No-reference quality assessment of DIBR based synthesized images

Fábio Freire Rodrigues
Electrical and Computer Engineering Department
Instituto Superior Técnico, IST
fabio.rodrigues@tecnico.ulisboa.pt

Abstract— Nowadays, companies are exploring a new set of video based applications, such as free viewpoint video, to provide the consumers with more engaging, immersive and tailored experiences. Free viewpoint video is a video experience where the viewer selects any viewpoint to observe the visual scene, and it is even possible to provide a smooth transitioning between viewpoints. However, it is not realistic to produce and transmit views for every possible viewpoint that the user may choose. Therefore, only a limited number of views is transmitted and, based on depth image based rendering (DIBR) methods, new views are synthesized at user side. However, this synthesis process may result in some artifacts in the synthesized view, reducing the overall quality of experience (QoE). Accordingly, the quality of the synthesized views should be evaluated, using an objective quality assessment metric. Some solutions have been already proposed in the literature for synthesized image quality assessment, most of them being full-reference methods (i.e., the original view is required). In this work, a novel no-reference quality metric to evaluate synthesized images is proposed, i.e., a metric that evaluates the quality of a synthesized image without its original version being available. This metric relies on extracting image features at different phases of the synthesis process, which are fused through support vector regression (SVR), a machine learning tool.

A dataset which contains synthesized images with compression and rendering artifacts was built and used to develop and assess the proposed metric. The metric performance is compared with conventional full-reference 2D image quality assessment metrics and with two state-of-the-art image quality assessment metrics developed specifically for synthesized images. The experimental results showed that the proposed solution outperforms the state-of-the-art metrics, being able to predict the images subjective scores with a Pearson correlation coefficient close to 0.9.

Keywords— image quality assessment, multiview video-plus-depth, depth image based rendering, view synthesis, image features, synthesized image dataset.

I. INTRODUCTION

In recent years, 3D video has become increasingly popular, since it provides a more immersive and natural representation of the real world. 3D visual representation formats are currently being explored by the industry to offer the consumers more engaging and tailored experiences. To fulfill the expectation of a better quality of experience (QoE), multiview video (MVV) has become a subject of increasing interest, leading to a new set of applications since it provides a richer and more immersive experience to the user. The MVV representation consists in two or more views that are simultaneously acquired from different viewpoints. However, due to production and transmission constraints, some applications that required a high number of views were not feasible until multiview plus depth (MVD) format has been developed. In this format, not only a high number of views is acquired, with an array of cameras, but also the associated depth. At the receiver side, rendering of additional views, usually between views already received, can be performed using the texture and depth maps. Three dimensional images and video has been in high demand for use in multiple fields, from entertainment, medicine, education to surveillance. Free viewpoint video is a video application that allows the user to select from which viewpoint a recorded scene is reproduced which means that each viewer can observe the visual scene from a unique viewpoint. Based on DIBR techniques, it is possible to render videos where users can freely move a virtual camera around the 3D space. However, such applications only provide a realistic experience, if the images synthesized and delivered to the end users have high quality. To automatically evaluate the quality of the synthesized views, which has a vital importance in the overall QoE, objective quality assessment metrics views are required. The quality evaluation of synthesized views is essential to guarantee an adequate QoE since it allows: i) the encoder to decide which views should be transmitted; ii) the decoder may request additional views to obtain better overall quality, if necessary; iii) it is possible to track the media quality that is being delivered to the end-user.

Recently, several solutions to evaluate the quality of synthesized views have been proposed in the literature. Most of the available quality metrics for synthesize views are full-reference metrics, i.e., they evaluate a synthesized image using the corresponding original version. However, in several applications, it is important to assess the quality of the visual experienced by the user, which is only possible using no-reference quality metrics, i.e., metrics that do not require the original view. Furthermore, the original views may not exist, and, therefore, the development of no-reference metrics for 3D synthetized views is an important research topic and will be addressed in the context of this work.

II. MULTIVIEW VIDEO: CONVEPTS,CODING AND RENDERING

This chapter describes the relevant components of a multiview video distribution chain and provides an overview of some of multiview video representation formats.

A. 3D Representation Formats

To fulfill the expectation of a better QoE, multiview video has become a subject of increasing interest, leading to a new set of applications. The following video formats are relevant:

- Multiview Video (MVV): requires the acquisition of N views. Since N views are captured a more immersive experience can be provided.
- Multiview Video-plus-Depth (MVD): not only the N views are acquired, but also the associated depth (Figure 1). This representation allows intermediate views to be synthesized at the receiver without being necessary to be transmitted.

Figure 1- Multiview video plus depth representation: capture of N views and corresponding depth [1].

B. *3D Video Transmission Chain*

MVD format allows to recreate the perception of depth and navigation in the scene. Figure 2 shows the transmission chain.

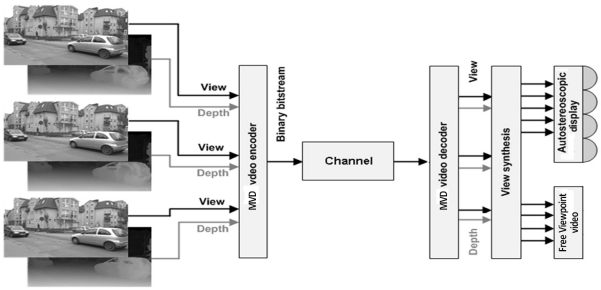

Figure 2 - Multiview video plus depth transmission chain.

The main blocks in the video framework are:

- Acquisition: Process of capturing multiple views of the 3D world scene, each one consisting of a texture and corresponding depth map of a scene.
- MVD video encoding and decoding: Compression of MVD video data using a coding standard according to the representation format. After the transmission of the data over a channel, the inverse operation (decoding) is performed.
- View synthesis: Generation of novel intermediate views based on the set of views captured and transmitted (left and right views), known as reference or lateral cameras.
- Display: The transmitted and synthesized views are shown to the user on a display.

