Multimed Tools Appl (2017) 76:13835–13858
DOI 10.1007/s11042-016-3766-7

Spatial error concealment for intra-coded depth maps in multiview video-plus-depth

Pedro A. Amado Assunção[1,4] · Sylvain Marcelino[1,2] · Salviano Soares[2,3] · Sérgio M. M. de Faria[1,4]

Received: 17 November 2015 / Revised: 3 July 2016 / Accepted: 6 July 2016 /
Published online: 25 July 2016

Abstract Transmission errors or packet loss in depth maps have great impact on the decoding quality and view synthesis of 3D and multiview video. Thus efficient methods to recover corrupted depth data are critical functions for accurate view rendering. This paper proposes an error concealment method for intra-coded depth maps, based on spatial intra and inter-view methods, which exploit neighbouring data of depth and colour images received error-free. A novel three-stage processing algorithm is devised to reconstruct sharp depth transitions (i.e. lost depth contours), using a disparity map and geometric interpolation based on parametric Bézier curves. The simulation results obtained from different views of various MVD sequences, for different packetisation modes and a wide range of packet loss rates (PLR), show that the proposed method consistently leads to quality improvement of synthesised images in comparison with reference methods.

This work was supported by the Fundação para a Ciência e Tecnologia, Portugal under PhD Grant SFRH/BD/64988/2009 and Project PTDC/EEA-TEL 114487/2009.

✉ Pedro Amado Assunção
amado@co.it.pt

Sylvain Marcelino
stam@co.it.pt

Salviano Soares
salblues@utad.pt

Sérgio Maciel de Faria
sergio.faria@co.it.pt

[1] Instituto de Telecomunicações, Leiria, Portugal

[2] Universidade de Trás-os-Montes e Alto Douro/ECT Engineering Department, Vila Real, Portugal

[3] IEETA, UA Campus, Aveiro, Portugal

[4] Instituto Politécnico de Leiria/ESTG, Leiria, Portugal

Keywords Error concealment · Depth map · Multiview video-plus-depth · Spatial interpolation

1 Introduction

In the last few years, services and applications using three-dimensional representation of visual information, such as 3D and multi-view images/video, have been driving R&D efforts worldwide. As a result, the use of 3D media content and associated technology is expected to expand into many different fields of application, reaching an increasing number of consumers. Besides entertainment, there are also several other fields where 3D images and video are becoming quite useful, such as medical, engineering, training and education. As a consequence of this recent evolution, compressed 3D and multiview visual content is expected to be a major contributor to the overall multimedia traffic delivered through various types of channels. Nowadays, there are two main types of coded representations of 3D and multiview image and video content, where the most common are the multiview video coding (MVC) and multiview video-plus-depth (MVD) [3, 40]. In MVC, stereoscopic viewing is directly obtained from two adjacent views, while in the MVD format each colour image is associated with a corresponding depth map, which is used to synthesise other views.

The impact of transmission errors or data loss is greater in 3D and multiview communications than in traditional 2D communications, because the perceived quality is highly sensitive to a wider variety of factors [9] and error propagation [21]. Therefore, careful design of error concealment algorithms for robust multiview image/video decoding is essential to minimise the strong effects of data loss in the perceived quality [19, 23]. In particular, the quality and integrity of intra-coded depth maps plays a crucial role in the quality of the synthesised views with great impact on the perceived quality [29]. Hence, the efficiency of error concealment algorithms specifically tailored for intra-coded depth maps is a critical performance factor in multiview video delivery services.

Since depth transitions in 3D visual scenes correspond to edges in depth maps, the boundaries between visual objects located at different distances from the viewer, i.e. depth levels, comprise relevant information for three-dimensional perception. Therefore, preserving sharp edges is an important requirement when processing depth maps, regardless its objective, either to cope with transmission errors, data loss or minimise coding distortion. However, it is worthwhile to note that a depth map with lower distortion does not necessarily lead to better synthesised images. For instance, in [30] several depth maps were compressed with different encoders and it was concluded that coding depth maps with higher PSNR does not guarantee that the corresponding synthesised views have better quality than those obtained using lower quality depth maps. Despite the significantly higher coding efficiency of H.264/AVC, the lower PSNR depth maps compressed with the Platelet-based encoder, produced synthesised images of higher quality. This is due to the capability of Platelet-based coding to better preserve the quality of edges in comparison with H.264/AVC. In [8] and [11], the authors use the edges of depth maps as crucial elements in different problems, though both of them related to the quality of 3D video. In the former, a method for enhancing synthesised views is proposed by using edge-adaptive filtering of depth maps. In the latter, the authors demonstrate that edge information provides an accurate measure of quality (or distortion) in 3D video. Explicit edge information was also used in [34] to improve the quality obtained in image restoration algorithms. Therefore, whenever any type of distortion affects the depth information, preserving the whole geometry of the depth maps through

high quality edges is of major importance to achieve enhanced 3D perceptual quality, since this type of details carry in fact relevant information for quality depth perception.

The results of the works cited above, established the grounds for the novel error concealment method proposed in this paper to reconstruct lost depth data, through specifically tailored spatial intra and inter-view techniques. The contributions of this research comprise (i) a novel error concealment algorithm based on a hierarchical processing structure with three stages, each one corresponding to the use of different type of data, available at MVD decoders and associated with the corrupted depth map, i.e., adjacent colour views and depth maps, associated colour view, adjacent regions of the corrupted depth map itself; (ii) the reconstruction of depth discontinuities (i.e., contours) using Bézier curves to improve the quality of recovered depth data by spatial interpolation, i.e., recovering depth contours in lost regions and then using them as boundaries to define different weights in bidirectional interpolation. Even in difficult cases, where large areas of depth maps are lost (e.g. large slices), the results show that synthesised views using recovered depth maps, consistently achieve better quality than using methods that do not take depth boundaries into consideration.

This paper is organised as follows: Section 2 presents an overview of related work available in the literature. Section 3 describes the proposed error concealment algorithm and Section 4 presents the simulation results. Finally Section 5 concludes the paper.

2 Related work

In the recent past, research on error concealment for 3D video decoders has been attracting attention from both academic and related industry, in order to find efficient solutions for coping with error prone transmission channels. The existence of different 3D coding and representation formats requires different approaches and methods to deal with data loss. Currently, the two main formats of wider interest are the MVC and MVD, but error concealment methods published in the literature for these formats, are mostly based on existing ones that were already conceived for 2D video. Nevertheless, even these are still attracting research interest and evolving solutions keep being published in the relevant literature, e.g., [5].

In the case of MVC, the error concealment methods developed so far are mostly targeted to H.264/AVC stereo decoders. For instance, in [31] and [32] the authors used techniques adapted from H.264/AVC, while in [25] and [42], the proposed methods mostly rely on temporal information. In [6], besides temporal information, disparity is also used to recover the lost data, which in general contributes to enhance the results in MVC.

Regarding MVD, most of the work developed so far, dealing with depth map error concealment, is focused on efficient use of temporal information extracted from coded sequences with depth maps. The most common data loss scenarios are the loss of a full frame [38], though in some cases only single blocks or groups of blocks need to be recovered (i.e., slices), as proposed in [43], where motion information is used based on the H.264/AVC standard. However, since the authors have only used one video sequence in the experimental validation, it is difficult to evaluate the effectiveness of the method.

In [37] and [26], the authors propose methods for error concealment in MVD but these are generic in the sense that the specific characteristics of depth maps are not taken into account. In [37] a general approach for MVD is proposed, based on classic temporal techniques with motion vectors, intra information and a novel inter-view technique, while in [26],

a frame concealment technique was developed using motion vector sharing. The case of entire loss of depth maps is addressed in [7, 12, 24, 44] and [23], using concealment methods for MVD mostly relying on temporal information. In [12], motion correlation between colour and depth is exploited for concealment and in [7], the motion information of colour images is also used to conceal depth maps, based on a 3D-warping technique. In [44], the proposed method also deals with full frame concealment using a pixel-based motion vector extrapolation technique. In [10], the authors propose an efficient concealment method for both colour and depth, also based on temporal information. In [24] a technique based on a 3D-warping is used, while the remaining holes are concealed with temporal method. In [23], colour and depth map information is used together in order to conceal errors on both of these types of data, this technique is based on motion compensation.

In regard to spatial concealment methods specifically tailored for intra-codee depth maps, to the authors' knowledge, these have not been extensively studied in the available literature. In [1], a spatial concealment method is proposed using depth information to restore the corresponding colour image, based on simple thresholding of the depth map. However, since the method depends on the 3D view content itself, the scope of its application is limited. A different approach is described in [13], where the correlation of motion information between colour images and their corresponding depth maps is used to recover lost regions in both the colour images and depth maps. However, despite the good objective results, depth contours are not preserved, causing noticeable blocking artifacts in the depth map, mainly at edges between foreground and background objects in the 3D scene. In [27] the authors assume that in presence of depth map errors, the corresponding colour image region is free of errors, then it can be used for the depth map recovery. Despite the good performance of such method in this specific case, only the quality of the depth map itself was evaluated and the quality of the corresponding synthesised views remains unknown.

