A DISTORTION EVALUATION FRAMEWORK IN 3D VIDEO VIEW SYNTHESIS

Andrei I. Purica⋆, Marco Cagnazzo†, Beatrice Pesquet-Popescu*, Frederic Dufaux*, Bogdan Ionescu†

⋆LTCI, CNRS, Télécom ParisTech, Université Paris-Saclay, 75013, Paris, France
†University Politehnica of Bucharest, 061071, Romania
Email:{purica, cagnazzo, pesquet, dufaux }@telecom-paristech.fr, bionescu@imag.pub.ro

ABSTRACT
Demand for 3D content increased over the last years and more attention was dedicated to methods used in 3D content creation and transmission. View synthesis methods introduce localized artifacts when creating new virtual views. Evaluating these methods require therefore a different approach, in order to identify and emphasize synthesis artifact prone areas, while diminishing the impact of other types of artifacts, such as those produced by quantization during the video coding. In this paper, we propose an artifact evaluation technique which was shown to provide a better differentiation of synthesis artifacts, while additional information can be extrapolated from the results about the spatial localization of typical synthesis artifacts and their impact on overall image quality.

Index Terms — View Synthesis, Visual quality assessment, Multi-View Video, SSIM, Depth-Image-Based-Rendering

1. INTRODUCTION
3D content has become widespread in the past years. Technology advancements in display, transmission and encoding fields allow 3D to be used in common everyday applications. Some of the typical usage scenarios involve immersive tele-conference systems, free view point television (FTV) [11], medical applications, gaming and entertainment [2].

Due to the high quantity of information contained in 3D data, new encoding and compression methods are required. Several formats exists for storing 3D video data. Some of the most commonly used are stereo video, MultiView Video and Multi-View-Video-plus-Depth (MVD) [3]. MVV format is comprised of a set of N video sequences, representing the same scene and acquired simultaneously by a system of N cameras, in different spatial configurations. Most commonly, a parallel camera line is used. MVD format is similar with cameras, in different spatial configurations. Most commonly, methods in the first category, such as DIBR, use explicit geometry information to warp pixels from the original view to their correct positions in the virtual view [6] [7]. Methods in the second category rely only on implicit geometry, for example pixel correspondences which can be computed using methods such as optical flow [8] [9]. Finally, methods in the third category use filtering and interpolation to synthesize the view. Examples include light field rendering [10], concentric mosaics [11] or lumigraph [12].

The Moving Picture Experts Group (MPEG), expressed interest in MVD formats for their ability to support 3D video applications. The idea was to develop a 3D extension of the High Efficiency Video Coding (HEVC) standard [13]. An experimental framework was developed to conduct experiments [14] with 3D video. A 3D-HEVC test model (3D-HTM) [15] was also created, which includes the View Synthesis Reference Software 1DFast (VSRS-1DFast) designed for parallel camera systems.

Some of the main issues in view synthesis are caused by areas which are hidden in the original view (e.g., background covered by foreground objects) but become visible when changing the point of view. These areas are referred to as disocclusions, and in the synthesized views they produce holes in the image. Traditionally, this problem is fixed using inpainting [16] [17] [18]. However, different techniques can be used that exploit the temporal correlations in a video sequence. The scene background can be extracted from multiple time instances of the video sequence, and used to fill the disoccluded areas [19] [20]. Temporal correlations can also be used. The authors in [21] compute motion vector fields in the rendered view to improve the synthesis. In [22] the authors use block-based motion estimation in the reference views, and retrieve information about disoccluded areas by warping the start and end point of the motion vectors.
Other sources of errors in view synthesis are related to the quality of depth maps. While artificially generated sequences have almost ideal depth maps, real sequences will have distortions and errors in the depth maps, due to the limitation of the recording devices. Low quality depth maps lead to distortions of foreground object and larger or incorrect holes (e.g. new holes might be generated and existing holes may be covered incorrectly during the warping process) in the synthesized views. Most synthesis methods, preferably, use two reference views, a left and right one with respect to the virtual view position. This is favored in order to avoid border disocclusions [23]. Another issue is the texture-depth alignment, which may lead to a mismatch of the left and right warping of the reference views. Other methods, such as [24] or [25], pre-process the depth maps, with the goal of providing a better foreground-background transition. In [26], a rendering technique is presented based on a background-foreground separation filtering step.