III. Multiview plus Depth Video Objective Quality Assesment

The main point of this chapter is to present the common artifacts in synthesized images and explain the two main types of video quality assessment: subjective and objective.

A. *Artifacts Characterization in 3D Synthesized Videos*

The synthesis of novel intermediate views typically results in artifacts of the rendered views, presented in the following:

- Ghosting artifact: presence of a shadow-like artifact around contours due to the misalignment of depth and texture [2] and due to the inefficient inpainting methods (Figure 3 a)).
- Incorrect rendering of textured areas: failures in filling complex textured areas, due to inefficient inpainting methods [3] (Figure 3 b)).
- Incorrect positioning of objects: an object may have wrong dimensions or may be slightly translated/shifted due to depth map acquisition or quantization errors [3].
- Blurry regions: caused by the inpainting method used to fill occluded areas in some part of the view synthesis process. This type of artifact is typically present around the background/foreground transitions [4] (Figure 3 c)).
- Geometric distortions: distortion around object boundaries due to depth estimation errors, depth quantization errors and inaccurate camera calibration parameters [4].
- Block effect: unnatural discontinuities with a squared shape caused by texture compression.

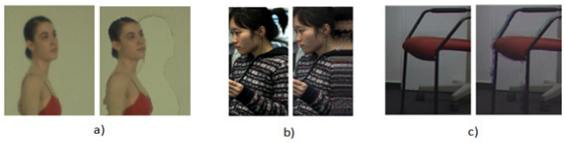

Figure 3 - a) Ghosting artifact [4]; b) Incorrect rendering of textured areas [3]; c) Blurring artifacts [3].

B. *Quality Assessment Models*

The two main types of video quality assessment, essential to guarantee an adequate QoE, are briefly described next:

1) *Subjective Quality Assessment:* Targets the evaluation of video quality by humans and is regarded as the ground truth for most objective video quality metrics. Subjective assessment is performed by conducting some tests, obtaining a numerical value of the perceived quality of the media – MOS (mean opinion score) – typically in the range 1-5. These tests are rather time expensive due to the long preparation and execution.

2) *Objective Quality Assessment:* Automatic evaluation using objective metrics to assess the video quality. The objective quality assessment metrics performance is obtained comparing MOS estimation values resulting from the metric with the MOS obtained from subjective evaluation.

IV. Subjective Quality Evaluation of Synthesized Images

This chapter presents the dataset developed to validate the metric to be proposed.

A. *IST View Synthesis Image Quality Dataset*

For the subjective quality assessment of synthesized views, two public available databases were considered: IRCCyN IVC DIBR Images database [5] and the SIAT Synthesized Video Quality Database [6].

A new synthesized image database was developed, because both IRCCyN IVC DIBR Images database and SIAT datasets do not fully fulfill our needs. On one hand, the assessed images contained in the IRCCyN IVC DIBR were synthesized using only one lateral view, and, therefore, the distortions in this database are mainly related to the hole filling strategies of different DIBR algorithms. However, the usual case is to exploit both reference views to generate the synthesized image, which allows to obtain higher quality virtual views. In addition, IRCCyN database does not include texture/depth compression distortions, which is a major drawback since compression artifacts (in both texture and depth maps) have a great impact on the quality of synthesized views. On the other hand, the objective of this work is to evaluate synthesized images, where temporal distortions are not present. In the SIAT database, the viewers evaluated videos and therefore some spatial distortions are masked by temporal aspects. Therefore, to evaluate spatial distortions, a database with synthesized images is necessary for a realistic scenario.

1) *Subjective Assessment Framework:* A subjective test assessment was conducted to obtain quality scores of synthesized images which will be used to assess the performance of a no-reference metric. Figure 4 shows how the images contained in the IST dataset were obtained.

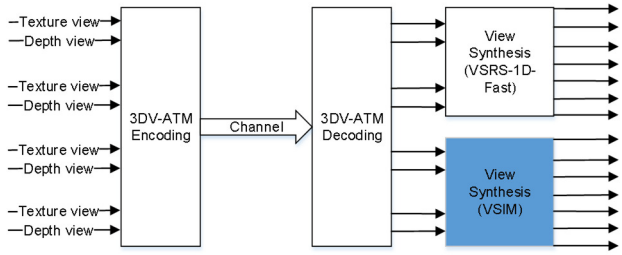

Figure 4 - Steps to obtain the images contained in the dataset

The IST View Synthesis Image Quality Dataset introduces a novel view synthesis technique to obtain features that can be used in the design of a no-reference quality assessment metric.

Table 1- IST test sequences and test conditions characterization for each sequence.

Sequence	Resolution	Input View Pair	Output View	VSRS 1D-Fast (Texture,Depth) QP Pair	VSIM (Texture,Depth) QP Pair
Book Arrival	1024x768	6-10	8	(22,26),(34,36),(42,44)	(22,26),(22,40)(34,36),(42,44)
Balloons	1024x768	1-5	3	(24,32),(32,40),(42,46)	(0,0),(24,32),(24,46),(32,40),(42,46)
Kendo	1024x768	1-5	3	(24,32),(36,38),(44,46)	(0,0),(24,32),(24,44),(36,38),(44,46)
Lovebird1	1024x768	4-6	5	(28,36),(34,44),(42,40)	(28,36),(28,50),(34,44),(42,40)
Newspaper	1024x768	2-4	3	(28,32),(38,44),(42,48)	(28,32),(28,50),(38,44),(42,48)
Dancer	1920x1088	1-9	5	(24,20),(32,28),(44,35)	(24,20),(24,40),(32,28),(44,35)
PoznanHall2	1920x1088	5-7	6	(24,28),(34,36),(40,42)	(24,28),(24,46),(34,36),(40,42)
PoznanStreet	1920x1088	3-5	4	(22,28),(30,44),(42,35)	(22,28),(22,44),(30,44),(42,35)
GT Fly	1920x1088	1-9	5	(24,28),(34,38),(44,48)	(24,28),(24,44),(34,38),(44,48)
Shark	1920x1088	1-9	5	(24,28),(36,40),(42,48)	(24,28),(24,44),(36,40),(42,48)
Number of synthesized images	-	-	-	2x3x10=60	2x4x10+2x20=120

The lateral views contained in the SIAT database were used to synthesize novel intermediate views using View Synthesis with Inverse Mapping (VSIM) algorithm [7].These synthesized views, as well as some views synthesized by the VSRS-1D-Fast software [8], available in SIAT database, were included in the subjective assessment. The assessment method used in the subjective test is the same as SIAT database: Absolute category rating with hidden reference (ACR-HR). ACR-HR is a single-stimulus method, where only a single image is presented at a time. Without informing the subjects, the test procedure also includes an original version of each synthesized image, shown as any other test stimulus. The quality scale used is a five-grade scale: Excellent, Good, Fair, Poor and Bad.

