion compensated P frame:
  - $P^i(BC_S)$ is estimated thanks to the MSE_S^i computed over the previous motion compensated P frame.
  - $P^i(BC_T)$ is estimated thanks to the $MVPD$ computed over the previous motion compensated P frame.
- Posteriors are then computed as the product between priors and likelihoods, but for sequences with a complex motion $P^i(BC_T)$ is neglected.
- T-links and n-links are then calculated from the obtained posteriors (eqs. 4.39, 4.40). The Min-Cut/Max-Flow algorithm is then ran over the chosen graph model to exploit the spatial relationship of lost macroblocks.

- The output of the algorithm is the estimated map of badly concealed macroblocks.

The different estimated data are used to feed NORM algorithm creating a no reference pixel based (NR-P) video quality monitoring. An overview of the final system is presented in Figure 6.1.

6.2 Conclusion and Future Developments

The presented work tries to solve the no reference pixel based video quality monitoring problem. We were interested in estimating the channel induced distortion using only the reconstructed video at the decoder side. To reach this objective we decided to use NORM, which is a no reference hybrid algorithm for channel distortion estimation. NORM is a NR-B/NR-P method that uses both bitstream and coded video information. In our target scenario the bitstream is unavailable and we decided to estimate NORM bitstream inputs in order to create an NR-P version of the same algorithm. This objective was reached estimating NORM bitstream inputs. In particular we focused our attention over the map of lost macroblocks estimation problem. Starting from mild assumptions, valid for a large class of concealment algorithms, we found features able to recognize lost macroblocks, using also a confidence to clean the obtained map from false positives. Finally we used a Max-Flow/Min-Cut algorithm to exploit spatial relationship between lost blocks, reaching the final classification. The NORM algorithm is then ran using the estimated bitstream information, obtaining a no reference pixel-based video quality monitoring.

Future works will be dedicated to search for new confidence features to obtain better channel distortion estimations at macroblock level for P frames, trying to use information related to side match distortion. Moreover different lattices for MRF will be defined, trying also to estimate the slices structure from the topology of map of lost macroblocks.