In addition to errors caused by the actual warping process, all synthesized views are also affected by the encoding quality of the reference views or depth maps. When using encoded reference views, the pixels that are warped in the synthesized view are subject to an absolute quantization error of up to half the quantization step. Encoded depth maps will also be subjected to quantization errors, especially since they are usually encoded using higher QPs. However, the quantization errors in depth maps will impact the synthesized view in a different manner. A small error in the depth map, results in warping the pixel to a slightly different position in the virtual view. While this is not a big issue for pixels located in areas with uniform texture, it can create very high distortions on the edges of objects (consider a scene with a black object on white background). Finally, another source of errors for depth maps is the quantization of real depth values to, usually, 256 levels.

Because of the multiple sources of errors which are usually not uniformly spread across the image, unlike encoding errors, evaluating the quality of a synthesized image is not a trivial problem. Measures such as Peak-Signal-to-Noise-Ratio (PSNR) provide a good objective evaluation but fail to emphasize the errors caused by object distortions. Evaluation methods that take into account the structure of the image were created, one of the most popular being the structural similarity based metric (SSIM) [27] (see Sec. 3.1). While SSIM takes into account the structural distortions of an image, small differences in background color reproduction might mask the impact of important artifacts. As discussed above, the majority of high errors in view synthesis are mostly located close to the edges of foreground objects. In this paper we propose a new view synthesis evaluation technique, based on SSIM, which focuses on comparing view synthesis artifacts around sensible, error-prone areas of the image. Two different methods are used for selecting the areas of interest in the evaluation. Firstly, we analyze the distribution of errors and separate high synthesis errors from quantization ones. A second approach is focused on directly evaluating the areas predicted differently by the two tested methods. We show this technique to bring a better differentiation of synthesis methods with respect to the impact of synthesis artifacts on the image quality. Also, additional information can be extrapolated on the spatial localization of distortions when compared to an SSIM or PSNR evaluation.

The rest of this paper is organized as follows. The next section presents the view synthesis methods used for evaluation. In Sec. 3 we describe the proposed evaluation technique. In Sec. 4 we show the results we obtained and Sec. 5 concludes the paper.

2. VIEW SYNTHESIS METHODS

We evaluate three different view synthesis methods. The DIBR implementation of VSRS-1DFast [15], a method that uses a filtering technique described in [26] and a method that blends temporal prediction with DIBR synthesis [28].

All methods use depth maps to compute disparity. Usually, depth maps are given with inversely quantized values between [0 255]. When dealing with 1D camera systems, disparity only has an x component which can be computed from the depth maps of the reference views as [2]:

\[ d(k) = f \cdot B \left[ \frac{Z(k)}{255} \left( \frac{1}{Z_{min}} - \frac{1}{Z_{max}} \right) + \frac{1}{Z_{max}} \right] \]  

(1)

where \( Z \) is the inverse quantized depth, \( Z_{min} \) and \( Z_{max} \) are the minimum and maximum depth values respectively, \( B \) is the baseline (i.e. the distance between the synthesized view and base one) and \( f \) is the focal length of the camera.

The first method (VSRS-1DFast) uses two or three texture and depth views and synthesizes any number of intermediary views. This method can work with sub-pixel precision up to a factor of four. The warping, interpolation and hole filling are carried out line-wise. Two reliability maps are used to mark disoccluded areas for filling. And a similarity enhancement step adapts the histograms of the left and right warped views. The two warped images, from left and right base views, can be either combined, or the disocclusions in one image can be filled using the other one.

The filtering method presented in [26] uses an additional intermediary step to perform a filtering of the combined, upsampled, warped images in order to perform a foreground background separation and interpolation.