2) *Test material:* The IST View Synthesis Image Quality Dataset contains 180 synthesized images where rendering and compression artifacts are present and need to be evaluated. The information about the synthesized views can be found in *Table 1*. For each video sequence, synthesis algorithm and QP pair, 2 synthesized images for different time instants were included in the dataset, i.e. an image with few rendering artifacts and other image with more severe ones. The only exception is for QP pair (0,0) where 20 synthesized images were included for different time instants. Since the SIAT database does not provide original texture and depth videos of the lateral sequences, they were obtained from [9] for two video sequences: Balloons and Kendo. The synthesized images from lateral views with low texture QP and high depth QP were introduced in the dataset to evaluate the impact of depth map compression in the synthesized views, while maintaining texture QP at a minimum. These QP combinations were selected to obtain a varied set of artifacts but also subjective scores across all the MOS scale.

3) Analysis of the subjective quality assessment scores: The analysis of the subjective quality scores is performed. Figure 5 shows the computed DMOS scores, for each of the 180 synthesized images contained in the developed dataset. The DMOS was obtained by subtracting the score of the distorted image from the score of the corresponding original version.

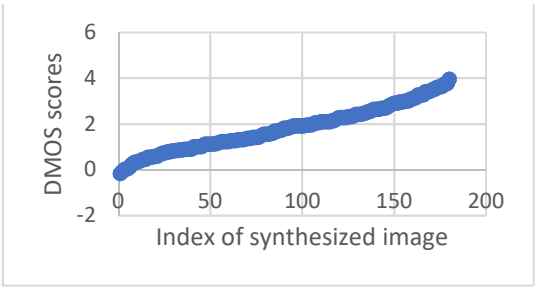

Figure 5 - Computed DMOS scores for each synthesized image.

There are two synthesized images with negative difference scores, i.e., the score of the original image is less than the score of the distorted image. These negative DMOS values are a common occurrence in this type of subjective assessment methodology. These negative scores are kept to ensure the original diversity of the subjective data. The database was designed to sample a range of visual quality in an approximately uniform fashion. Figure 5 shows that the database also exhibits reasonably uniform distribution of scores along the DMOS axis and therefore the quality of the synthesized sequences spans a wide range, from excellent to bad.

V. NO-REFERENCE QUALITY METRIC FOR SYNTHESIZED IMAGES USING FEATURE FUSION

This chapter describes the solution designed and implemented to evaluate synthesized images. The proposed metric relies on the extraction of several features. Then, a mapping function is required to map these features into a quality score.

A. No-reference Image Quality Metric Architecture

In image quality assessment, there are two important parts: extraction of appropriate features and the pooling of the features. For feature pooling, a Support Vector Regression (SVR) was selected since it was already applied in similar image quality assessment problems. SVR is the application of Support Vector Machine (SVM) for solving regression problems. In image/video quality assessment, images are represented as a vector of visual features, and the label is the quality score of that image. The SVM implementation selected is the SVM^{light} package [10], implemented in C, that is able to solve the regression problem. Figure 6 represents the training step of the SVR.

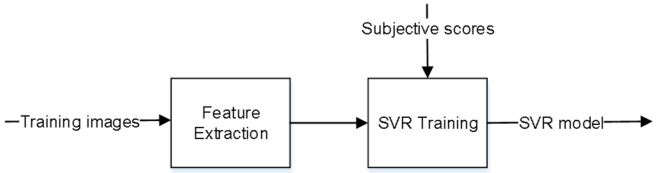

Figure 6 - Learning stage of the SVR.

The modules of the learning phase of the quality metric are described in the following:

- Feature extraction: Several features are extracted from a subset of images selected for training the algorithm. The extracted features are fed into the SVR.
- SVR Training: Builds a model based on the input data: the features and quality scores as DMOS values. DMOS values are considered the target ground truth quality values, obtained from the subjective assessment. The goal of the SVR is to find a

regression function, such that it accurately predicts the outputs corresponding to a new set of input features.
After training the SVR, the next step is to test it against new data samples. Figure 7 represents the testing phase of the quality metric.

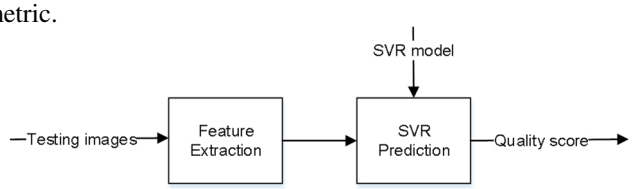

Figure 7 - Testing stage of the SVR.

The modules of this step are briefly described in the following:
• Feature Extraction: The same features extracted in the learning step are now extracted for the new unseen images and fed to the SVR prediction module.
• SVR Prediction: The prediction module starts by reading the SVR model built during the training phase. Using the regression function contained in the SVR model, the features extracted from the test image can be mapped to quality score. The output of the SVR prediction module is the DMOS predictions of the images used for test.

