Circuits Syst Signal Process (2013) 32:815–824
DOI 10.1007/s00034-012-9504-0

Spatial Error Concealment Based on Edge Visual Clearness for Image/Video Communication

Ján Koloda · Victoria Sánchez · Antonio M. Peinado

Received: 24 February 2012 / Revised: 20 September 2012 / Published online: 12 October 2012

Abstract In this paper, we propose a technique for concealing missing image/video blocks based on the concept of visual clearness of an edge. A scanning procedure based on the Hough transform allows us to find the relevant edges, and the visually clearest ones are employed in an interpolation based reconstruction. Specifically, several interpolations are combined according to a set of weights which allows the reconstruction of more complex textures. These weights are derived from the visual clearness associated to an edge and are unique for every pixel within the missing macroblock. The resulting algorithm is quite efficient, simple, and competitive in comparison with other state-of-the-art techniques.

Keywords Error concealment · Directional interpolation · Block-coded image/video · Hough transform

1 Introduction

Multimedia transmission applications are prone to suffer from deterioration of QoS. Due to strict real-time requirements, the retransmission of lost or severely damaged packages can be impossible. The block-based video coding standard H.264/AVC has introduced several error resilience mechanisms that draw on specific data organization tools such as network abstraction layer units (NALU), flexible macroblock order-

J. Koloda (✉) · V. Sánchez · A.M. Peinado
Department of Signal Theory, Networking and Communications and CITIC-UGR, Universidad de Granada, 18071 Granada, Spain
e-mail: janko@ugr.es

V. Sánchez
e-mail: victoria@ugr.es

A.M. Peinado
e-mail: amp@ugr.es

ing (FMO), or arbitrary slice ordering (ASO) [4, 17]. These tools allow the decoder to apply error concealment (EC) algorithms [8], in order to achieve an acceptable visual quality of the received stream.

The EC algorithms benefit from the fact that video signals are highly correlated, spatially and/or temporally. This criterion is used to classify the EC algorithms into two groups: spatial EC (SEC), which utilizes only the information provided by the current frame, and temporal EC (TEC), which makes use of temporal information such as motion vectors (MV). This classification is nonexcluding, and combining temporal and spatial information leads to significant improvements [6]. However, the most extended block-based coding standards, such as H.264 and MPEG-4, use both intracoding (I-frames) and prediction (P/B-frames). Since the intracoded frames serve as a "firewall", that separates visually different scenes or resets the prediction error, utilizing temporal information for their concealment could be risky. Therefore, SEC techniques are the most suitable choice to conceal the I-frames [17]. Moreover, these frames are employed as templates to predict several consecutive P/B-frames [4], so a poor reconstruction of one I-frame would distort not only the frame itself but the sequence of several frames in a row.

A simple technique for spatial concealment is bilinear interpolation [11]. Since high-frequency features, such as edges, are visually more relevant than uniform textures, more advanced interpolation techniques have been proposed exploiting directional features in the neighborhood of the missing macroblock [5]. A reconstruction of broken edges in the transformed domain is treated in [13]. In [9], the Hough transform, a powerful tool for edge description, was used to set the angle for the interpolation. However, the performance drops when multiple edges need to be connected. A more robust approach is suggested in [2] permitting one to connect several edges. Nevertheless, the technique is highly adapted to consecutive block loss, and so it may suffer from a lack of generality. Moreover, the applied interpolation process is rather simple and is unable to restore more complicated edges and textures. Restoration of broken edges based on extrapolation is introduced in [19]. A block-matching technique with a decision algorithm is treated in [10]. The algorithm restores successfully fine textures although correct classification of the missing macroblock is crucial. In addition, trying to match the entire macroblock may generate artificial edges (the so-called blocking). This is partially solved by utilizing block-based bilateral filtering [18]. Inpainting-based techniques are also used for quality texture reconstruction [3]. Modeling an image as a Markov random field allowed the authors in [12] to implement an efficient concealment algorithm. However, the reconstruction of high-frequency features tends to oversmoothing.