The method in [28] uses temporal correlations to produce multiple predictions of a synthesized frame from different time instants and uses an adaptive blending method to combine the temporal and inter-view predictions. The motion vector fields (MVF) in the synthesized views (\( v_r \)) are obtained by imposing an epipolar constraint between the MVF in the reference view (\( v_r \)) and the disparity fields at two time instants (\( d_l \) and \( d_r \)) as shown in Eq. 2

\[ v_r(k + d_l(k)) = v_r(k) + d_l(k) - d_r(k) \]  

(2)
where \( k = (x, y) \) is a position in the image. Using a future and a past reference frame for motion estimation in each base view allows the computation of four MVFs. Thus, four temporal predictions and two inter-view predictions are obtained. An adaptive blending method reduces the motion estimation errors by using only inter-view prediction for these areas.

3. PROPOSED SYNTHESIS EVALUATION

As discussed in Sec. 1, view synthesis methods should be evaluated not only with respect to PSNR, but also with respect to a metric that takes into account the structure of the image, like SSIM. However the main issues in synthesis are the disoccluded areas and the distortion of foreground object and other artifacts caused by texture-depth misalignment or imprecise depth maps. Thus, when comparing different methods, areas around foreground object edges and disocclusions should be emphasized in the evaluation. Smooth background areas typically have low errors (see Sec. 4). It is reasonable to assume that methods which bring only small corrections in areas will not have a visual impact even though PSNR gains can be achieved. In what follows, we will describe the SSIM metric and show how a selection of artifact prone areas can be achieved for better evaluating view synthesis methods.

3.1. Structural Similarity index (SSIM)

In [27], Wang et al. assume that the human visual system is highly focused on the perception of structural information of a scene. The proposed measure is designed to assess the degradation of structural information. SSIM separates the similarity measurement in three components: luminance, contrast and structure. A condensed form of the SSIM is given by:

\[
SSIM(r, d) = \frac{(2\mu_r \mu_d + C_1)(2\sigma_{rd} + C_2)}{(\mu_r^2 + \mu_d^2 + C_1)(\sigma_r^2 + \sigma_d^2 + C_2)}
\]

(3)

where, \( r \) and \( d \) refer to windows in the reference and distorted images, \( \mu_r \) and \( \mu_d \) are the means of \( r \) and \( d \), \( \sigma_r \) and \( \sigma_d \) are the standard deviations and \( \sigma_{rd} \) is the correlation coefficient between \( r \) and \( d \). \( C_1 \) and \( C_2 \) are two variables used to stabilize the division with weak denominator. The score of an image can then be obtained by centering the windows in each pixel and averaging the SSIM index, this is known as mean SSIM (MSSIM).

3.2. Histogram based area selection

When testing two view synthesis methods, a first way of selecting the areas prone to synthesis errors would be to look for pixels which have a relative high absolute error. This can provide a good indication on the quality of the synthesis methods. Errors produced by the quantization during the encoding of the reference views and errors caused by depth quantization or interpolation process are usually uniformly spread and do not necessarily depend on the structure of the scene or the view synthesis method employed. This can also be observed in Fig. 1 where two binary masks are shown. Black indicates pixels that have an absolute error larger than twice the mean absolute error. Fig. 1(a) shows the mask for a frame encoded with 3D-HEVC at QP 25 and Fig. 1(b) is obtained from the same frame synthesized with VSRS-1DFast from non-encoded reference views. It is easily noticeable that in the case of encoding, high errors are spread across the image. In the case of synthesis, impactful errors are concentrated and their spatial positioning is dependent on the structure of the scene. Focusing the synthesis evaluation on these areas can provide a better indication of the methods quality for object distortion, while ignoring other less impactful error sources, like quantization errors produced by encoding.

![Fig. 1. Binary masks on Balloons sequence frame 1. Black indicates pixels with high absolute errors.](image)

(a) 3D-HEVC encoding  
(b) Synthesis

As discussed in Sec. 1, view synthesis methods should be evaluated not only with respect to PSNR, but also with respect to a metric that takes into account the structure of the image, like SSIM. However the main issues in synthesis are the disoccluded areas and the distortion of foreground object and other artifacts caused by texture-depth misalignment or imprecise depth maps. Thus, when comparing different methods, areas around foreground object edges and disocclusions should be emphasized in the evaluation. Smooth background areas typically have low errors (see Sec. 4). It is reasonable to assume that methods which bring only small corrections in areas will not have a visual impact even though PSNR gains can be achieved. In what follows, we will describe the SSIM metric and show how a selection of artifact prone areas can be achieved for better evaluating view synthesis methods.