B. No-reference Features Extraction

A no-reference metric has to predict the perceived image quality without having access to the corresponding original version. Thus, the features to be extracted must be obtained from information available at the decoder, and should characterize well the possible sources of degradation. The considered features can be divided into three categories:
• Class A - Independent of the synthesized view and of the synthesis method: Measures the quality of the lateral texture views and corresponding depth maps from which the synthesized view was rendered. These features do not use any information about the synthesized view and can be extracted from the bitstream or from side information available in a separate logical channel.
• Class B – Extracted during the synthesis process: These features are extracted during the synthesis of a view and are based on intermediate data used to synthesize the view under analysis. These features are extracted when views are synthesized with VSIM algorithm.
• Class C -Use the final synthesized view: The final synthesized view, which is shown to the viewer, is used directly to measure its quality. The features are independent of the algorithm of the rendering method and thus in some way are more general.
All these three types of features can be used in different application scenarios, e.g. in some scenarios it is not possible to get access to the synthesized view or the algorithm used to generate it and have different strengths and weakness, e.g. it is possible to identify accurately the occluded areas in Type B features.

1) Features independent of synthesized view and synthesis method

The extracted features are the following:
• **Texture QP (A1):** The quantization parameter (QP) of the lateral texture views can be used to measure compression artifacts contained in the synthesized view.
• **Depth QP (A2):** Depth map accuracy significantly influences the quality of the synthesized view. Therefore, lossy depth map compression may lead to artifacts such as incorrect positioning of objects, or objects with wrong dimensions. The QP of the lateral depth maps can be used to measure some artifacts of the synthesized view.
• **SSIM of the lateral views (A3):** The full-reference structural similarity (SSIM) is computed to measure the quality of the decoded lateral views texture. For each lateral texture view that can be used to synthesize the novel view, an SSIM value is determined. The mean of these SSIM values is used as feature.

The texture and depth quantization parameters of the lateral views can be sent in the bitstream [11] with the corresponding view, while any full-reference quality metric of these views can be made available in a separate logical channel. For example, when adaptive streaming is used (e.g. DASH) it is possible to characterize the quality of the images or video segments inside a media presentation description that is transmitted to the decoder (by an HTTP response) to inform which representations are available.

2) Features extracted during the synthesis process

The features extracted during the virtual view synthesis are:
• **Multiple Correspondences (B1):** This feature is the percentage of pixels of the virtual view with more than 1 pixel projected from a single lateral view and is computed to measure the quality of the lateral depth maps. Due to depth map estimation errors, multiple pixels of the lateral view can be projected to the same position on the virtual view.
• **Projected Views MSE (B2):** Since there are two lateral views that are projected on the same virtual view, the difference between them may give an insight of the quality of the synthesized view - in fact, it is expected that the more accurate the projections are, the lower will be the difference between both. As difference metric, the mean squared error (MSE) computed between the two projected views, on high gradient magnitude regions (which are the most perceptually relevant regions), was selected. Figure 8 shows the architecture to compute the Projected Views MSE feature.

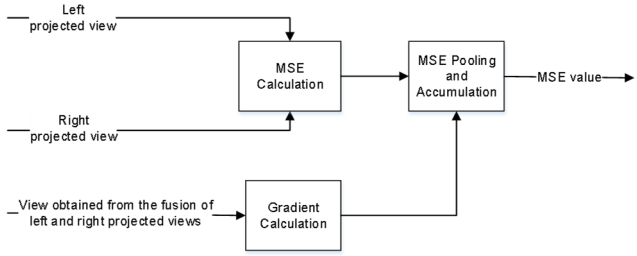

Figure 8 - Architecture of the Projected Views MSE (B2).

The following steps describe how to compute the B2 feature value:

a) MSE Calculation: For each pixel of the virtual view, the MSE between the left and right projected pixels is computed. The MSE of pixels of the virtual view without projected pixels from one or both lateral views are set to 0 but excluded from the MSE pooling and accumulation step.

b) Gradient Calculation: The gradient magnitude (Prewitt operator [12]) of the view obtained from the fusion of the projected lateral views, is computed, resulting in the gradient image G_1. To keep only the higher gradient magnitudes, where the distortions are more common and visible, the gradient magnitudes lower than 40 are set to 0, resulting in the gradient image, G_2. The gradient image G_2 is then binarized, converting magnitudes higher than 40 to 1, and the rest to 0, resulting in a binary image, B_1. To also take into account the neighboring regions of the high gradient regions, the edges of B_1 are dilated

using the morphological operator dilate [13], with a 7×7 structuring element, resulting in the binary image B_2.

c)MSE Pooling and Accumulation: By intersecting the outputs of the two previous steps, only the MSE values for the high magnitude gradient regions are kept; these values are then averaged.

• **Holes Percentage (B3):** Holes are pixels of the virtual view without corresponding pixels positions on the lateral views. The final synthesized view is obtained after inpainting these holes. Naturally, the amount of hole pixels influences the performance of the inpainting method and, consequently, the quality of the final synthesized view. This feature is the percentage of hole pixels, i.e. pixels that need to be interpolated (inpainted) after merging the projected lateral views. Figure 9 shows the architecture to compute the Holes Percentage feature.

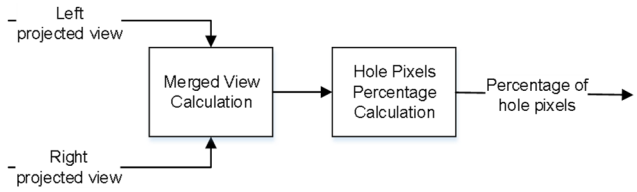

Figure 9 - Architecture of the Holes Percentage (B3).

The steps to compute this feature are the following:

a)Merged View Calculation: The merged view is the view obtained from merging the projections of the lateral views. After merging the projected left and right views, there are still some remaining holes.

b)Hole Pixels Percentage Calculation: This step calculates the number of hole pixels in the merged view and the respective percentage relatively to the total number of pixels in the image

• **Projected Views Hausdorff Distance (B4):** Edge areas are regions of the synthesized view where artifacts are more likely to occur (due to occlusions). In addition, artifacts located on edges have a higher impact in the human perception than in smooth regions. As already mentioned, the projection accuracy depends on the depth map quality, on the accuracy of the method used for projection, and on the camera calibration accuracy. In order to measure all these potential sources of distortion, the left lateral texture view is projected on, and compared with, the right lateral view. The comparison metric used is the Hausdorff distance [14] that measures the distance between corresponding edges of both views. Figure 10 presents the architecture to compute the Projected Views Hausdorff Distance feature.