In most of the above cited publications, the impact of depth map error concealment in the quality of the synthesised views is not always evaluated, which limits its performance validation and comparison. In the next sections a new method for efficient spatial intra and inter-view depth concealment for multiview-plus-depth images using depth contour reconstruction and interpolation is proposed.

3 Proposed method

The spatial error concealment method proposed in this work is structured as a hierarchical three-stage processing algorithm, where each stage is defined by the type of data used in spatial concealment of missing regions in depth maps. Figure 1 shows the three cases that may occur in MVD, when a corrupted depth map is received and no temporal information is used: (i) reliable data from both adjacent colour images is available; (ii) the only reliable data that is available comes from the colour image associated with the corrupted depth map; (iii) the colour images do not provide any useful information for concealment of the lost region in the corrupted depth map. Whenever the texture data from colour images is located in image regions where the disparity cannot be computed with good accuracy, this is classified as a non-reliable data region to be used by the concealment algorithm. This is possible to occur in three cases: 1) missing texture data due to losses; 2) occluded regions in the texture view; 3) due to algorithmic inefficiency, the texture disparity tested in the border regions of the lost area in the depth map does not produce a good enough match between both depth maps.

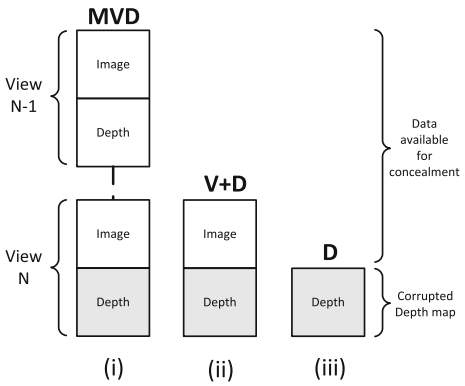


Fig. 1 Depth loss cases in spatial MVD, characterised by the data available for error concealment

Taking into account the three cases identified above, an algorithmic structure is proposed for spatial error concealment of depth maps, comprising the 3-stages shown in Fig. 2. *Stage 1* corresponds to case (i), where reliable data from the two views is used for error concealment of lost areas in the corrupted depth map. At this stage, the disparity between available views is used to retrieve missing depth values from the depth map of the adjacent view, i.e., disparity compensation between depth maps.

If not all lost areas of the depth map are fully recovered in *Stage 1*, because the auxiliary view/depth cannot provide useful disparity information, it means that only one view plus depth is available. Then this is processed in *Stage 2*, corresponding to case (ii) in Fig. 1. In *Stage 2* the contours of the lost regions in the corrupted depth map are first reconstructed by using information from its associated colour image. If all depth map contours of interest are not fully recovered after *Stage 2*, because they were not found in the colour view, for instance, then the algorithm proceeds to *Stage 3*, where only depth data of the corrupted depth map itself is used for concealment, i.e., case (iii) in Fig. 1. At this Stage, the remaining lost contour segments are reconstructed using Bézier curves. A similar method for contour reconstruction was used in [36], though in a different context. Finally, the lost regions of the depth map are recovered using selective weighted interpolation based on the reconstructed contours. In the next subsections each stage of the proposed algorithm is explained in detail.

3.1 Stage 1 - Depth recovery using disparity compensation

Two views (*N-1* and *N*) are used in this processing stage, each one comprising a colour image and the corresponding depth map (see Fig. 1). It is assumed that *Depth N* is corrupted, i.e., with missing regions. The disparity D_N, computed from colour images N-1 and N, is used to find the best matching depth values in *Depth N-1* to recover the lost ones in *Depth N*. When *Depth N* is fully lost, *Stage 1* is the only processing stage of the algorithm that is used to recover the whole depth map. In this case, *Depth N-1* is used as a substitute of *Depth N* after disparity compensation. Occlusions are filled by gradient-based linear interpolation. The following steps are carried out at *Stage 1*.

3.1.1 Disparity computation

The first step is to compute a disparity map D_N between *Image N-1* and *Image N*. The disparity is determined using an algorithm based on a multi-level adaptive method combined

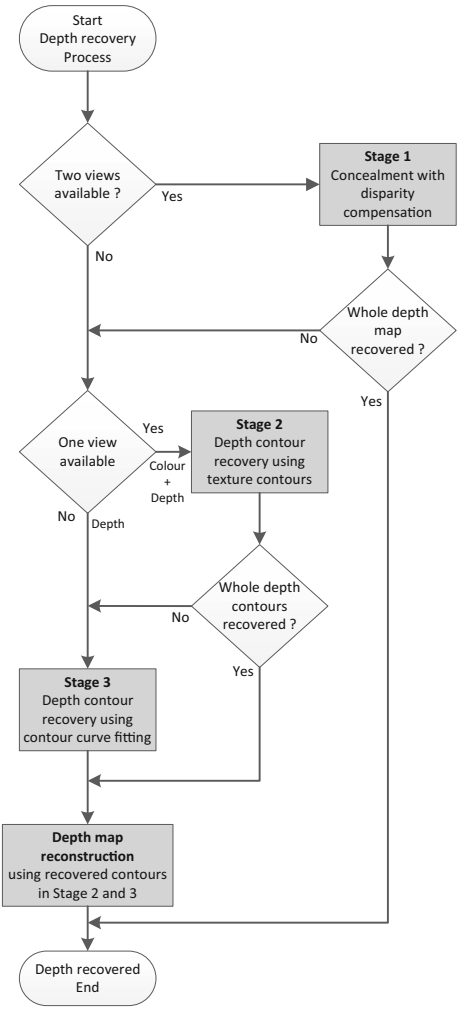


Fig. 2 Algorithmic structure of the error concealment method

with a multigrid technique [22]. It presents low implementation complexity and good efficiency in comparison with others, as demonstrated in the study carried out with the *Middlebury Stereo Evaluation* dataset [16].

3.1.2 Disparity compensation

The next step in *Stage 1* is to recover lost depth values using disparity compensation. Assuming standard block-based coding, each lost region corresponds to either a single block or a group of blocks. Each missing region in the depth map is reconstructed by using the disparity compensated values obtained from the corresponding region in the error-free map of the adjacent view.

Since disparity is computed from the colour images, *view N-1* and *view N*, the resulting disparity map is not ground truth data and does not ensure the most accurate values for filling the corresponding lost regions in the corrupted depth map. Therefore, to ensure the

best match between depth maps, the disparity values are further refined before being used to retrieve the depth values from view *N-1* to *N*. The refinement process devised to compute suitable disparity values to recover the missing block is based on the following method. The disparity values corresponding to the lost region are quantised using a fixed number of intervals ranked in increasing order. In this work 15 quantisation intervals were used, this number of intervals was found through several experiments to be enough to define the initial search points that are used to find the best match. Then each quantised value is used to define an initial search point for finding the best matching block in *phDepth N-1* that should be used to recover the corresponding corrupted one in *Depth N*. The refinement is applied over a range of 4 pixel, using the minimum Sum of Absolute Differences (SAD) as the best matching criterion. In this process, the searching area used for finding the best match only includes the depth values belonging to the boundaries of the lost region, because the region itself is missing.

Figure 3 illustrates this process using a single lost block in a corrupted depth map. The neighbour depth values of the Candidate Block from depth map *N-1* are used to find the best match for the Lost Block in *Depth N*, i.e., the top and bottom neighbour values of the lost block, $Dt_{i,j}$ and $Db_{i,j}$ respectively, are used for searching the minimum SAD with $Ct_{i,j}$ and $Cb_{i,j}$. In Fig. 3, h and l are the height and width of both the lost and candidate blocks and W is the width of the search window only in horizontal height. The best match is found by (1), using three rows on the top and another three on the bottom (i.e., $r = 3$) of the lost block.

$$SAD_{min} = \min_{n \in W} \sum_{\substack{1 \leqslant i \leqslant l \\ 1 \leqslant j \leqslant r}} |Dt_{i,j} - Ct_{i,j}|_n + |Db_{i,j} - Cb_{i,j}|_n \tag{1}$$

If SAD_{min} is greater than a decision threshold related to the number of pixels n_{pel} used in the computation, i.e., $k \times SAD > n_{pel}$ then the lost depth block is not recovered with disparity compensation, but is further processed in the following stages. The constant $k = 20$ was found through an experimental study using six sequences, as used in Section 4 using several error patterns. The amount of processing and concealment done at *Stage 1* depends

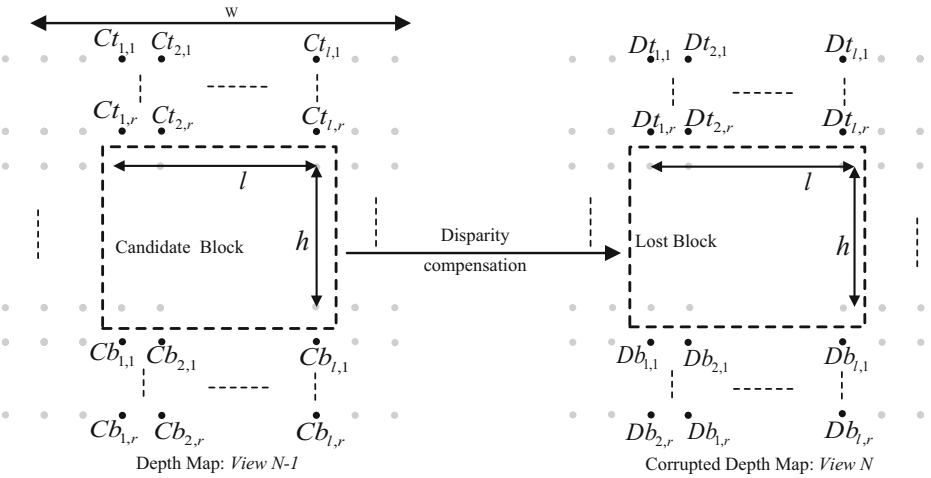


Fig. 3 Depth map block matching using disparity

on k, but its actual value was found not to be a critical factor in the overall concealment performance.