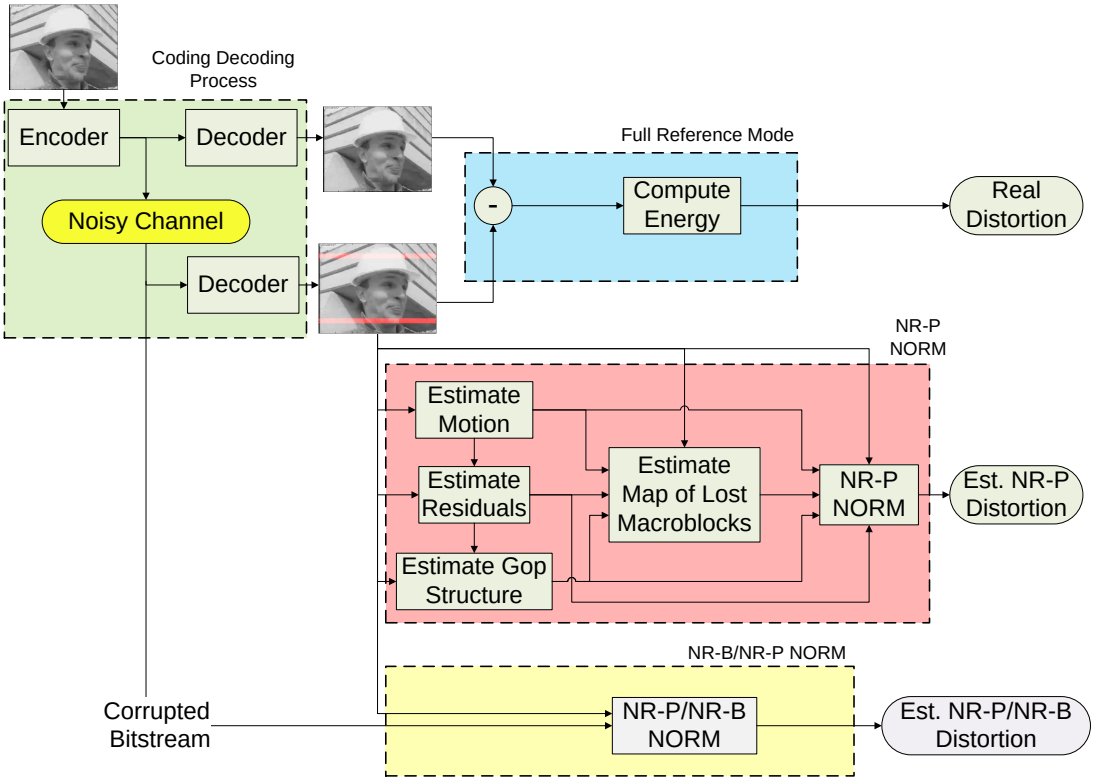

Figure 6.1: Overview of the final system

Bibliography

Boykov, Y. and Funka-Lea, G. (2006). Graph cuts and efficient nd image segmentation, *International Journal of Computer Vision* **70**(2): 109–131.

Boykov, Y. and Kolmogorov, V. (2001). An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision, *Energy minimization methods in computer vision and pattern recognition*, Springer, pp. 359–374.

Boykov, Y., Veksler, O. and Zabih, R. (1998). Markov random fields with efficient approximations, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Citeseer, pp. 648–655.

BT 500-10, I. (2002). Methodology for the subjective assessment of the quality of television pictures., *ITU, Geneva, Switzerland* .

Fawcett, T. (2004). ROC Graphs: Notes and Practical Considerations for Researchers, *Machine Learning* **31**.

Gilbert, E. et al. (1960). Capacity of a burst-noise channel, *Bell Syst. Tech. J* **39**(9): 1253–1265.

Greig, D., Porteous, B. and Seheult, A. (1989). Exact maximum a posteriori estimation for binary images, *Journal of the Royal Statistical Society. Series B (Methodological)* **51**(2): 271–279.

He, Z., Cai, J. and Chen, C. (2002). Joint source channel rate-distortion analysis for adaptive mode selection and rate control in wireless video coding, **12**(6): 511–523.

ITU (2003). *Information Technology - Coding of Audio-visual Objects - Part 10: Advanced Video Coding*. ISO/IEC International Standard 14496-10:2003.

(JVT), J. V. T. (n.d.). H.264/AVC reference software version JM14.2. downloadable at http://iphome.hhi.de/suehring/tml/download/.

K. Stuhlmuller, N. Farber, M. L. and Girod, B. (2000). Analysis of video transmission over lossy channels, *IEEE J. Sel. Areas Commun* **18**(6): 1012–1032.

Katsaggelos, A. and Galatsanos, N. (1998). *Signal recovery techniques for image and video compression and transmission*, Kluwer Academic Publishers Norwell, MA, USA.

Lam, W., Reibman, A. and Liu, B. (1993). Recovery of lost or erroneously received motion vectors, *1993 IEEE International Conference on Acoustics, Speech, and Signal Processing, 1993. ICASSP-93.*, Vol. 5.

Marziliano, P., Dufaux, F., Winkler, S. and Ebrahimi, T. (2002). A no-reference perceptual blur metric, *Proc. of IEEE Int. Conf. on Image Processing*, Vol. 3, Rochester, NY, USA, pp. 57–60.

N. Farber, K. S. and Girod, B. (1999). Analysis of error propagation in hybrid video coding with application to error resilience, *IEEE International Conference Image Processing*, Kobe, JP.

Naccari, M., Tagliasacchi, M. and Tubaro, S. (2009). No-reference video quality monitoring for H.264/AVC coded video, **11**(5): 932–946.

P 910, I. (2008). Subjective video quality assessment methods for multimedia applications internation comunication union., *ITU, Geneva, Switzerland* .

R. Zhang, S. L. R. and Rose, K. (2000). Video coding with optimal inter/intra-mode switching for packet loss resilience, *IEEE J. Sel. Areas Commun* **18**(6): 966–976.

Reibman, A. R., Vaishmpayan, V. A. and Sermadevi, Y. (2004). Quality monitoring of video over a packet network, **6**(2): 327–334.

S. Kanumuri, P. C. Cosman, A. R. R. and Vaishmpayan, V. A. (2006). Modeling packet-loss visibility in mpeg-2 video, *IEEE Trans. Multimedia* **8**(2): 341–355.

Sullivan, G. J., Wiegand, T. and Lim, K.-P. (2003). Joint model reference encoding methods and decoding concealment methods, *Technical Report JVT-I049*, Joint Video Team (JVT).