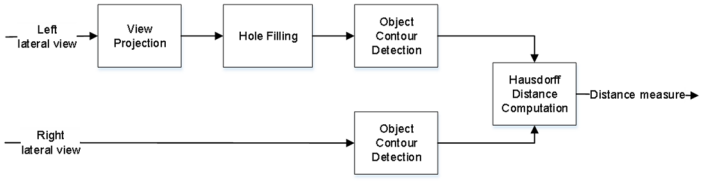

Figure 10 - Architecture of the Projected Views Hausdorff Distance (B4).

The modules to compute the feature are explained in the following:

a)View Projection: The left lateral view is projected to the right lateral view position using:

$$x_R = x_L - \frac{f\ b}{Z_L(x_L, y_L)} \quad (1)$$

where x_R is the horizontal position in the right view of the pixels projected from the left view, (x_L, y_L) represents the coordinates of a given pixel in the left lateral view, f is the focal length, b is the baseline (distance between cameras) and Z_L is the depth in z-distance format [15] of the left view. Since the cameras are parallel, the vertical position of the pixels does not change.

b)Hole Filling: The projected view contains holes due to disocclusions and depth estimation/quantization errors; these holes are filled with the pixels of the right view.

c)Object Contour Detection: Edges pixels are first detected on both views (projected and right). Edges on textured areas are then filtered out, since the human visual system (HVS) is much more sensitive to edges corresponding to objects contours. The steps to detect the contour edges are the following:

i. Calculation of the gradient magnitude of the image using the Prewitt operator [12], resulting in gradient image G_1.
ii. Selection of the higher gradient magnitude pixels from G_1, by setting to 0 gradient magnitudes lower than 40, resulting in the gradient image G_2.
iii. Textured regions are usually composed by groups with small number of edge pixels. In order to filter it out, G_2 is first binarized by setting to 1 gradient magnitudes different from 0, resulting in a binary image B_1. Connected edge pixels regions with less than 128 pixels are then removed from B_1, using the morphological operator area opening [16]. This operator finds the connected components of the binary image (groups of pixels), compute its area and remove the group if its area is lower than an input threshold. The output of this step is a binary image, B_2.
iv. Textured areas are also characterized by edges close to each other. These edges can be eliminated using an operator to fill the small gaps between them, such as the morphological closing operator [16]. The output of this step is a binary image, B_3.
v. Compute the gradient magnitude of the closed image B_3 and binarize it, obtaining a binary image B_4. B_4 basically contains the outer contours of the closed textured regions of B_3.
vi. Multiply pixel by pixel the binary image B_1 by the binary image B_4, resulting in the binary image B_5. B_1 contains all the edges from textured regions, where edge pixels are barely connected. B_4 contains the exterior edges of small textured regions, with groups of pixels well connected, that should be eliminated as well. Multiplying these images, results in an image where the exterior edges of textured regions are defined by small groups of pixels (barely connected, which can be filtered out afterwards.
vii. Filter out the group of pixels with less than 16 pixels from binary image B_5.

d)Hausdorff Distance Computation: Compute the distance between the edges (object contours) of the right view and the edges of the projected left view, using an Hausdorff based distance [14]. The steps are the following:

i. For each edge pixel a of the projected left view in the right view (after the holes have been filled), the distance from that pixel to the set of edge pixels of the right view, B, is computed using $d(a, B) = min_{b \in B} \|a - b\|$, where b is an edge pixel belonging to B. This is performed symmetrically, i.e. the distance from each edge pixel of the right view to the set of edge pixels of projected left view is also computed. The output are two matrices of distances with the same size as the views.
ii. Computing the Hausdorff distance for the whole image would make regions with more errors to be faded by regions with fewer errors. Therefore, both distance matrices are divided into blocks of Nx N, where N=32, and

for each distance matrix, the directed block distance is calculated with:

$$d_d(A_n, B_n) = M(d(a, B)_{a \in A_n}) \quad (2)$$

where d_d is denoted as directed block distance, n is the block index, A_n and B_n is the set of edge pixels on block n and the operation M represents the median.

iii. Since $d_d(A_n, B_n)$ may be different from $d_d(B_n, A_n)$, these distances are combined using, for each block:

$$d_u\big(d_d(A_n, B_n), d_d(B_n, A_n)\big) = \min(d_d(A_n, B_n), d_d(B_n, A_n)) \quad (3)$$

where d_u is denoted as undirected distance. The output is a matrix of size $\frac{h}{N} \times \frac{w}{N}$, where h and w is the height and width of the views being compared. Each element of the matrix contains an undirected distance measure between the edges of both views.

iv. Exclude blocks whose undirected distance is equal to 0 (no edges), to avoid non-edge regions to dominate and thus, attenuate the importance of the distorted edge regions.

v. The final feature is the mean of the remaining undirected distances associated to each block.

• **Inter Lateral Views SSIM (B5):** The SSIM between the right view and the left view projected in the right view position is computed, since this may reflect depth map inaccuracies, errors from the synthesis process and quantization errors due to the compression of lateral views. Figure 11 shows the architecture to compute the Inter Lateral Views SSIM feature.

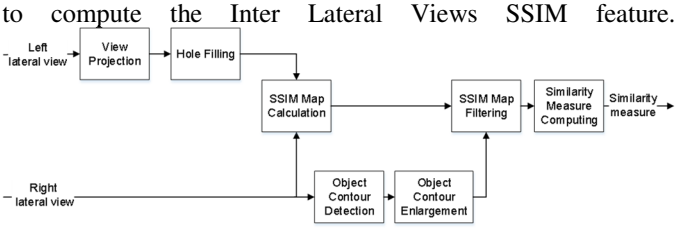

Figure 11- Architecture of the Inter Lateral Views SSIM (B5).