3.1. Structural Similarity index (SSIM)

In [27], Wang et al. assume that the human visual system is highly focused on the perception of structural information of a scene. The proposed measure is designed to assess the degradation of structural information. SSIM separates the similarity measurement in three components: luminance, contrast and structure. A condensed form of the SSIM is given by:

\[
SSIM(r, d) = \frac{(2\mu_r \mu_d + C_1)(2\sigma_{rd} + C_2)}{(\mu_r^2 + \mu_d^2 + C_1)(\sigma_r^2 + \sigma_d^2 + C_2)}
\]

(3)
The binary mask used for synthesis distorted area selection can then be computed as:

\[
B(x, y) = \begin{cases} 
0 & \text{if } |(I_r(x, y) - I_d(x, y))| < T \\
1 & \text{if } |(I_r(x, y) - I_d(x, y))| \geq T 
\end{cases} 
\]

(6)

where \(I_r\) and \(I_d\) are the reference and distorted images.

However, this approach can produce different masks for two evaluated synthesis methods (\(B_{d1}\) and \(B_{d2}\)). In order to assure a consistent evaluation in both compared methods, the SSIM index should be computed using a single mask. This can be achieved by performing the evaluation in the locations obtained by merging the two masks as shown in Eq. \(7\):

\[
B_{hist}(x, y) = B_{d1}(x, y) \lor B_{d2} 
\]

(7)

where \(\lor\) is the logical or operation.

The score of each method can then be obtained by averaging the SSIM index over all pixels selected with the binary mask:

\[
SSIM_{hist}^{d1}(I_r, I_d_1, I_d_2) = \frac{1}{M \times N} \sum_{x=1}^{M} \sum_{y=1}^{N} B_{hist}(x, y) \sum_{x=1}^{M} \sum_{y=1}^{N} SSIM(r(x, y), d1(x, y))) \times B_{hist}(x, y) 
\]

(8)

where \(d_1\) and \(d_2\) refer to the two distorted images obtained by different synthesis methods and \(M, N\) are the width and height of the image.

### 3.3. Error prone area selection

Another option for selecting relevant spatial locations that need to be evaluated when comparing synthesis methods, is to look directly at the differences between methods. We can select these areas by generating a new selection mask containing all areas which were rendered differently by the two methods as shown in Eq. \(9\):

\[
B_{epas}(x, y) = \begin{cases} 
0 & \text{if } |(I_{d1}(x, y) - I_{d2}(x, y))| < T \\
1 & \text{if } |(I_{d1}(x, y) - I_{d2}(x, y))| \geq T 
\end{cases} 
\]

(9)

where \((T)\) is a threshold.

When comparing two synthesis methods, we are interested to see their behavior in areas where pixels are predicted differently. Evaluating areas where both methods provide similar pixel predictions will not offer a good comparison of the methods. Establishing a selection threshold in this case is easier. Since we are interested in relative large differences, the mean absolute error can provide a good threshold.

### 4. EXPERIMENTAL RESULTS

In order to verify our evaluation method we use the 3D-HEVC test model (3D-HTM). The encoder and renderer configurations follow the Common Test Conditions (CTCs) for conducting experiments with 3D-HEVC, more details can be found here [29]. The tested video sequences are: Balloons, Kendo, NewspaperCC and PoznanHall2. The first three sequences have a resolution of \(1024 \times 768\) with 30 fps and a total of 300 frames. The later has a resolution of \(1920 \times 1088\) with 25 fps and a total of 200 frames. For each sequence, we use two encoded reference views with their associated depth maps and synthesize an intermediate view. For Balloons and Kendo sequences, we use views 1&5 as reference and synthesize view 3. Views 2&6 and 5&7 are used as reference for NewspaperCC and PoznanHall2 sequences respectively, while views 4 and 6 are synthesized. The encoding is performed with 3D-HEVC using four QPs for texture: 25, 30, 35, 40. Different QPs are used for the depth maps, as recommended by the CTCs: 34, 39, 42, 45.