3.1.3 Depth correction

Although depth maps from adjacent views are highly correlated, there might be slight differences between their average depth values for the same region of the 3D scene. This is equivalent to have different average intensities between depth maps *N* and *N-1*. Thus, intensity correction is performed after recovering a lost depth block, which requires edge information to distinguish regions with different intensity. In the proposed method, the Canny edge detection algorithm is used for such purpose. Figure 4 shows an example of the underlying process devised for intensity correction using depth map edges. In this example a depth Candidate Block from *view N-1* was used to recover a Lost Block in *view N*. The intensity correction is achieved by adding a correcting factor $\Delta_{(i,j)}$ to each depth value $P_{(i,j)}$, as given by (2),

$$P_{(i,j)} = P_{(i,j)} + \Delta_{(i,j)} \tag{2}$$

with $\Delta_{(i,j)}$ defined by (3):

$$\Delta_{(i,j)} = \frac{(D_{(i,0)} - C_{(i,0)})d2 + (Db_{(i,h+1)} - C_{(i,h+1)})d1}{d1 + d2} \tag{3}$$

where i and j are the coordinates of depth value $P_{(i,j)}$ to be corrected, h is the height of the block, $d1$ and $d2$ are the distances between $P_{(i,j)}$ and the top or bottom values in the concealed depth map. $D_{(i,0)}$ and $Db_{(i,h+1)}$, $C_{(i,0)}$ and $C_{(i,h+1)}$ are the top/bottom neighbour values in the depth map of *view N* and *view N-1*, respectively.

In the example illustrated in Fig. 4, the interpolated value $P_{(i,j)}$ is located below the contour. In this case, the upper neighbour value $D_{(i,0)}$ is not used because it belongs to a region with different intensity, as defined by the contour. Thus, the difference between $D_{(i,0)}$ and $C_{(i,0)}$ is not considered and $d2 = 0$.

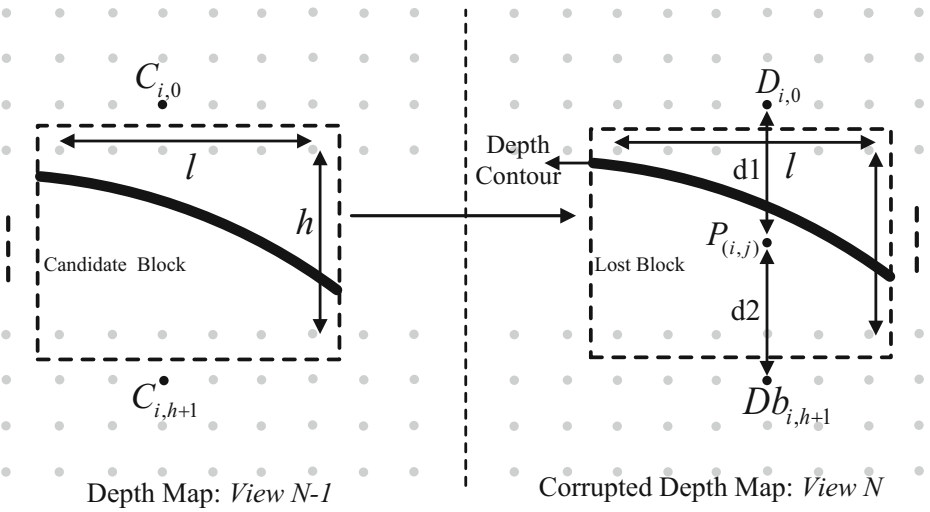


Fig. 4 Intensity compensation of the recovered depth map

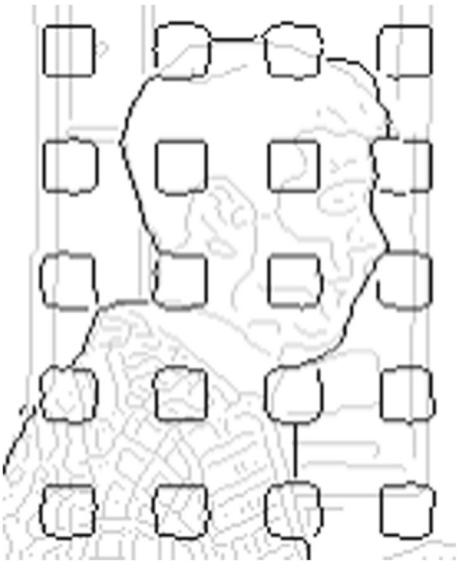


Fig. 5 Object contours: colour (*light gray*) and depth (*dark gray*)

3.2 Stage 2: Depth countour reconstruction

In *Stage 2*, reconstruction of the contours inside the missing regions of the depth maps is based on contour information extracted from the corresponding colour image. In general, object contours in depth maps also exist in the corresponding colour images. However, colour image contours include texture information which does not necessarily include object contours. Moreover, object contours in colour images are not always spatially coincident with their corresponding ones in depth maps. This mismatch is shown in Fig. 5 where colour and depth contours are represented along with a uniform pattern of possible missing blocks.

As shown in Fig. 5, there are nearly-matching colour contours that can be used to reconstruct broken depth contours, while others do not have any correspondence. In order to find the colour image contours that are useful for reconstruction of missing contour segments inside lost regions of a depth map, the method illustrated in Fig. 6 is proposed.

Is this example there is a broken contour (C_{D1} and C_{D2}) across a lost region in the depth map and another contour (C_{C1} and C_{C2}) taken from the corresponding colour image. The angles $\alpha 1$, $\alpha 2$, $\beta 1$ and $\beta 2$ are defined as the angles of the tangent vectors pointing to the

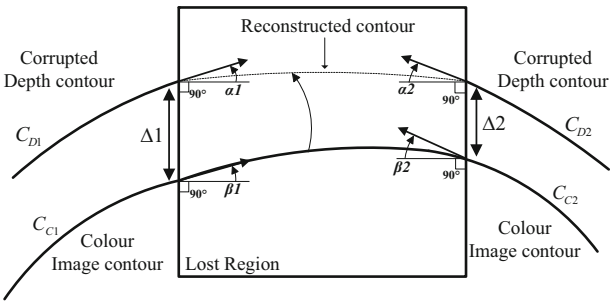


Fig. 6 Matching colour contours

lost area. In this case, $\alpha 1$ and $\alpha 2$ represent the tangent angles corresponding to C_{D1} and C_{D2}, $\beta 1$ and $\beta 2$ represent tangent angles corresponding to C_{C1} and C_{C2}. $\Delta 1$ and $\Delta 2$ are the distance between the end points of C_{C1} and C_{D1}, C_{C2} and C_{D2}, respectively. In order to obtain a precise computation of $\alpha 1$, $\alpha 2$, $\beta 1$ and $\beta 2$, up to 5 contour depth points, located behind each end-point, are used to define the tangent vectors. Thus, in order to recover broken depth contours from colour image contours, the following procedure is defined:

1- Match: Every end-point of each broken depth contour segment is checked for a possible match in the colour image. This is validated by checking whether there is a colour image contour close to the end point (e.g., up to a distance of 10 points). In Fig. 6, the candidate for C_{D1} is C_{C1}.

2 - Find: The tangent angles of the broken depth contour and colour contour candidates are compared in order to find out if the tangent angles define a similar direction. The direction is considered similar when the difference is smaller than 30 degrees, which was found a suitable value for this type of similarity.

3 - Fit: Then, any colour contour inside the lost area is tested whether it fits onto the broken section of the depth contour. This is done by verifying the continuity of the colour contour against the possible candidate. In Fig. 6, C_{C1} corresponds to the same contour as C_{C2}, thus C_{C2} is the candidate to reconstruct the broken depth contour C_{D2}.