The steps to compute the feature are described next:

a)View Projection: The left lateral view is projected on, and compared with, the right lateral view. Projecting a lateral view causes the same type of artifacts that are present in the final synthesized view; however, the other lateral view is free of rendering artifacts and, therefore, can play the role of the "original" frame.

b)Hole Filling: Small holes of the projected view are filled using a median filter with 3x3 size. The remaining holes are filled with the pixels of the right view.

c)SSIM Map Calculation: The SSIM map between the projected left view and the right view is computed. The output is a matrix where each element represents the local SSIM value.

d)Object Contour Detection: Since the HVS is much more sensitive to errors on edges, the objects contours of the right lateral view are detected using the same procedure described in feature B4.

e)Object Contour Enlargement: The objects contours are enlarged (dilated) using the morphological operator dilate [13], with a 7×7 structuring element, in order to keep not only the SSIM values coincident (spatially) with regions of high gradient magnitude but also in their neighboring regions.

f)SSIM Map Filtering: In the SSIM map, the values that coincide (spatially) with the enlarged edges from the previous step are kept, while the others are removed. The SSIM values of the map that coincide (spatially) with hole positions of the projected left view (before hole filling) are removed.

g)Similarity Measure Computing: The similarity measure is the mean of the filtered SSIM values.

• **Inter Lateral Views Phase Congruency Similarity (B6):** Phase Congruency (PC) is a frequency based measure that assumes that locations where the phase of the Fourier components is maximal, represent important perceptual features in the image (such as edges, lines and corners) [17]. Feature B6 computes the PC similarity between the right lateral view and the left projected view. The phase congruency similarity metric used by this feature is the same as in the FSIM metric [18]. FSIM is a widely known metric to assess the similarity between two images [18]. Figure 12 shows the architecture that is used to extract the Inter Lateral Views Phase Congruency Similarity feature.

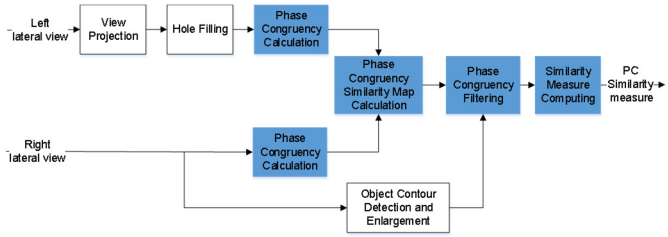

Figure 12- Architecture of the Inter Lateral Views Phase Congruency Similarity (B6).

The modules View Projection, Hole Filling and Object Contour Detection and Enlargement are computed as in feature B5. Therefore, to avoid repetition, only the novel modules (represented in blue) for this feature are explained in the following:

a)Phase Congruency Calculation: The phase congruency maps, PC_1 and PC_2, are computed for the right view and for the projected left view after holes filling.

b)Phase Congruency Similarity Map Calculation: The similarity for the phase congruency maps is computed by:

$$S_{PC}(x) = \frac{2PC_1(x)\, PC_2(x) + T_1}{PC_1^2(x) + PC_2^2(x) + T_1} \quad (4)$$

where T_1 is a positive constant to increase the stability of $S_{PC}(x)$. The output is the phase congruency similarity map, S_{PC}, which may take values in the interval [0,1].

c)Phase Congruency Filtering: In the phase congruency similarity map, only the values that are spatially coincident with the dilated edges obtained in the previous step are kept. S_{PC} values spatially coincident with holes positions of the projected left view (before hole filling) are removed.

d)Similarity Measure Computing: The feature Inter Lateral Views Phase Congruency Similarity corresponds to the average of the remaining S_{PC} values.

• **Inter Lateral Views Gradient Similarity (B7):** The image gradient conveys important visual information, because it can capture both contrast and structure information. Since the local contrast has a perceptual impact on the image quality, the image gradient magnitude is used in this feature. The similarity between the gradient maps of the right lateral view and the left view projected in the right view position is computed. To compare the two gradient maps, the FSIM metric [18] gradient map similarity expression is used. The architecture of the Inter Lateral Views Gradient Similarity feature is presented in Figure 13.

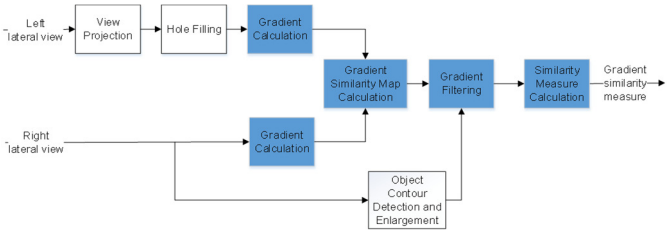

Figure 13 - Architecture of the Inter Lateral Views Gradient Similarity (B7).

The modules View Projection, Hole Filling and Object Contour Detection and Enlargement are computed as in feature B5. Therefore, to avoid repetition, only the novel modules (represented in blue) for this feature are described:

a)Gradient Calculation: The gradient maps are calculated for the projected left view and right view, resulting in G_1 and G_2.

b)Gradient Similarity Map Calculation: The similarity for the gradient maps is computed with:

$$S_G(x) = \frac{2G_1(x)\,G_2(x) + T_2}{G_1^2(x) + G_2^2(x) + T_2} \quad (5)$$

where T_2 is a positive constant. The output is a gradient similarity map, S_G, which may take values in the interval [0,1].

c)Gradient Filtering: In the gradient similarity map, the values that are spatially coincident with the dilated edges obtained in the Object Contour Detection and Enlargement step are kept. Gradient similarity values spatially coincident with holes positions of the projected left view (before hole filling) are removed.

d)Similarity Measure Calculation: The Inter Lateral Views Gradient Similarity feature is given by the mean of the filtered S_G values for the entire frame.

3) Features which use the synthesized view

In a no-reference scenario, a viable option to estimate the quality of the synthesized views is to extract directly features of the synthesized view but also exploiting the availability of the lateral views used for rendering. These features are associated to the rendering technique and are described next.