In this paper, the concept of visual clearness associated to an edge is introduced, and the most visually relevant edges are utilized for concealing missing image/video blocks based on a weighted combination of directional interpolations. Furthermore, we consider that pixels in the corrupted macroblock are not equally affected by the interpolations, and so pixel-dependent weights need to be assigned when combining these interpolations. The resulting technique is able to restore complicated edges and more complex textures, thus providing high-quality reconstructions.

The paper is organized as follows. In Sect. 2, the concept of visual clearness associated to an edge is introduced. Reconstruction based on the visually clearest direc-

tions is described in Sect. 3, and simulation results and comparisons with other SEC techniques are presented in Sect. 4. The last section is devoted to conclusions.

2 Visual Clearness Associated to an Edge

In this section, we establish a criterion to assess the importance of an edge. This criterion is based on the visual clearness of a given edge. At the end of the section, we will be able to introduce the parameter representing this visual clearness.

In order to obtain the relevant directions, it is necessary to explore first the directional behavior in the neighborhood of the missing macroblock. Thus, an edge detection must be applied first to provide a binary edge map. For this purpose, we use one of the most simple and efficient procedures, the Canny's edge detector [1]. In comparison with other detectors, such as Sobel's one, it is less sensible to noise as it first smooths the image by filtering it with a Gaussian kernel. Moreover, due to its nonmaximum suppression feature, the detected edges are clear, and no thinning algorithm needs to be applied.

Next, the predominant directions need to be computed. For this purpose, a gradient-based voting mechanism [10] has been widely used. However, the angular resolution of such a technique is rather poor, and edges with the same directions but different spatial location are treated as a single line. Instead, a Hough transform-based procedure is applied.

The Hough transform provides simple yet powerful descriptors and is based on the fact that many shapes can be expressed in a parametric form. In this paper, for the sake of simplicity, we use a linear kernel that assumes that any line can be expressed as

$$\rho = x\cos\theta + y\sin\theta, \tag{1}$$

where ρ is the perpendicular distance between the line and the origin, and θ is the slope of the normal. The Hough transform is applied to the binary image provided by the edge detector. Thus, Eq. (1) involves that every pixel (x, y) that belongs to a linear edge produces the same (ρ, θ). Therefore, the set of pixels that comprise a linear segment are transformed into a single point with position (within the transformed matrix) indicated by the parameters ρ (row) and θ (column). Thus, the bidimensional spatial domain (x, y) is transformed into a new domain (ρ, θ) where the transform value at each point (Hough coefficient, $H(\rho, \theta)$) is directly related to the number of pixels contained in the segment defined by (ρ, θ).

Directional interpolation is based on the directional behavior of the missing macroblock. Given the high spatial correlation, this behavior can be deduced from that of its neighborhood. To explore it, a scanning is carried out as shown in Fig. 1. The proposed scanning consists in moving a mask with the same dimensions as the lost macroblock, pixel by pixel (scanning step of 1 pixel) along its four sides: top (Fig. 1(b)), left, bottom, and right. At each step (or mask position), the Canny's edge detector is applied, and the Hough transform is calculated over the corresponding binary image in order to find relevant edges. An edge is said to be relevant if its prolongation crosses the missing macroblock. The direction of every relevant edge, θ, is stored in

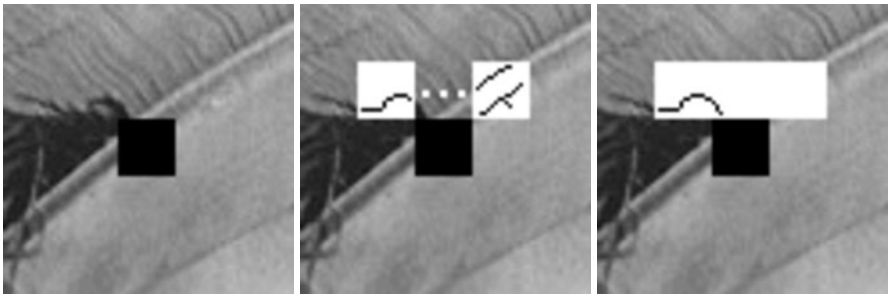

Fig. 1 Edge detection. (**a**) Received image with a missing macroblock, (**b**) top side scanning, (**c**) top side edge detection without scanning

a set $\mathcal{D}$. Note that if the available neighborhood were treated as a whole (as in [2]), extremely strong edges would mask other relevant ones when applying thresholding in the edge detection process (Fig. 1(c)). The scanning procedure isolates those edges, making the edge detection more robust.