T. Shu, J. A. and Gu'erin, R. (2005). Real-time monitoring of video quality in ip networks, *International Workshop on Network and Operating System Support for Digital Audio and Video*, Stevenson, WA, USA.

Tagliasacchi, M. and Tubaro, S. (n.d.). Blind Estimation of the QP parameter in H. 264/AVC decoded video.

Tan, K. and Ghanbari, M. (2000). Frequency domain measurement of blockiness in MPEG-2 coded video, *Proc. of IEEE Int. Conf. on Image Processing*, Vol. 3, Vancouver, Canada.

Wenger, S. (2003). H. 264/AVC over IP, **13**(7): 645–656.

Winkler, S. (2009). Video quality measurement standards - current status and trends, *Proc. International Conference on Information, Communications, and Signal Processing (ICICS)*, pp. 7–10.

Yamada, T., Miyamoto, Y. and Serizawa, M. (2007). No-reference video quality estimation based on error-concealment effectiveness, *IEEE Packet Video*, Lausanne, Switzerland.

Yamada, T., Yachida, S., Senda, Y. and Serizawa, M. (2010). Accurate video-quality estimation without video decoding, Dallas, TX, USA.

Yang, H. and Rose, K. (2007). Advances in recursive per-pixel end-to-end distortion estimation for robust video coding in h.264/avc, *IEEE Trans. Circuits Syst. Video Technol* **17**(7): 845–856.

Estratto in italiano

L'utilizzo di reti IP per la trasmissione di contenuti multimediali è ormai una realtà consolidata. Solitamente le reti utilizzate forniscono un servizio *best effort*, vale a dire che non vi è sicurezza riguardo la ricezione dei contenuti inviati. In tali circostanze l'utente e il fornitore dei servizi potrebbero essere interessati a stipulare un contratto di tipo SLA (service level agreement) che fissi la qualità del video percepita a lato utente. Pertanto è di estrema importanza avere una misura della qualità del video ricostruito dal decoder. Solitamente i contenuti video vengono compressi da un encoder prima dell'invio al decoder che attua la ricostruzione, il video così ricevuto può contenere due tipi differenti di distorsione, di quantizzazione e di canale. La distorsione di quantizzazione è dovuta alla codifica *lossy* applicata al video, mentre l'errore di canale è imputabile a perdite di canale avvenute durante la trasmissione. Generalmente il decoder cerca di ricostruire i macroblocchi persi tramite un algoritmo di concealment. Tali algoritmi utilizzano le informazioni correttamente ricevute o già recuperate per produrre una versione *ricostruita* del macroblocco perso. Tuttavia la ricostruzione non sempre raggiunge gli standard qualitativi adeguati e la distorsione di canale sarà quindi percepibile a livello visivo.

Il nostro obiettivo principale è la stima della distorsione di canale lato decoder utilizzando solo le informazioni presenti nel video ricostruito, ovvero un metodo *no reference pixel based (NR-P)*. Solitamente gli algoritmi presenti in letteratura utilizzano anche il contenuto del *bitstream* (vettori di moto, residui di predizione,etc...), purtroppo non sempre questo approccio è applicabile poiché il *bitstream* potrebbe essere criptato o ottenuto da decoder di terze parti. In tali casi sono disponibili solo i valori dei pixel del video ricostruito.

L'obiettivo finale è stato raggiunto creando una versione NR-P di NORM (No-Reference video quality Monitoring); un algoritmo in grado di stimare la distorsione di canale lato decoder utilizzando informazioni provenienti dal video decodificato e dal bitstream. I parametri provenienti dal bitstream sono stati stimati dal video codificato con particolare attenzione per la mappa dei macroblocchi persi. Partendo da assunzioni valide per la gran parte degli algoritmi di concealment si è cercato di riconoscere i macroblocchi mal ricostruiti a livello visivo. Ogni frame è stato esaminato ricercando i macroblocchi mal ricostruiti attraverso l'energia dei residui di predizione, a cui è stata in seguito applicata una confidenza in grado di evitare falsi positivi. Infine la relazione spaziale tra macroblocchi persi è stata modellizzata tramite *Markov random fields*.

Le differenti stime degli input provenienti dal bitstream sono state poi utilizzate per l'esecuzione dell'algoritmo NR-P NORM. Le distorsioni di canale stimate sono ben correlate (coefficiente di correlazione lineare maggiore di 0.9) rispetto alla distorsione reale di canale calcolata a livello di frame e sequenza.