The synthesis methods we evaluate are detailed in Sec.2. For VSRS-1DFast we use the CTCs recommended configuration. The similarity enhancement and sub pixel precision options are active. The warping and filtering technique (Wf) presented in [26] uses a filtering window of size 7 and a sub pixel precision factor of 1/4. The method based on temporal and inter-view prediction blending (P+Badapt) in [28] uses an optical flow implementation for motion estimation [30] and a temporal prediction distance of two. Each method is evaluated using PSNR and MSSIM. SSIM\(_{hist}\) and SSIM\(_{epas}\) are used to evaluate and compare Wf and P+Badapt with VSRS-1DFast.

Table 1 shows the PSNR results for the three tested methods: VSRS-1DFast, Wf and P+Badapt. On the bottom of the table we can see the average result across sequences and the last row shows the gain obtained by the later two methods. As can be seen both Wf and P+Badapt out perform VSRS-1DFast while the best results are obtained by P+Badapt. Another aspect of interest is that the gain remains relatively sta-
Table 1. Average PSNR for all tested methods and sequences at each QP.

<table>
<thead>
<tr>
<th>Sequence</th>
<th>VSRS-1DFast</th>
<th>Wf</th>
<th>P+Badapt</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSNR (dB)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QPs</td>
<td>25 30 35 40</td>
<td>25 30 35 40</td>
<td>25 30 35 40</td>
</tr>
<tr>
<td>Balloons</td>
<td>34.37 34.07 33.43 32.41</td>
<td>34.39 34.14 33.52 32.31</td>
<td>34.74 34.45 33.8 32.72</td>
</tr>
<tr>
<td>Kendo</td>
<td>34.98 34.31 33.77 32.75</td>
<td>35.37 34.9 34.15 33.08</td>
<td>35.37 34.87 34.13 33.06</td>
</tr>
<tr>
<td>NewspaperCC</td>
<td>32.29 30.95 28.78 28.31</td>
<td>32.81 29.69 29.39 28.83</td>
<td>29.85 29.74 29.44 28.9</td>
</tr>
<tr>
<td>PoznanHall2</td>
<td>36.24 35.87 35.36 34.55</td>
<td>36.35 36.02 35.51 34.77</td>
<td>36.49 36.2 35.7 34.86</td>
</tr>
<tr>
<td>Average</td>
<td>33.70 33.37 32.83 32.32</td>
<td>33.98 33.69 33.14 32.3</td>
<td>34.11 33.82 33.27 32.38</td>
</tr>
<tr>
<td>∆PSNR</td>
<td>-</td>
<td>0.28</td>
<td>0.32 0.31 0.3</td>
</tr>
</tbody>
</table>

Table 2. Average MSSIM for all tested methods and sequences at each QP.

<table>
<thead>
<tr>
<th>Sequence</th>
<th>VSRS-1DFast</th>
<th>Wf</th>
<th>P+Badapt</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSSIM</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QPs</td>
<td>25 30 35 40</td>
<td>25 30 35 40</td>
<td>25 30 35 40</td>
</tr>
<tr>
<td>Balloons</td>
<td>.9583 .9541 .9460 .9322</td>
<td>.9571 .9530 .9450 .9313</td>
<td>.9597 .9556 .9479 .9343</td>
</tr>
<tr>
<td>Kendo</td>
<td>.9635 .9593 .9528 .9430</td>
<td>.9631 .9590 .9526 .9429</td>
<td>.9638 .9600 .9538 .9444</td>
</tr>
<tr>
<td>NewspaperCC</td>
<td>.9065 .8989 .8771 .8573</td>
<td>.9004 .8939 .8802 .8590</td>
<td>.9020 .8957 .8824 .8621</td>
</tr>
<tr>
<td>PoznanHall2</td>
<td>.9352 .9322 .9272 .9190</td>
<td>.9358 .9330 .9281 .9198</td>
<td>.9370 .9340 .9290 .9208</td>
</tr>
<tr>
<td>Average</td>
<td>.9384 .9339 .9258 .9129</td>
<td>.9391 .9347 .9265 .9133</td>
<td>.9406 .9363 .9233 .9154</td>
</tr>
<tr>
<td>∆MSSIM</td>
<td>-</td>
<td>.0007</td>
<td>.0009 .0007 .0004</td>
</tr>
</tbody>
</table>