4 - Recover: If the colour contour segment meets the above conditions, then it is used to recover the depth contour. The colour contour segment is copied to connect the two end points of the broken depth contour. In the case of Fig. 6, the colour contour C_{C1} to C_{C2} is copied to the end-points of C_{D1} and C_{D2} by moving it from its original location with a distance of $(\Delta 1 + \Delta 2)/2$. If the copied contour end-points do not match exactly the end points of the damaged contour these are interpolated with a straight line. Usually this is no more than two or three contour points.

In the case where no suitable colour contours are found to recover the damage depth contours, these loose ends will be interpolated at *Stage 3*, described in the next section.

3.3 Stage 3: Depth contour recovery using curve fitting

This stage implements a refinement process after *Stage 2*, where the pairs of remaining end-points are further processed to reconstruct broken depth contours not recovered during *Stage 2*. The proposed method for this refinement is based on matching end-points using geometrical similarities and Bézier curves [28].

3.3.1 Matching end-points

Since several contour segments might remain damaged after *Stage 2*, in order to reconstruct them separately, the corresponding end-points are first matched together in pairs. Such pair of end-points correspond to the points, where the contour is broken by the lost region.

Figure 7 shows an example of two contours passing through the lost region. C_{D1}, C_{D2}, C_{D3} and C_{D4} are segments of the corrupted depth contours that pass through the lost region and the respective end-points are $E1$, $E2$, $E3$ and $E4$. The pair of end-points that are more likely to belong to the same contour is found by testing every possible combination of matching contours. In the example of Fig. 7, C_{D1} is being tested with C_{D2}, C_{D3} and C_{D4}.

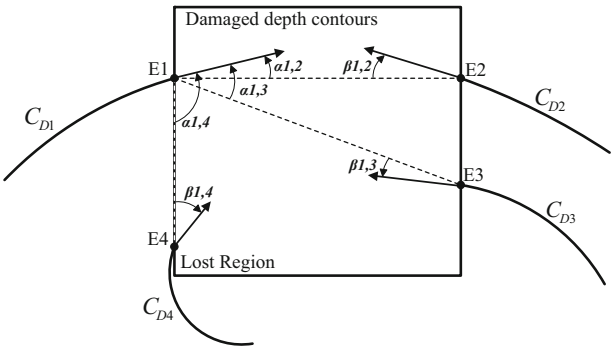


Fig. 7 Matching corrupted depth contours

The angle $\alpha_{i,j}$ is defined by the tangent vector of C_{Di} and the straight line between the end-points $\overline{E_i, E_j}$. Angles $\beta_{i,j}$ are defined by the tangent vector of C_{Dj} and the straight line between the end-points $\overline{E_i, E_j}$. The index numbers i, j of the angles $\alpha_{i,j}$ and $\beta_{i,j}$ are defined as: i is the number of the contour being tested, in the example of Fig. 7 $i = 1$; j is the number of the contour that is being compared with i, in the example of Fig. 7, $j = 2, 3, 4$;

After computing all the previous angles $\beta_{i,j}$ and $\alpha_{i,j}$, the best pair of end-points (E_i, E_j) is obtained by the following (4):

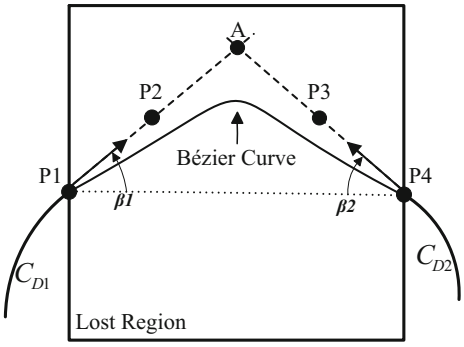
$$(E_i, E_j) = \min_{\forall i,j}\{(\alpha_{i,j} + \beta_{i,j}); (\alpha_{i,j+1} + \beta_{i,j+1}); (\alpha_{i,j} + \beta_{i,j+n})\} \tag{4}$$

The best match (E_i and E_j) is given by the minimum of the sums of the pair of angles α and β. If the resulting end-points combinations give rise to intersecting contours, then such matching pair is not considered valid and another possible match is checked.

3.3.2 Contour interpolation with Bézier curves

After finding the best matching endpoints, cubic Bézier curves are used to connect the lost contour segments. These are based on four control-points $P1 \ldots P4$, where the first ($P1$) and the last ($P4$) are the end-points of the broken contour segment. The control points $P2$ and $P3$ are computed based on the tangent vectors at the end-points $P1$ and $P4$ (see Fig. 8). Control points $P2$ and $P3$ are found by drawing two virtual lines passing at P1 and P4, with angles β_1 and β_2, respectively. Then $P2$ and $P3$ are the points located at the middle

Fig. 8 Bézier curve fitting



of segments $\overline{P1-A}$ and $\overline{P4-A}$, where A is the intersection point of the two virtual lines (see Fig. 8). Finally, these four control points are used to define the Bézier curve, which in turn is used to reconstruct the lost segment of the contour.

A cubic Bézier curve is represented by $Q(t)$ (5), where $t \in [0, 1]$. P_n are the control-points described above ($n \in [1, 4]$) and Pn_x and Pn_y are the coordinates of Pn.

$$Q(t) = (1-t)^3 P1 + 3t(1-t)^2 P2 + 3t^2(1-t)P3 + t^3 P4 \tag{5}$$

$Q(t)$ can also be written in a matrix form as follows:

$$Q(t) = [x(t) \;\; y(t)]^T = CF \tag{6}$$

The parametric curves $x(t)$ and $y(t)$ representing $Q(t)$ are:

$$x(t) = a_x t^3 + b_x t^2 + c_x t + d_x \tag{7}$$
$$y(t) = a_y t^3 + b_y t^2 + c_y t + d_y \tag{8}$$

Defining C as the coefficient matrix of $x(t)$ and $y(t)$, and F as the parameter vector of the parametric curves, i.e.,

$$C = \begin{bmatrix} a_x & b_x & c_x & d_x \\ a_y & b_y & c_y & d_y \end{bmatrix} \tag{9}$$
$$F = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix}^T \tag{10}$$

Since (6) does not depend on any control point, in order to find the Bézier curve that represents the lost segment of the depth contour, one has to define an equation where P1...P4 are included. This can be obtained by using the definition of the Bézier curve $Q(t)$ based on the basis matrix B [17], i.e.,

$$Q(t) = PBF \tag{11}$$

with

$$B = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \tag{12}$$

where the elements of P are the coordinates Pi_x, Pi_y of the four control points Pi, $i = 1 \cdots 4$, i.e.,

$$P = \begin{bmatrix} P1_x & P2_x & P3_x & P4_x \\ P1_y & P2_y & P3_y & P4_y \end{bmatrix} \tag{13}$$

After substituting $Q(t)$ in (11) by CF from (6), the problem solution consists of finding the elements of matrix C. The result is the following set of equations:

$$a_x = -P1_x + 3 \times P2_x - 3 \times P3_x + P4_x \tag{14}$$
$$a_y = -P1_y + 3 \times P2_y - 3 \times P3_y + P4_y \tag{15}$$
$$b_x = 3 \times P1_x - 6 \times P2_x + 3 \times P3_x \tag{16}$$
$$b_y = 3 \times P1_y - 6 \times P2_y + 3 \times P3_y \tag{17}$$
$$c_x = -3 \times P1_x + 3 \times P2_x \tag{18}$$
$$c_y = -3 \times P1_y + 3 \times P2_y \tag{19}$$
$$d_x = P1_x \tag{20}$$
$$d_y = P1_y \tag{21}$$

After computing matrix C, the parametric curves $x(t)$ and $y(t)$ are completely defined. Then, the interval of t ([0,1]) is divided by twice the number of depth values comprising the segment $\overline{P1-P4}$, in order to obtain the number of points of the Bézier curve. Finally, based on the computed curve, the closest points inside the lost region are chosen to build to the reconstructed contour. When all contours are recovered, the depth values of the remaining lost blocks are interpolated using the method described in the next section.

3.4 Depth map reconstruction

After recovering all lost contour segments across the lost region, in *Stage 2* and *Stage 3*, the missing depth values must be estimated to fill the whole lost region.

Figure 9 shows the process for determining the lost depth value $Y_{(i,j)}$ by using weighted interpolation and the reconstructed contour. $Y_{(i,j)}$ is interpolated based on four depth values ($y_k, k = 1...N$ and $N = 4$) located in the boundaries of the Lost Block. However, depending on the existence of a recovered contour in the lost block, not all of the adjacent depth values are necessarily used. If a contour segment crosses a block, then the depth values located on the opposite side of the contour are not used for interpolation, because they belong to a region with very different depth level. For this reason, in the example of Fig. 9, y_4 is not used for interpolating $Y_{(i,j)}$. The weighted interpolation is implemented as defined by (22):

$$Y_{(i,j)} = \frac{\sum_{k=1}^{N} y_k \times [B - d(Y_{(i,j)}, y_k) - 1]}{\sum_{k=1}^{N} d(Y_{(i,j)}, y_k)}, \tag{22}$$

where $d(Y_{(i,j)}, y_k)$ is the distance between the coordinates of $Y_{(i,j)}$ and y_k and B is the block size.