The candidates for relevant edges are found as follows. First, the strongest direction in every mask position is selected from the transformation matrix (the strength of a direction is expressed by the Hough coefficient value). Then, using the information provided by the Hough coefficient coordinates, θ (slope) and ρ (offset), we can determine whether the edge crosses the corrupt macroblock. If it does, the edge is considered relevant and stored in $\mathcal{D}$, otherwise the direction is discarded, and the next strongest one is examined. If none of the directions satisfies the aforementioned condition, no direction is stored.

Due to the image resolution, perfectly straight lines might not lead to a single pair (ρ, θ). Moreover, curved edges are treated as a set of linear segments with similar (ρ, θ). Using a coarser resolution of the Hough transform, these (ρ, θ) pairs lead to a single value of H allowing the edge to be treated as a single line. However, too coarse resolutions introduce significant imprecision and should be avoided. In our simulations, pixel-by-pixel resolution is applied for ρ and steps of 2^o for θ.

Finally, we introduce the concept of visual clearness, σ_i, associated to an edge. In order to define the visual clearness σ_i of an edge i, let $\boldsymbol{E}_i$ be the set of all pixels that comprise the edge. We will then compute σ_i as the product of the strength of the edge direction and the visual separation between the regions at both of its sides. The strength of an edge is proportional to its associated Hough coefficient value (which is proportional to the edge length), and the visual separation is given by the average 2D spatial gradient per edge pixel,

$$\sigma_i = H_i \frac{1}{|\boldsymbol{E}_i|} \sum_{j \in \boldsymbol{E}_i}^{|\boldsymbol{E}_i|} \sqrt{dx_j^2 + dy_j^2}, \quad i = 1, \ldots, |\mathcal{D}|, \tag{2}$$

where H_i is the associated Hough coefficient, dx_j and dy_j are the horizontal and vertical gradients centred over the jth pixel of the edge, and $|\mathcal{D}|$ is the number of relevant directions found by the scanning procedure.

3 Reconstruction Based on the Visually Clearest Directions

Among the directions stored in the set $\mathcal{D}$, the N visually clearest ones are selected. That is, those edges with the N largest σ_i values are selected. The missing macroblock is reconstructed by combining directional interpolations based on these N selected directions. Previously, the N corresponding interpolations are computed as

$$I_i(x,y) = \frac{d_2}{d_1+d_2} p_1^{(i)} + \frac{d_1}{d_1+d_2} p_2^{(i)}, \quad i = 1,\ldots,N, \tag{3}$$

where the missing pixel $p(x,y)$ is replaced by a weighted mean of $p_1^{(i)}$ and $p_2^{(i)}$, the two closest pixels in the ith direction that have been correctly received and decoded. The variable d_1 (d_2) is the Euclidean distance between p and $p_1^{(i)}$ ($p_2^{(i)}$).

Clear lines tend to be visually more important, so the N interpolations are combined according to the visual clearness of their corresponding edges. Therefore, every interpolation I_i has an associated weight defined as

$$w_i = \frac{\sigma_i}{\sum_{j=1}^{N} \sigma_j}. \tag{4}$$

Scalar weights are widely used in state-of-the-art techniques that involve combination of interpolations [10]. In more complex environments where several directions are present, this approach leads to oversmoothing. The proposed algorithm fixes this problem by assuming that the interpolations are not equally relevant for every pixel of the missing macroblock. In fact, pixels lying in the proximity of the prolongation of an edge tend to be more influenced by the interpolation defined by the corresponding direction. Therefore, every pixel has an associated weight computed as

$$\pi_i(x,y) = 1 - \delta_i^2(x,y), \tag{5}$$

where $\delta_i(x,y)$ is the normalized distance between the pixel $p(x,y)$ and the line defined by the ith edge. The normalization factor corresponds to the maximum distance between a pixel and a line within a macroblock, that is, to the length of its diagonal. The use of square power has been set heuristically since it provides better results.