Table 3. VSRS-1DFast and Wf evaluation for all QPs with SSIM_{hist} and SSIM_{epas}.

<table>
<thead>
<tr>
<th>Sequence</th>
<th>VSRS-1DFast</th>
<th>Wf</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>hist epas</td>
<td>hist epas</td>
</tr>
<tr>
<td>QPs</td>
<td>25 30 35 40</td>
<td>25 30 35 40</td>
</tr>
<tr>
<td>Balloons</td>
<td>.8546 .9186 .8550 .9120</td>
<td>.8464 .9112</td>
</tr>
<tr>
<td>Kendo</td>
<td>.8798 .9279 .8743 .9192</td>
<td>.8671 .9239</td>
</tr>
<tr>
<td>NewspaperCC</td>
<td>.6062 .8309 .5592 .8173</td>
<td>.6194 .8358</td>
</tr>
<tr>
<td>PoznanHall2</td>
<td>.7517 .8922 .7359 .8834</td>
<td>.7466 .8932</td>
</tr>
<tr>
<td>Average</td>
<td>.7731 .8924 .7561 .8830</td>
<td>.7710 .8910</td>
</tr>
<tr>
<td>∆</td>
<td>-</td>
<td>- .0021</td>
</tr>
</tbody>
</table>

Table 4. VSRS-1DFast and P+Badapt evaluation for all QPs with SSIM_{hist} and SSIM_{epas}.

<table>
<thead>
<tr>
<th>Sequence</th>
<th>VSRS-1DFast</th>
<th>P+Badapt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>hist epas</td>
<td>hist epas</td>
</tr>
<tr>
<td>QPs</td>
<td>25 30 35 40</td>
<td>25 30 35 40</td>
</tr>
<tr>
<td>Balloons</td>
<td>.8615 .9138 .8585 .9080</td>
<td>.8702 .9195</td>
</tr>
<tr>
<td>Kendo</td>
<td>.8844 .9238 .8779 .9159</td>
<td>.8879 .9237</td>
</tr>
<tr>
<td>NewspaperCC</td>
<td>.6073 .8265 .5514 .8121</td>
<td>.6245 .8366</td>
</tr>
<tr>
<td>PoznanHall2</td>
<td>.7494 .9077 .7312 .9034</td>
<td>.7630 .9122</td>
</tr>
<tr>
<td>Average</td>
<td>.7756 .8929 .7548 .8849</td>
<td>.7642 .8890</td>
</tr>
<tr>
<td>∆</td>
<td>-</td>
<td>- .0085</td>
</tr>
</tbody>
</table>

Table 4 shows the results obtained when evaluating Wf against VSRS-1DFast with the proposed methods: SSIM_{hist}, SSIM_{epas}. We can see that losses or gains are slightly increased and better differentiated in comparison to MSSIM results. Also, when computing the difference between the average values we no longer have a gain at low QPs. This indicates that while the Wf method improves the overall image in comparison to VSRS-1DFast, it does not provide any benefits toward reducing the object boundary distortions. The PSNR and MSSIM gains provided by this method are given by a reduction in small errors. This is expected since the method proposes a sub-pixel precision warping technique with high accuracy, without tackling the structural aspect of the scene.

The comparison results of P+Badapt and VSRS-1DFast are reported in Table 4. A significant increase in ∆ values can be observed in comparison to MSSIM. SSIM_{hist} focuses on high synthesis errors which are most likely caused by object boundary distortions, as discussed in Section 1. We can conclude that P+Badapt improves the synthesis from a structural point of view. This is also visible in Fig. 3 where P+Badapt shows noticeable improvements on object edges.