3.5 Complexity discussion

Overall, it can be shown that the relative complexity of a robust decoder with the concealment algorithm described above lies between that of a simple decoder without error concealment mechanisms and an encoder. It is known that the complexity of an H.264/AVC encoder E_c is roughly about 10 to 90 times higher than the corresponding decoder, D_c, i.e.,

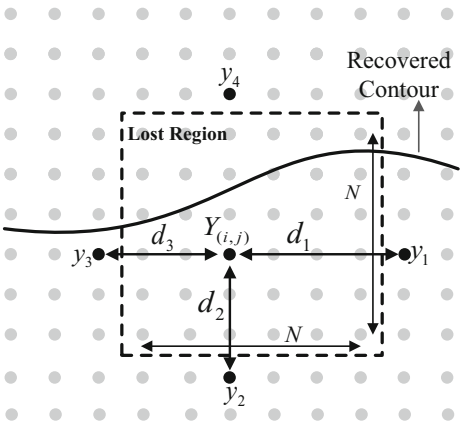


Fig. 9 Weighted interpolation using a reconstructed contour

$10 \times D_c < E_c < 90 \times D_c$ [33, 35]. In the proposed method, disparity estimation (DE) is the most complex task and for this application a block-based algorithm, similar to motion estimation, can be considered as a reference to compare the computational complexity. Thus based on [20], it is assumed that disparity estimation complexity (DE_c) is roughly the same as motion estimation complexity (ME_c), i.e., $DE_c = ME_c$. Since ME_c represents about 70 % of the overall encoding complexity [4], this means that the additional complexity of a robust decoder lies between $7 \times D_c$ and $63 \times D_c$. Note that this is valid for the very worst case where DE is computed for the whole frame. From another point of view, it is also worth to note that the increased complexity of the robust decoder is still below the maximum complexity of an encoder, i.e., $90 \times D_c$.

4 Simulation results and discussion

The spatial concealment algorithm presented in previous sections was evaluated using the MVD format with corrupted depth maps. The quality of the synthesised views (PSNR) was used as the performance metric of different spatial concealment methods applied to depth maps used in the synthesis. The objective quality is evaluated using the virtual view synthesised with the uncorrupted depth map instead of the actual view, to avoid the influence of the DIBR algorithm in the results.

The synthesis of virtual views was performed by using the reference software VSRS 3.5 (View Synthesis Reference Software) [39]. To obtain the necessary disparity map for *Stage 1*, the *OpenCV 2.1* software package was used [15, 22]. The contours of both the colour images and depth maps are extracted using the *Canny* edge detector [2]. The code was written in C and MATLAB 2012 and the simulations were run on Intel I7 2600, 3,4GHz 8GB RAM.

To evaluate and compare the performance of the proposed algorithm, two reference methods were used for spatial concealment of the same corrupted depth maps: (i) method R1 based on weighted spatial interpolation to fill in the missing depth regions using the uncorrupted depth values located along the borders of each missing region; (ii) method R2, based on disparity compensation using the depth values of the uncorrupted adjacent view. The spatial interpolation procedure is defined by (22), where $P(x, y)$ is the value being interpolated at coordinates x and y. No contour information is used in reference methods R1 and R2. In the presence of error bursts affecting several consecutive blocks, weighted interpolation only uses the nearest (e.g., top and bottom) error-free border values ($N = 2$).

Two colour views and the corresponding depth maps of six sequences with 100 frames each, were used in the simulations: *Book Arrival* (1024x768), *Balloons*, *Dancer* (1920x1080), *Shark* (1920x1080), *Kendo* (1024x768) and *Champagne* (1280x960). The visual content of each sequence has different characteristics, spatial resolution and different types objects in the 3D scene, which results in a diverse dataset to evaluate the proposed algorithm. In sequence *Book Arrival view 8* and *view 10* were used to synthesise *view 9*, in *Balloons* and *Dancer*, *view 1* and *view 3* were used to synthesise *view 2*, in *Shark* and *Kendo view 1* and *view 5* were used to synthesise *view 3* and in *Champagne view 37* and *view 39* were used to synthesise *view 38*.

To evaluate the performance of the proposed spatial concealment method, the colour images and the corresponding depth maps used in the simulations were encoded with H.264/AVC Intra mode at fixed QP=30, using the reference software *JM17* [14]. Eight slice groups per depth map were packetized into one single packet, which never exceeded 1500 bytes in these tests. Two different types of slices were defined in slice mode and another

(a) Pack. mode 1 (b) Pack. mode 2 (c) Pack. mode 3

Fig. 10 Error patterns originated from different packetisation modes

one using the Flexible Macroblock Order (FMO) [41]. In the former, each slice comprises either one or two rows of macroblocks, while in the latter the dispersed mode was used. Note that FMO is not defined in the standard extensions based on the high profile thus, this case finds application in simulcast encoding schemes using the baseline profile. These packetisation modes result in three different types of error patterns, as shown in Fig. 10: (a) single macroblocks of 16x16 (b) rows of height=16; (c) rows of height=32. Packet losses were simulated using the Gilbert-Elliot model for different average packet loss ratios (PLR) and burst sizes [18]. Average PLRs of 5 %, 10 %, 20 % and 40 % with an average burst size of 3 were obtained from each sequence. To obtain stable results, each simulation run was repeated 10 times.

The proposed algorithm was evaluated in two different modes, referred to as PM(a) and PM(b) in Table 1. Mode PM(a) corresponds to the full algorithm operating as depicted in Fig. 2, while mode PM(b) is intended to evaluate the case where no texture information of colour images is available for any reason (e.g., lost, recovered from loss with poor quality). In the case of PM(b), the algorithm in Fig. 2 follows from the start directly to Stage 3 and concealment is fully obtained by using uncorrupted data from the depth map itself and Bézier curves, as described in Section 3.3.2. Note that this is the most difficult concealment case, because no other data from adjacent views is used. The PSNR of the synthesised virtual views obtained by using depth maps concealed with the proposed method, operating in modes PM(a) and PM(b), is compared with that obtained from reference methods R1 and R2, for different PLR and the three types of packetisation modes described above. Table 1 shows the difference (Δ_{PSNR}) between the PSNR of synthesised views using the error-free depth map ($PLR = 0$ % in column 0 %) and the PSNR obtained from the various concealed depth maps ($PLR = x$ %). The lower this difference, the better the quality.

The case of full loss of the depth map was also simulated using 30 random and independent losses. The average PSNR of the virtual views synthesised with the fully reconstructed depth maps was compared with those synthesised with the corresponding error free depth maps. The results are shown in Table 2. In this case, the proposed method (PM(c) in Table 2) uses the disparity compensated depth map from the adjacent view to reconstruct the whole lost depth map. In the algorithm of Fig. 2, this corresponds to use *Stage 1*. The results are compared with those obtained from reference method R3, i.e., the least complex method.

4.1 Discussion

The experimental results shown in Table 1 demonstrate that the proposed method is able to consistently outperform the reference ones. The performance is better in sequences that have depth maps with many objects, because the lost areas are recovered with good accuracy,

Table 1 PSNR (dB) of synthesised views using the concealed depth maps under three different packetisation modes and four PLR