Finally, the corrupt macroblock is reconstructed by means of a weighted superposition of N directional interpolations. That is, for every pixel $p(x,y)$ of the missing macroblock, we apply

$$p(x,y) = \sum_{i=1}^{N} \frac{w_i \pi_i(x,y)}{\sum_{j=1}^{N} w_j \pi_j(x,y)} I_i(x,y). \tag{6}$$

Some of the state-of-the-art algorithms, such as [2, 5], divide the available neighborhood into support regions. Thus, a pixel can be only interpolated relying on pixels within the same support region. This hard division, however, may create artificial borders and false textures. Weights $\pi_i(x,y)$ smooth the transitions from one region to another, preserving the continuity of the image signal.

Finally, it should be noted that N is the maximum number of directions to be used and is, by no means, mandatory. If there are fewer directions available or they are not visually clear enough, the number of interpolations to be considered will be less than N. The algorithm thus automatically adapts itself to the type of texture surrounding the missing macroblock.

In summary, for each missing macroblock, the proposed algorithm can be resumed as follows:

Step 1: perform the scanning at each side of the corrupt macroblock in order to obtain the set of relevant directions $\mathcal{D}$.

Step 2: compute the visual clearness for every relevant direction.

Step 3: select the N visually clearest directions and compute I_i $(i = 1, \ldots, N)$ as shown in Eq. (3).

Step 4: calculate the corresponding weighting factors ω_i and $\pi_i(x, y)$ for $i = 1, \ldots, N$ and for every missing pixel $p(x, y)$.

Step 5: combine the N directional interpolations corresponding to the N visually clearest directions by applying Eq. (6).

4 Simulation Results

The performance of the proposed algorithm is tested over the images of *Lena* (512×512), *Pirate* (1024×1024), the first frame of *Foreman* sequence (288×352), *Office* (592×896), *Airplane* (512×512), *Zelda* (512×512), *Boat* (512×512), and *House* (256×256). The test is carried out for macroblock dimensions of 16×16, and the rate of block loss is approximately 25 %, corresponding to a single packet loss of a frame with dispersed slicing structure [4]. The reconstruction quality of the proposed algorithm is compared with others, such as the bilinear interpolation (BIL) [11], projections onto convex sets (POC) [13], SEC based on the Hough transform (SHT) [2], content adaptive SEC (CAD) [10], directional extrapolation (EXT) [19], nonnormative SEC for H.264/AVC codec (AVC) [14], an adaptive Markov random fields method (MRF) [12], inpainting (INP) [3], and bilateral filtering [18].[1] Our proposal was tested for $N = 2$, $N = 3$, $N = 4$, and $N = 5$, where N is the number of interpolations combined according to Eq. (6). In addition, larger scanning steps of 2 pixels (N_2), 4 pixels (N_4), and 8 pixels (N_8) are also tested for $N = 5$.

In order to better take into account the perceptual quality, the multiscale structural similarity (MS-SSIM) index [16] is used for comparison along with the objective PSNR measure. Regarding MS-SSIM, the image is sequentially low-pass filtered and subsampled, and so a set of images is obtained, including the original resolution. Then, the SSIM index is applied for every subimage within the set. The SSIM index aims at approximating the human visual system (HVS) response looking for similarities in luminance, contrast, and structure [15]. This index can be seen as a convolution

[1] Implementations of most of these techniques, as well as the implementation of our algorithm, is available online at [20].