Another interesting aspect is the behavior of ∆SSIM_{hist} table across QPs. Using the MSSIM metric shows similar results, as can be observed in Table 2.
and $\Delta \text{SSIM}_{\text{epas}}$ across QPs. While the $\Delta \text{MSSIM}$ and $\Delta \text{PSNR}$ report similar values across QPs, we can see that in Table 3 $\text{SSIM}_{\text{hist}}$ has a tendency to increase at lower QPs. This behavior can be explained by the threshold selection process described in 3.2. As the QP increases the quantization errors are in turn increased and they become closer to the high synthesis errors. Thus, the evaluated areas may contain more artifacts caused by quantization, reflecting the better overall warping precision of Wf over VSRS-1DFast and masking the structural distortions. However, definitive conclusions should be drawn from a more thorough evaluation of $\text{SSIM}_{\text{hist}}$ and $\text{SSIM}_{\text{epas}}$, using smaller QP steps.

Furthermore, in Fig. 4(a) we show the behavior over time of MSSIM and our proposed evaluation technique. While MSSIM score is relatively similar across frames, variations can be observed for $\text{SSIM}_{\text{hist}}$ and $\text{SSIM}_{\text{epas}}$ scores. Additional information about a methods strengths or weaknesses can be extrapolated from analyzing these variations. Let us look for example at three time instances marked in Fig. 4(a) with vertical red lines ($t_1$, $t_2$ and $t_3$, frames 40, 58 and 85 respectively). We can clearly notice an increase in $\text{SSIM}_{\text{epas}}$ at $t_2$ in comparison to $t_1$. This is consistent with MSSIM, however, it is hardly noticeable. Let us look at the $\text{SSIM}_{\text{epas}}$ masks for the two time instances in Fig. 4(b) and 4(c) to identify the cause. We can see the error prone area marked with a red square in Fig. 4(b). In Fig. 4(c) this area is obstructed by a person walking in front of it and the errors are concealed. Also, observe that $\Delta \text{SSIM}_{\text{epas}}$ is smaller between frames 50 and 70. This points to Wf method achieving higher quality than VSRS-1DFast in this area. Obviously when the area is obstructed the gain is reduced.

At $t_3$ we can see a sudden drop in $\text{SSIM}_{\text{epas}}$ which is not viewable in MSSIM. Looking at the selection mask we can observe the person approaching another foreground object which is identified as an error prone area by the selection mask. This is marked with a red square in Fig. 4(d). To better understand why we have quality loss on this frame, let us look at the texture. In Fig. 4(e) we can see the reference frame and the VSRS-1dFast and Wf synthesized frames respectively. It is easily noticeable that both methods will have new artifacts in this area at $t_3$. This type of artifact appears due to the proximity of the two objects in the foreground. The area in-between them is not visible in the left or right base views (i.e. disoccluded area). This additional information on the tested methods, in terms of structural configuration of the scene and error prone areas, cannot be easily extrapolated by using only MSSIM or PSNR.

5. CONCLUSIONS

In this paper, we presented a distortion evaluation technique for view synthesis methods based on the SSIM metric. We compute the SSIM index on areas which are prone to synthesis errors such as object boundaries and complex textures. The area selection is performed either through a separation between structural artifacts caused by synthesis and quantization errors from the encoding process of left and right base views, or by directly selecting areas which are predicted differently by two evaluated synthesis methods. The evaluation was performed on three view synthesis methods and four multiview sequences, using 3D-HEVC encoding at four QPs 25, 30, 35 and 40, against PSNR and SSIM results. The proposed technique is shown to provide a better differentiation between
Fig. 4. Figure 4(a) - SSIM and SSIM\textsubscript{epas} over time. Figures 4(b), 4(c) and 4(d) show the selection masks for SSIM\textsubscript{epas}. Figure 4(e) shows details of NewspaperCC sequence.

synthesis methods. Also, additional information can be extrapolated about the scene structure and spatial positioning of artifacts, while providing a good indication of the impact of synthesis errors.

6. REFERENCES