			Pack. mode 1				Pack. mode 2				Pack. mode 3			
			$\Delta_{PSNR} = PSNR_{0\,\%} - PSNR_{x\,\%}$											
Sequence	Method	0 %	5 %	10 %	20 %	40 %	5 %	10 %	20 %	40 %	5 %	10 %	20 %	40 %
Book Arrival	PM(a)	40.27	0.48	0.59	0.75	1.70	0.23	0.33	0.33	0.72	0.32	0.35	0.44	0.94
	PM(b)		0.44	0.54	0.68	1.73	0.63	0.81	1.14	2.81	0.68	0.80	1.09	2.36
	R1		0.65	0.92	1.30	3.67	0.68	0.82	1.05	2.59	0.99	1.23	1.55	3.40
	R2		2.31	2.56	3.16	5.04	0.60	0.81	0.89	1.53	0.63	0.73	0.88	1.57
Balloons	PM(a)	41.53	0.64	0.73	0.96	2.34	0.57	0.64	0.82	1.75	0.80	0.76	1.14	2.17
	PM(b)		0.61	0.69	0.95	2.47	0.56	0.63	0.85	1.95	0.95	0.97	1.50	3.04
	R1		0.87	1.21	1.73	5.13	0.76	0.90	1.18	2.95	1.24	1.37	1.96	4.27
	R2		1.40	1.47	1.87	3.11	1.77	2.03	2.42	3.90	1.96	1.91	2.47	3.97
Dancer	PM(a)	38.04	0.03	0.12	0.19	0.74	0.03	0.08	0.08	0.16	0.02	0.09	0.10	0.21
	PM(b)		0.16	0.32	0.38	0.92	0.24	0.37	0.42	1.19	0.30	0.56	0.65	1.80
	R1		0.34	0.65	0.95	3.28	0.11	0.21	0.24	0.85	0.17	0.35	0.38	1.26
	R2		0.48	0.69	0.83	1.37	0.57	0.71	0.86	1.49	0.55	0.73	0.86	1.49
Shark	PM(a)	39.77	0.33	0.38	0.37	0.78	0.26	0.29	0.20	0.49	0.25	0.28	0.21	0.21
	PM(b)		0.48	0.58	0.63	1.31	0.46	0.58	0.64	1.45	0.71	0.83	1.04	2.37
	R1		0.58	0.76	0.94	2.34	0.59	0.74	0.83	1.88	0.83	0.96	1.16	1.26
	R2		0.62	0.67	0.71	1.23	0.64	0.73	0.75	1.44	0.65	0.71	0.78	1.49
Kendo	PM(a)	41.79	1.00	1.15	1.64	3.25	1.10	1.15	1.51	2.40	1.02	1.25	1.63	2.74
	PM(b)		0.82	0.92	1.58	3.44	1.03	1.10	1.44	2.42	1.07	1.29	1.77	3.30
	R1		1.24	1.59	2.25	5.82	1.15	1.15	1.60	3.09	1.37	1.64	2.23	4.07
	R2		1.70	1.99	2.41	3.70	1.76	1.94	2.46	3.69	1.66	1.96	2.49	3.69
Champagne	PM(a)	38.95	0.80	1.28	1.58	3.30	0.70	0.92	1.32	2.72	0.73	0.86	1.20	2.40
	PM(b)		0.94	1.43	1.75	3.40	1.91	1.91	3.89	9.83	2.40	3.43	4.88	9.98
	R1		1.46	2.27	2.94	6.14	1.17	1.52	1.91	4.34	2.02	2.37	3.05	5.55
	R2		3.44	3.97	4.40	7.37	3.80	4.21	4.94	8.54	3.72	4.10	5.00	7.06

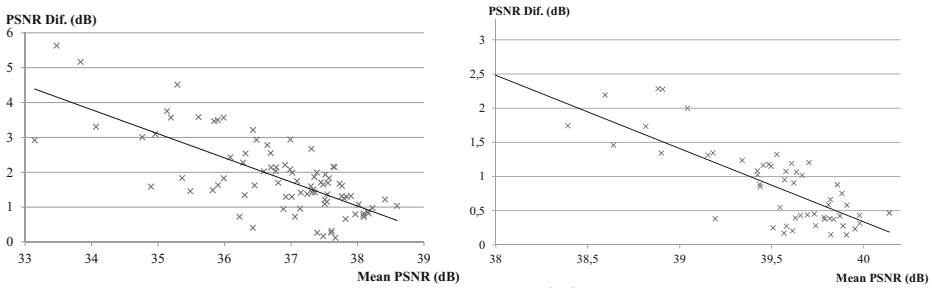
preserving the depth edges which are of major importance to achieve high quality synthesised images. The PSNR obtained by the proposed method is higher than the one achieved by the reference method R1, up to 4.04dB (*Champagne*), in the worst case scenario at 40 % error loss in a depth map, with lost blocks of 16x16 (*packetisation mode 1*). The higher quality gain in Champagne is due to its higher spatial detail (i.e., more objects at different depth levels) in comparison with Balloons, for instance. Since Balloons is smoother than Champagne, which is a friendly characteristic for the reference method R1, the difference between the proposed and reference methods is smaller, in the case of Baloons. In general, the quality gain obtained with the proposed method is higher when recovering highly detailed missing regions, which are in fact the most difficult ones to reconstruct. This is the reason why the Champagne sequence provides better results than other sequences.

The lowest gains over the reference method R1 occur for slice modes where packet loss leads to missing regions with small height, such as, for instance *Pack. Mode 1*. In this case, particularly in depth maps with highly homogeneous regions, the results obtained from the

Table 2 Full loss: PSNR difference ($\Delta PSNR$) between synthesised views using error-free and reconstructed depth maps

Sequence	Method	Δ_{PSNR}	Err. free
Book arrival	PM(c)	2.68	40.27
	R3	8.45	
Balloons	PM(c)	6.41	41.53
	R3	9.28	
Dancer	PM(c)	2.96	38.04
	R3	3.38	
Shark	PM(c)	2.85	39.77
	R3	3.47	
Kendo	PM(c)	6.17	41.79
	R3	9.38	
Champagne	PM(c)	8.39	38.95
	R3	11.8	

reference methods are close to the proposed one (.e.g., Dancer, Kendo). The advantage of using the proposed method increases at higher error rates with larger lost areas, as shown by the results in Table 1. This is due to the use of depth contours and selective spatial interpolation, which increase the reconstruction accuracy of depth values. The lowest gains over reference method R2 occur when the computed disparity is more accurate. In the synthetic sequences *Dancer* and *Shark*, and also on the natural video sequences *Book Arrival* and *Kendo*, the computed disparity map is more accurate than in sequences *Balloons* and Champagne, which leads to better results than R1 but still bellow the results of the proposed method PM(a). The proposed method PM(a) is able to refine the disparity information with good accuracy, resulting in a good quality concealed depth map. The disparity refinement process described in Section 3.1.2 is of major importance especially in large lost areas where the proposed method PM(b) and also the reference methods can not accurately conceal the depth map. Table 2 shows that the proposed method also consistently outperforms the reference one. This is due to the fact the the proposed method is capable of accurately compensating for for the different viewpoints in the reconstruction process of lost depth maps.



(a) Champagne (10% PLR) (b) Book Arrival (20% PLR)

Fig. 11 Bland-Altman plots: proposed (PM(a)) vs reference (R1)

To show the statistical behaviour of the proposed method in comparison to a reference (R1), a *Bland-Altman* plot is shown in Fig. 11a and b, corresponding to sequences Champagne e Book Arrival with losses of 20 % and 10 %, respectively. This plot shows that the

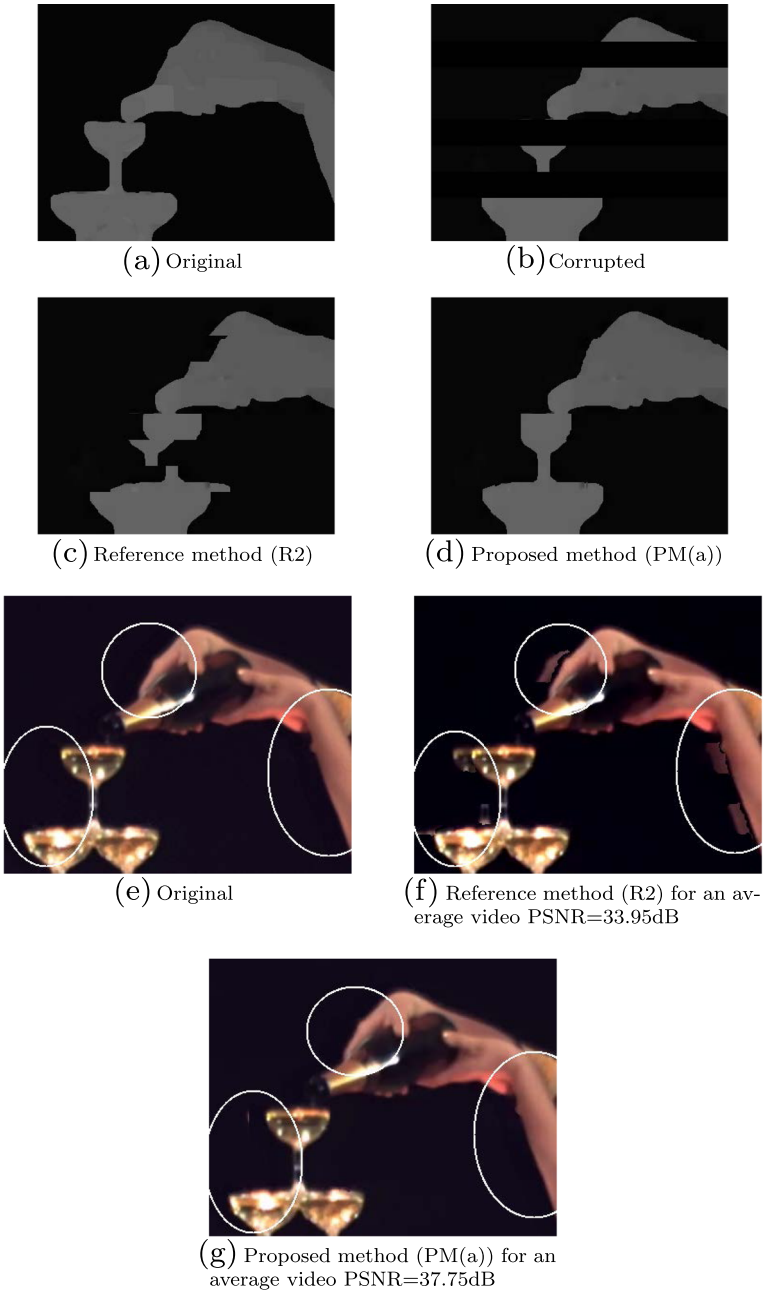


Fig. 12 Example 1: Concealed depth maps and respective synthesised views (example of 2 lost packets using packetization 3 in the 1st frame of *Champagne*), at 20 % PLR

difference between the proposed method and the reference one is greater for smaller mean PSNR, which means that its efficiency is greater for higher loss rates as it corresponds to lower PSNR (higher distortion). This beahviour is consistent throughout all sequences.