Table 1 PSNR values (in dB) and MS-SSIM values (scaled by 100) of test images reconstructed by several algorithms. The best performances for each image are shown in *boldface*

PSNR MS-SSIM	*Lena*	*Pirate*	*Foreman*	*Office*	*Airplane*	*Zelda*	*Boat*	*House*
BIL	30.00	27.82	27.12	27.54	25.58	33.44	26.95	26.83
	96.82	94.90	95.36	94.12	93.49	97.12	93.12	94.39
POC	28.04	26.42	28.49	27.56	26.18	29.91	26.05	27.38
	93.17	91.85	93.82	93.62	93.87	92.51	92.63	92.22
SHT	30.55	28.12	28.09	27.58	25.62	33.45	27.20	28.41
	97.13	95.18	95.94	94.09	93.41	97.22	93.21	96.20
CAD	31.96	28.44	34.85	29.29	27.41	34.01	27.73	31.11
	97.38	95.38	98.30	96.12	95.70	97.74	94.29	97.37
EXT	29.10	27.57	29.59	27.72	26.23	32.14	26.63	27.38
	95.83	94.89	97.18	94.84	95.36	96.73	94.01	93.98
AVC	30.42	28.74	29.11	29.99	27.79	34.43	28.25	28.37
	96.72	95.62	97.06	96.07	95.90	97.63	94.95	94.92
MRF	32.17	29.52	32.99	29.77	27.98	35.03	27.91	30.08
	97.75	96.33	98.21	96.45	96.12	98.03	94.92	96.89
INP	30.85	28.44	34.44	29.65	26.29	33.62	27.79	29.90
	97.17	95.27	98.35	96.66	94.91	97.44	95.22	96.97
BLF	32.15	29.36	34.75	30.06	28.34	33.83	28.37	30.52
	97.52	95.99	98.27	96.30	96.59	97.25	95.55	97.05
$N = 2$	32.46	29.60	34.93	30.98	28.51	35.07	28.44	31.30
	98.00	96.55	98.37	97.15	96.72	98.26	95.82	97.46
$N = 3$	32.70	29.90	35.09	31.34	28.71	35.44	28.66	31.45
	98.04	96.71	98.44	97.35	96.77	98.36	95.99	97.45
$N = 4$	32.74	29.98	35.15	31.44	28.76	35.67	28.77	31.47
	98.06	96.77	98.49	97.37	96.82	98.44	96.11	97.47
$N = 5$	**32.80**	**30.03**	**35.21**	**31.52**	**28.80**	**35.75**	**28.84**	**31.48**
	98.09	**96.80**	**98.53**	**97.43**	**96.82**	**98.46**	**96.11**	**97.50**
$N_2 = 5$	32.58	29.98	35.00	31.50	28.77	35.72	28.81	31.46
	97.91	96.53	98.40	97.39	96.66	98.36	96.03	97.36
$N_4 = 5$	32.53	29.96	34.97	31.49	28.74	35.61	28.73	31.46
	97.85	96.51	98.37	97.31	96.60	98.30	95.94	97.37
$N_8 = 5$	32.28	29.76	34.64	31.47	28.62	35.36	28.50	31.45
	97.66	96.48	98.22	97.33	96.64	98.24	95.69	97.36

of a fixed-sized mask with the residual error between the reference image and the concealed image [7]. A unique mask size is used for each of the images within the testing set. Thus, both fine and coarse textures and objects are taken into account.

The results in Table 1 show that the proposed algorithm outperforms the others for all the tested images both in terms of PSNR and MS-SSIM. Note that the algorithm performance saturates as N increases. In many cases, two directions are sufficient for