• **DSQM Metric (C1):** DIBR-Synthesized Image Quality Metric (DSQM) is a recently proposed metric [17] to assess the quality of synthesized images and can be used directly as a feature. The algorithm uses the lateral views from which the virtual image is synthesized to estimate the distortion induced by the DIBR process. In particular, block-based perceptual feature matching based on signal phase congruency is used to estimate the synthesis distortion. As stated in feature B6, phase congruency is a frequency based measure which detects where the Fourier components are maximal in phase [17]. The architecture of the DSQM metric is presented in Figure 14.

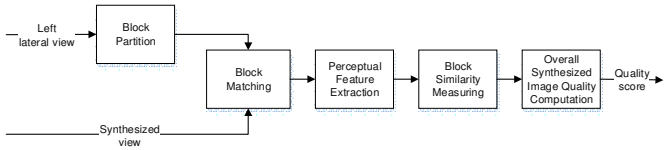

Figure 14 - Architecture of the DSQM Metric (C1).

A more detailed description can be found in [17]. The main blocks of the architecture are explained next:

a)Block Partition: This feature is computed at the block level and thus, the left lateral view is partitioned into blocks, which are of size 64x64 in this case.

b)Block Matching: For each block of the left view, the best match in the synthesized view is found using a typical block-matching approach; this can be understood as disparity estimation between views. Figure 15 shows a lateral view and its synthesized view. The figure also shows a block p and its corresponding matching block, q, in red color. The green color rectangle shows the search window for block p in the synthesized image. There is no vertical displacement because the cameras are in a 1D parallel setup. The output of the module are the matching block pairs.

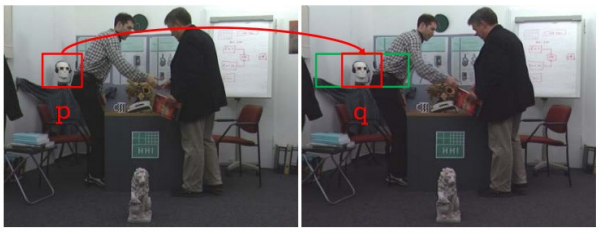

Figure 15 - Block p of the lateral view (left) and corresponding matching block, q, in the synthesized view, within the search area, in green.

c)Perceptual Feature Extraction: The phase congruency is computed for each block of the matching block pairs, resulting in PC_p and PC_q. The phase congruency is a pixel based measure, and, therefore, the pooling is given by the mean of the phase congruency values of PC_p and PC_q block maps, $\mu(PC_p)$ and $\mu(PC_q)$.

d)Block Similarity Measuring: The absolute difference, Q, between the mean of the phase congruency maps of the two corresponding blocks is computed to estimate the synthesis distortion due to DIBR. The Q value is obtained with:

$$Q = \left|\mu(PC_p) - \mu(PC_q)\right| \quad (6)$$

e)Overall Synthesized Image Quality Computation: The overall quality of the synthesized image is computed by averaging the quality scores, Q, computed in the previous step. The final DSQM metric is calculated by:

$$DSQM = \frac{1}{K}\sum_{i=1}^{K} Q_i \quad (7)$$

where K is the number of the total matching block pairs and Q_i is the quality of the i-th matching pair. DSQM is a complete metric and smaller DSQM values represent better image quality, because a lower Q_i means a higher similarity.

• **Synthesized View Phase Congruency (C2):** Instead of computing the entire FSIM metric [18], whose final similarity value results from combining the phase congruency and gradient similarities, the similarities can be evaluated separately. The phase congruency similarity, implemented in FSIM, has shown to be an adequate feature by itself, and therefore is computed between the left lateral view after disparity compensation and the synthesized view. Since the original version of the synthesized view is not available, the left lateral view (after disparity compensation) can be used as a reference since it does not contain rendering artifacts. The architecture of the Synthesized View Phase Congruency is presented in Figure 16.

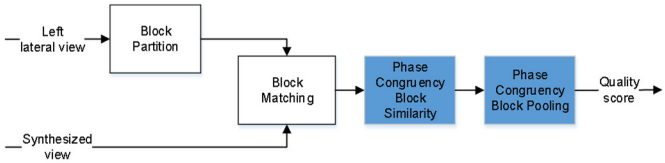

Figure 16 - Architecture of the Synthesized View Phase Congruency (C2).

The block partition and the image block matching modules are implemented as described in feature C1. Therefore, only the remaining modules of the architecture in Figure 16, represented in blue, are explained next:

a)Phase Congruency Block Similarity: For each block of a matched block pair, the phase congruency map is computed. The phase congruency is given by the ratio between the energy and the sum of the Fourier amplitudes [18]. Then, the phase congruency similarity map is computed using equation (4). The PC block similarity value is given by the mean of the phase congruency similarity map of each block. The phase congruency similarity is computed at block level, because computing it at image level would include block discontinuities resulting from the disparity compensation.

b)Phase Congruency Block Pooling: The final feature corresponds to the average of the PC block similarity measures.

• **Synthesized View Hausdorff Distance (C3):** The Hausdorff distance between the edges of the synthesized view (after disparity compensation) and the edges of the left lateral view is computed and used as quality metric of the synthesized view. The architecture of the Synthesized View Hausdorff Distance feature is shown in Figure 17.

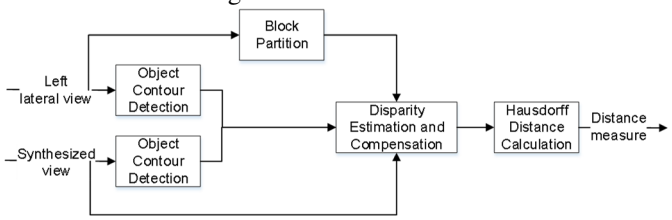

Figure 17 - Architecture of the Synthesized View Hausdorff Distance (C3).