The objective results can be subjectively confirmed in Figs. 12 and 13, where the synthesised images are shown with the respective depth maps. The example of Fig. 12 shows a region detail of the original depth map (Fig. 12a), the corrupted depth with two lost

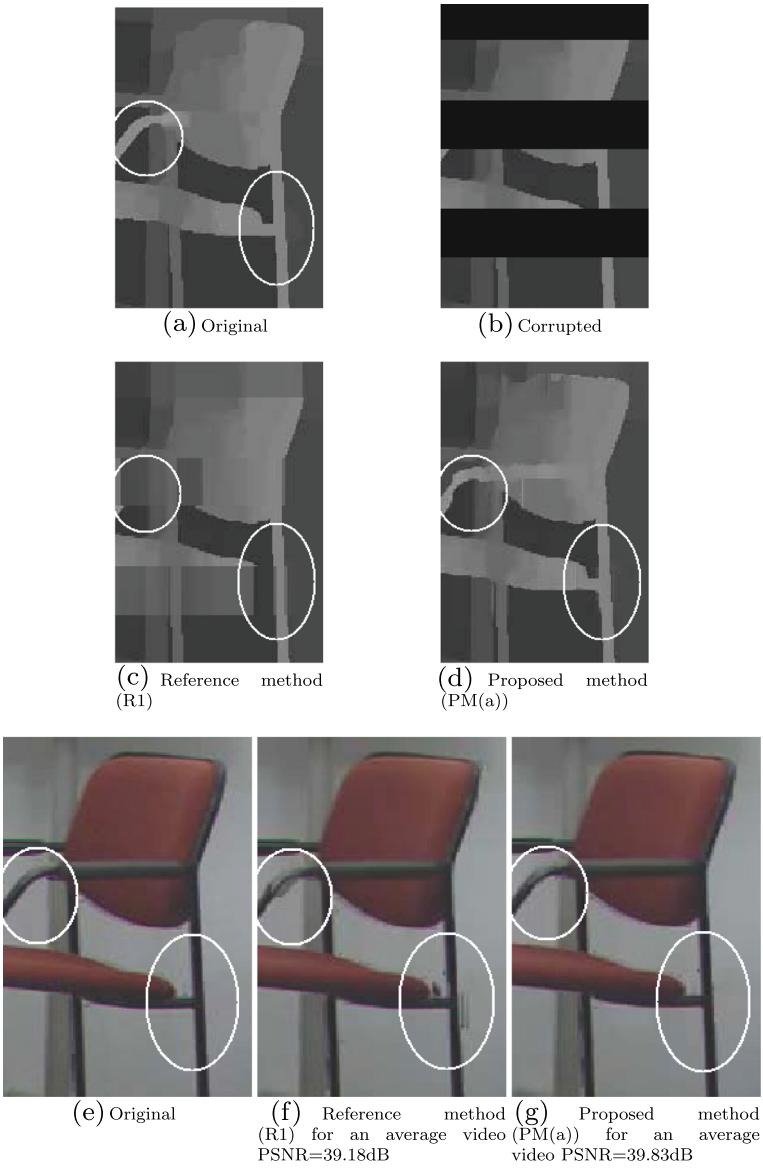


Fig. 13 Example 2: Concealed depth maps and respective synthesised views (example of 2 lost packets using packetization 3 in the 1st frame of *Book Arrival*), at 20 % PLR

packets (Fig. 12b), depth map recovered with the reference method (Fig. 12c) and a depth map recovered with the proposed method (Fig. 12d). The bottom images of Fig. 12 show the correspondent synthesised views using the original depth map (Fig. 12e), depth map concealed with the reference method R2 (Fig. 12f) and the synthesised views using the proposed method (Fig. 12g). These details clearly show the type of artifacts caused by synthesis with inaccurate depth.

The highlighted regions of Figs. 12 and 13 show the impact of depth map accuracy in view synthesis. The detailed regions show that even when large depth areas are lost, the proposed method is able to reconstruct the lost regions with very good quality. In the case where overlapped and thin objects exist in the image, like in the case of the chair of Fig. 13, the proposed method has also good performance because it is able to restore the broken contours with higher fidelity from the corresponding colour image.

When a small number of objects are present in the scene, there are fewer variations of depth, and consequently the depth map is more homogeneous and with less corresponding contours. In this case it is much easier to conceal the errors of a depth map and the performance of the proposed method is, in some cases, identical to the reference methods. When large variations of depth values are present in the depth map, the results show that the proposed method significantly outperforms the reference ones. By observing the details of the synthesised view, when abrupt changes in depth values occur, the proposed method is able to achieve a better recovery than the reference method.

5 Conclusion

This paper described a novel method for spatial error concealment of intra-coded depth maps, which is able to efficiently recover lost depth areas by first reconstructing the most significant edges in the corrupted depth map, using both colour image information and curve fitting with Bézier curves. It was found that higher quality views are obtained in comparison with the common spatial weighted interpolation and disparity compensation techniques. The results show that even when large depth areas are lost, it is possible to achieve good spatial error concealment efficiency, demonstrated by increased quality of virtual views. The concealment method proposed in this paper does not depend on any particular type of coding algorithm, which makes it a promising candidate for implementation in decoders of 3D video delivered over error-prone networks, using the multivew video-plus-depth format.

References

1. Ali A, Karim H, Arif N, Sali A (2010) Depth image-based spatial error concealment for 3-D video transmission. In: 2010 IEEE student conference on research and development (SCOReD), pp 421–425
2. Canny J (1986) A computational approach to edge dxetection. IEEE Trans Pattern Anal Mach Intell PAMI-8(6):679–698
3. Chen Y, Wang Y, Ugur K, Hannuksela M, Lainema J, Gabbouj M (2008) The emerging MVC standard for 3D video services. EURASIP J. ASP 2009:8:1–8:13
4. Chen Z, Zhou P, He Y (2002) Fast Integer Pel and Fractional Pel Motion Estimation for JVT. Documents JVT-F017, JVT 6th Meeting, Awaji Island, Japan
5. Chung B, Yim C (2014) Hybrid error concealment method combining exemplar-based image inpainting and spatial interpolation. Signal Process Image Commun 29(10):1121–1137