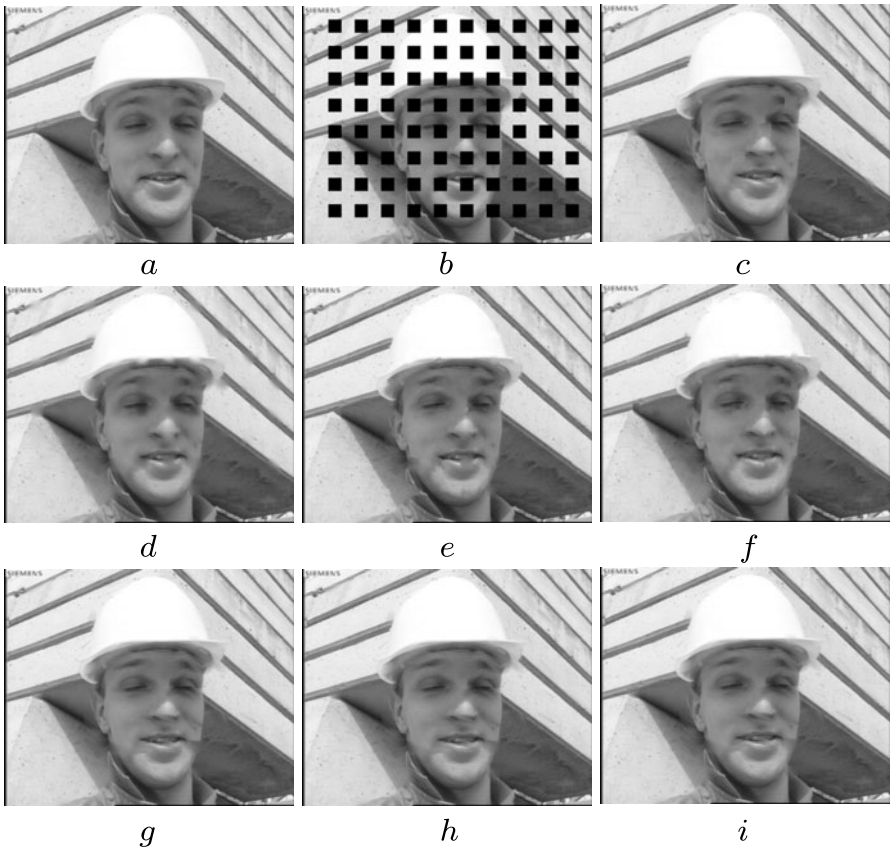

Fig. 2 Subjective comparison of different algorithms for 16×16 pixels macroblocks. (**a**) Original image, (**b**) corrupted image, (**c**) reconstructed image by CAD, (**d**) MRF, (**e**) INP, (**f**) BLF, (**g**) proposed algorithm with $N = 2$, (**h**) $N = 3$, and (**i**) $N = 4$

a good reconstruction, and utilizing more directions might not achieve any considerable improvement. In fact, Table 1 shows that even the simple reconstruction with two directions provides better reconstruction quality than all the other state-of-the-art techniques listed in the table. However, as a general rule, the more complex the texture, the more directions should be considered.

In order to better illustrate the subjective quality, Fig. 2 shows a comparison of the methods that provide the highest perceptual quality reconstructions (highest MS-SSIM) in Table 1. We see that the superiority of our algorithm, in terms of PSNR, is also corroborated at the subjective level.

Finally, the simulations reveal that the scanning procedure (including edge detection, Hough transform, and computing the visual clearness) comprises up to 90 % of the overall computational load. Using pixel-by-pixel scanning is relatively computationally expensive; however, in many cases, coarser resolutions would achieve almost identical results. Increasing the scanning step to 4 pixels, the processing time becomes similar or even lower than the processing time of more complex tested algo-

rithms such as CAD, INP, or BLF. By applying 4-pixel scanning step the reconstruction quality is reduced in roughly 0.1 dB on average (in comparison to pixel-by-pixel scanning) and still does outperform the other techniques for all the tested images. Moreover, note that if we continue increasing the scanning step, the computational load can be reduced even further with relatively moderate effect on the reconstruction quality (see N_8 in Table 1).