6. Chung T, Sull S, Kim C (2010) Frame loss concealment for stereoscopic video based on inter-view similarity of motion and intensity difference. In: 2010 17th IEEE international conference on image processing (ICIP), pp 441–444
7. Chung T, Sull S, Kim C (2011) Frame loss concealment for stereoscopic video plus depth sequences. IEEE Trans Consum Electron 57(3):1336–1344
8. Ekmekcioglu E, Velisavljevic V, Worrall S (2011) Content Adaptive Enhancement of Multi-View Depth Maps for free Viewpoint Video. IEEE J Sel Top Signal Process 5(2):352–361
9. Hasan M, Arnold J, Frater M (2014) Subjective evaluation and statistical analysis for improved frame-loss error concealment of 3d videos. In: visual communications and image processing conference, 2014 IEEE, pp 181–184
10. Hewage C, Martini M (2011) Joint error concealment method for backward compatible 3D video transmission. In: 2011 IEEE 73rd vehicular technology conference (VTC Spring), pp 1–5
11. Hewage C, Martini M (2012) Edge-based reduced-reference quality metric for 3-D video compression and transmission. IEEE J Sel Top Signal Process 6(5):471–482
12. Hewage C, Worrall S, Dogan S, Kondoz A (2008) Frame concealment algorithm for stereoscopic video using motion vector sharing. In: 2008 IEEE international conference on multimedia and expo, pp 485–488
13. Hewage C, Worrall S, Dogan S, Kondoz A (2008) A novel frame concealment method for depth maps using corresponding colour motion vectors. In: 3DTV conference: the true vision - capture, transmission and display of 3D video, pp 149–152
14. http://iphome.hhi.de/suehring/tml/: SuehringJ M17H.264/AVC, Last accessed on 02/09/2015
15. http://opencv.itseez.com/: OpenCV-2.1.0, Last accessed on 02/09/2015
16. http://vision.middlebury.edu/stereo/eval/: Middlebury Stereo Evaluation, Last accessed on 02/09/2015
17. Hughes JF, van Dam A, McGuire M, Sklar DF, Foley JD, Feiner SK, Akeley K (1990) Computer graphics, principles and practice. Addison Wesley
18. Jain R, Ramakrishnan KK, Chiu DM (2010) Definition of a general and intuitive loss model for packet networks and its implementation in the netem module in the linux kernel. Technical Report, Digital Equipment Corporation
19. Khattak S, Maugey T, Hamzaoui R, Ahmad S, Frossard P (2015) Temporal and inter-view consistent error concealment technique for multiview plus depth video. IEEE Trans Circ Syst Video Technol PP(99):1–1
20. Kim Y, Kim J, Sohn K (2007) Fast disparity and motion estimation for multi-view video coding. IEEE Transactions on Consumer Electronics 53(2):712–719
21. Kordelas A, Politis I, Dagiuklas T (2015) Transport analysis and quality evaluation of mvc video streaming. In: multimedia tools and applications, pp 1–26
22. Kosov S, Thormahlen T, Seidel H (2009) Accurate real-time disparity estimation with variational methods. In: advances in visual computing, lecture notes in computer science, vol 5875. Springer, Berlin Heidelberg, pp 796–807
23. Lie WN, Lin GH (2015) Error concealment for the transmission of H.264/AVC-compressed 3D video in color plus depth format. J Visual Commun Image Represent 32:237–245
24. Lin TL, Chang TE, Huang GX, Chou CC, Thakur US (2014) Improved interview video error concealment on whole frame packet loss. J Vis Commun Image Represent 25(8):1811–1822
25. Liu S, Chen Y, Wang Y, Gabbouj M, Hannuksela M, Li H (2008) Frame loss error concealment for multiview video coding. In: international symposium on circuits and systems, 2008. ISCAS 2008. IEEE, pp 3470–3473
26. Liu X, Peng Q, Fan X (2011) Frame loss concealment for multi-view video plus depth. In: 2011 IEEE 15th international symposium on consumer electronics (ISCE), pp 208–211
27. Liu Y, Wang J, Zhang H (2010) Depth image-based temporal error concealment for 3-D video transmission. IEEE Trans Circ Syst Video Technol 20(4):600–604
28. Marcelino S, Assuncao P, De Faria SMM, Soares S (2011) Error recovery of image-based depth maps using bezier curve fitting. In: IEEE international conference on image processing (ICIP) pp 2293–2296
29. Marcelino S, Faria S, Pepion R, Callet P, Soares S, Assuncao P (2015) Quality evaluation of depth map error concealment using a perceptually-aware objective metric. In: 3DTV-Conference: the true vision - capture, transmission and display of 3D video (3DTV-CON), pp 1–4
30. Merkle P, Morvan Y, Smolic A, Farin D, Muller K, de With P, Wiegand T (2008) The effect of depth compression on multiview rendering quality. In: 3DTV conference: the true vision - capture, transmission and display of 3D video, pp 245–248
31. Micallef B, Debono C (2010) Error concealment techniques for multi-view video. In: Wireless days (WD), 2010 IFIP, pp 1–5

32. Micallef B, Debono C, Farrugia R (2011) Performance of enhanced error concealment techniques in multi-view video coding systems. In: 2011 18th international conference on systems, signals and image processing (IWSSIP), pp 1–4
33. Ostermann J, Bormans J, List P, Marpe D, Narroschke M, Pereira F, Stockhammer T, Wedi T (2004) Video coding with H.264/AVC: tools, performance, and complexity. IEEE Circ Syst Mag 4(1):7–28
34. Rares A, Reinders M, Biemond J (2005) Edge-based image restoration. IEEE Trans Image Process 14(10):1454–1468
35. Saponara S, Blanch C, Denolf K, Bormans J (2003) The JVT advanced video coding standard: Complexity and performance analysis on a tool-by-tool basis. In: packet video workshop (PVW03)
36. Soares L, Pereira F (2004) Spatial shape error concealment for object-based image and video coding. IEEE Trans Image Process 13(4):586–599
37. Stankiewicz O, Wegner K, Domanski M (2010) Error concealment for MVC and 3D video coding. In: picture coding symposium (PCS), pp 498–501
38. Tai SC, Wang CC, Hong CS, Luo YC (2015) An effiicient full frame algorithm for object-based error concealment in 3d depth-based video. Multimedia tools and applications pp 1–21
39. Tanimoto M, Fujii T, Suzuki K (2009) View synthesis algorithm in view synthesis reference software 3.5 (VSRS3.5) Document M16090, ISO/IEC JTC1/SC29/WG11 (MPEG)
40. Vetro A, Wiegand T, Sullivan G (2011) Overview of the stereo and multiview video coding extensions of the H.264/MPEG-4 AVC Standard. Proc IEEE 99(4):626–642
41. Wiegand T, Sullivan G, Bjontegaard G, Luthra A (2003) Overview of the H.264/AVC video coding standard. IEEE Trans CSVT 13(7):560–576
42. Xiang X, Zhao D, Ma S, Gao W (2011) Auto-regressive model based error concealment scheme for stereoscopic video coding. In: 2011 IEEE international conference on acoustics, speech and signal processing (ICASSP) pp. 849–852
43. Yan B A novel H.264 based motion vector recovery method for 3D video transmission. IEEE Transactions on Consumer Electronics 53(4):1546–1552
44. Yan B, Zhou J (2012) Efficient frame concealment for depth image based 3D video transmission. IEEE Trans Multimed PP(99):1

Pedro A. Amado Assunção received the Licenciado and M.Sc. degrees in Electrical Engineering from the University of Coimbra, Portugal, in 1988 and 1993, respectively, and the Ph.D. in Electronic Systems Engineering from the University of Essex, UK, in 1998. He is currently Professor of Electrical and Computer Engineering, Electronics and Multimedia Communication Systems at the Polytechnic Institute of Leiria and senior researcher at Instituto de Telecomunicacoes, Portugal. He has been involved in several projects in the field of multimedia communications and he served as a reviewer and/or technical programme committe of many international conferences and journals. He is author/co-author of more than one hundred papers published in international scientific conferences and journals, one book, eight book chapters and four US patents. His current research interests include UHD, multiview and Plenoptic video coding, communications and processing, video codec complexity control and networking adaptation based on user-driven approaches, error concealment and quality assessment. He is the Chair of the EU COST Action IC1105, 3D Content Creation, Coding and Transmission over Future Media Networks (3DConTourNet) and Senior Member of the IEEE.

Sylvain Marcelino received the B.Sc. in electrical engineering from Escola Superior de Tecnologia e Gestão of Instituto Politécnico de Leiria, Portugal (2006), M.Sc. and PhD degrees from the Universidade de Trás-os-Montes e Alto Douro (UTAD), Portugal, in 2009 and 2016, respectively. He worked on a research project dealing with fast transcoding of H.264/AVC compressed streams at Instituto de Telecomunicações, Portugal and his research interests include digital signal and image processing and coding, 3D and multiview video and error concealment.

Salviano Soares received the degree of Licenciado in Electrical Engineering from the University of Trásos-Montes e Alto Douro (UTAD), Vila Real, Portugal, in 1991, and the MSc and PhD in Electronic and Telecommunications, from the University of Aveiro, Portugal, in 1996 and 2005, respectively. He is an assistant professor in the Engineering Department, Science and Technology School, UTAD. He is also a researcher at IEETA - Institute of Electronics and Telematics Engineering, University of Aveiro, Portugal. He has been involved in R&D projects in the field of wireless applications, in the area of voice quality evaluation and in smart homes and Ambient Assisting Living. He leads the R&D team to deliver engineering solutions for vending operators namely the project Vending Telemetry, incentive system for technology research and development in companies - SI I&DT, funded by National Strategic Reference Framework (QREN in Portugal). He is author/co-author of more than sixty scientific/technical papers, two book chapters, portuguese marks and one international patent. His main research interests are in the signal processing and its applications.

Sérgio M. M. de Faria received the Engineering degree in Electrical Engineering in 1988 and the M.Sc. degree in Electrical Engineering in 1992, both from the Universidade de Coimbra, Portugal, and the Ph.D. degree in Electronics and Telecommunications from the University of Essex, UK, in 1996. He is Professor with the Department of Electrical Engineering, in Escola Superior de Tecnologia e Gestão of Instituto Politécnico de Leiria, Portugal, since 1990. He has collaborated in master courses with Universidade de Coimbra, Portugal. He is an Auditor with A3ES organization for Electrical and Electronic Engineering courses in Portugal. He is a Senior Researcher with Instituto de Telecomunicações, Portugal. His research interests include 2D/3D image and video processing and coding, motion representation, and medical imaging. In this field, he has published edited 2 books and authored 5 book chapters, 18 journal papers, and over 110 referred conference papers. He has been participating and he is responsible for several, national and international (EU), funded projects. He is an Area Editor of Signal Processing: Image Communication. He has been a Scientific and Program Committee Member of many international conferences. He is a reviewer for several international scientific journals and conferences (IEEE, IET and EURASIP). He is a Senior Member of the IEEE.