5 Conclusions

Block-based coding standards, such as H.264 and MPEG-4, are widely spread in video transmission over packet based networks. However, the packetized stream suffers from packet losses, and so some of the macroblocks are not received or decoded properly. Moreover, as the aforementioned standards use interframe prediction, a single packet loss could cause an error propagation, distorting the whole video sequence. In this paper, we have developed an error concealment scheme that utilizes only the spatial information of the current frame, making it specially suitable for reconstruction of I-blocks. The concept of visual clearness of an edge is introduced in order to find a suitable set of weights that are used in a pixel-level weighted combination of interpolations that allows one to reconstruct even nonlinear features more accurately. The proposed algorithm shows a significant improvement while keeping a relatively moderate computational complexity.

Regarding future work, the reconstruction of more complex visual features by combining edge and texture reconstruction is a goal worth exploring.

Acknowledgements This work has been supported by the Spanish MEC/FEDER project TEC2010-18009.

References

1. J. Canny, A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell. **8**, 679–698 (1986)
2. H. Gharavi, S. Gao, Spatial interpolation algorithm for error concealment, in *Proceedings of ICASSP*, (2008), pp. 1153–1156
3. P.F. Harrison, Texture synthesis, texture transfer and plausible restoration. Ph.D. Thesis, Monash University (2005)
4. ITU-T, ITU-T Recommendation H.264, International Telecommunication Union (2005)
5. W.Y. Kung, C.S. Kim, C.C.J. Kuo, Spatial and temporal error concealment techniques for video transmission over noisy channels. IEEE Trans. Circuits Syst. Video Technol. **16**, 789–802 (2006)
6. M. Ma, O.C. Au, S.H. Gary Chan, M.T. Sun, Edge-Directed error concealment. IEEE Trans. Circuits Syst. Video Technol. **20**, 382–394 (2010)
7. J. Østergaard, M.S. Derpich, S.S. Channappayya, The high-resolution rate-distortion function under the structural similarity index. EURASIP J. Adv. Signal Process. (2011)
8. A.M. Peinado, A.M. Gómez, V. Sánchez, Error concealment based on MMSE estimation for multimedia wireless and IP applications, in *Proceedings of PIMRC*, (2008), pp. 1–5, (invited paper)
9. D.L. Robie, R.M. Mersereau, The use of hough transforms in spatial error concealment, in *Proceedings of ICASSP*, vol. 4, (2000), pp. 2131–2134
10. Z. Rongfu, Z. Yuanhua, H. Xiaodong, Content-adaptive spatial error concealment for video communication. IEEE Trans. Consum. Electron. **50**, 335–341 (2004)

11. P. Salama, N.B. Shroff, E.J. Coyle, E.J. Delp, Error concealment techniques for encoded video streams, in *Proceedings of ICIP*, (1995), pp. 9–12
12. S. Shirani, F. Kossentini, R. Ward, An adaptive Markov random field based error concealment method for video communication in error prone environment, in *Proceedings of ICIP*, vol. 6, (1999), pp. 3117–3120
13. H. Sun, W. Kwok, Concealment of damaged block transform coded images using projections onto convex sets. IEEE Trans. Image Process. **4** (1995)
14. V. Varsa, M.M. Hannuksela, Non-normative error concealment algorithms, in *ITU-T SG16, VCEG-N62*, vol. 50 (2001)
15. Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assesment: from error visibility to structural visibility. IEEE Trans. Image Process. **13**, 600–612 (2004)
16. Z. Wang, E.P. Simoncelli, A.C. Bovik, Multi-scale structural similarity for image quality assessment. IEEE Signal. Syst. Comput. **2**, 1398–1402 (2003)
17. T. Wiegand, G.J. Sullivan, G. Bjontegaard, A. Luthra, Overview of the H.264/AVC video coding standard. IEEE Trans. Circuits 560–576 (2003)
18. G. Zhai, J. Cai, W. Lin, X. Yang, W. Zhang, Image error-concealment via block-based bilateral filtering, in *IEEE International Conference on Multimedia and Expo*, (2008), pp. 621–624
19. Y. Zhao, H. Chen, X. Chi, J.S. Jin, Spatial error concealment using directional extrapolation, in *Proceedings of DICTA*, (2005), pp. 278–283
20. http://dtstc.ugr.es/~jkoloda/download.html. Available online