The synthesized view and the left lateral view represent the same visual scene, although, from different perspectives. Therefore, to measure the distance between its edges, it is necessary to estimate and compensate the disparity between these two views. The modules to compute the feature are described in the following:

a)Object Contour Detection: The contours of the objects of the left and synthesized views are computed using the procedure described in feature B4.

b)Block Partition: The left lateral view is partitioned into blocks of size 64x64.

c)Disparity Estimation and Compensation: The block matching between the left lateral view and the synthesized view is computed, as in feature C1. The computed displacements are used to compensate the shifts of the synthesized image contours.

d)Hausdorff Distance Calculation: Computes the distance between the edges (object contours) of the compensated synthesized view and the edges (object contours) of the left lateral view. The steps followed are the same as described in feature B4, although the directed distance for each block is computed according to [14] using (8), because it was the directed distance which provided higher correlation between the feature value and the subjective scores.

$$d_d(A,B) = min_{a \in A} d(a,B) \tag{8}$$

VI. Experimental Results

A. Test Conditions

As stated in Chapter 5, the features extracted to evaluate the synthesized views can be divided into three classes: A, B and C. Depending on the available features, the proposed metric can be used in different scenarios, namely:

• Scenario A: The quality of the synthesized view is evaluated using only features from class (A). For instance, in some streaming applications (e.g. to decide which views must be transmitted) it is important to assess the quality of the synthesized view using only information about the quality of the lateral views, such as the QP of these views.

• Scenario C: The synthesized view is evaluated using only features from class (C), e.g., in a scenario where there is not information about the quality of the lateral views neither from the synthesis process. In this scenario, only the synthesized view is available to evaluate its quality. Also, the metric does not depend on the synthesis process or the visual information received and can be used whenever there is no access to decoding data or the data being processed in the synthesis algorithm.

• Scenario AC: The synthesized view is evaluated using features of classes (A) and (C), e.g., in a scenario where only the lateral and synthesized views are available, and it is not possible to have access to the synthesis process. This may occur often, since usually the quality metrics does not have access to information besides visual data shown or used for rendering.

• Scenario ABC: Besides features from groups A and C, features from group B, which are extracted during the synthesis process, can be used. Features of all classes can be computed when there is access to all data used and produced by the synthesis method as well as the synthesized image.

The developed IST View Synthesis Image Quality Dataset, was used to assess the performance of the proposed no-reference quality metric; this database comprises two datasets – full dataset and partial dataset – according to Table 2.

Table 2 - *Database division according to the synthesis algorithm.*

	Synthesis algorithm	Number of images	
Full dataset	VSRS-1D-Fast	60	
	VSIM	120	Partial dataset

The reasons behind the database division were:

• Full dataset: Contains all the images synthesized with VSRS-1D-Fast and with VSIM. Since the images resulting from the VSRS-1D-Fast were not synthesized in this work, but extracted from the SIAT database, the respective Class B features cannot be computed. Therefore, to evaluate the metric on the full dataset, only Class A and Class C features can be used.

• Partial dataset: Contains only the images synthesized by the VSIM algorithm. Since all these images were synthesized in this work, in this partial dataset all features classes can be used.

B. Support Vector Regression

In this section, some steps to improve the SVR performance are described; the strategy to select the training and the testing sets is also presented.

After the extraction of the appropriate features, a step that helps improving the SVR performance is the normalization of all the features. The normalization of each feature was done by subtracting its mean value, μ_j, and by dividing by its standard deviation, σ_j :

$$x^i_{j,normalized} = \frac{x^i_j - \mu_j}{\sigma_j} \tag{9}$$

where i stands for the image number, j stands for the feature number, and x is the feature vector. The mean and standard deviation are computed in the feature vector of the training set. Another step to improve the SVR performance is the appropriate setting of three training parameters:

- $\boldsymbol{C}$: is the penalty factor or cost.
- $\boldsymbol{\varepsilon}$: is the SVR algorithm's tolerance for errors.
- Kernel: Four kernel functions used in SVR are linear kernel, polynomial, radial basis function (RBF) and sigmoid [19].

After feature normalization, SVR kernel and parameters selection, the training and testing stages of the SVR take place, using the Cross-Validation (CV) procedure. This procedure consists in dividing the dataset into subsets, then train the learning model on some subsets (training sets), and test the model on the remaining subsets (testing sets). In this work, the followed approach is similar to a K-fold strategy. The dataset of synthesized images is split into 10 folds, each fold containing all the images corresponding to the same MVD sequence: Book Arrival, Balloons, Kendo, Lovebird1, Newspaper, Dancer, PoznanHall2, PoznanStreet, GT Fly and Shark. From the 10 folders, 9 are used as training data and the remaining one is used for testing the model. After repeating the process 10 times, with each folder used exactly once as the validation data, the correlation between the subjective scores (DMOS) and the corresponding predicted scores, is calculated.

C. Feature Quality Evaluation

In this section, the quality of the features proposed and described in Chapter 5, is evaluated. To evaluate the quality of a feature, the Pearson linear correlation coefficient (PLCC) between the feature values and the DMOS values, obtained from the subjective assessment, is calculated. Table 3 shows the correlation results between each feature and DMOS scores, for both full and partial datasets.

Table 3 - *PLCC coefficient between the feature values and DMOS.*

Category	**Feature**	**PLCC** Full Dataset	**PLCC** Partial Dataset
Independent of synthesized view and any synthesis method	A1	0.681	0.749
	A2	0.647	0.723
	A3	-0.670	-0.689
Extracted during the synthesis process	B1	-	0.449
	B2	-	0.629
	B3	-	0.478
	B4	-	0.611
	B5	-	-0.546
	B6	-	-0.610
	B7	-	-0.527
Use the synthesized view	C1	0.647	0.685
	C2	-0.215	-0.471
	C3	0.274	0.371

D. Feature and Kernel Function Selection

Since it is not possible to know in advance which feature combinations are the best, and it would be very time consuming to compute the PLCC between all subset of features combined to the possible Kernel functions and SVR parameters, C and ε, the procedure adopted to select the features and Kernel is the following:

- Test several combinations of features, using the linear kernel, with default parameters C, ε of SVM^{light}.
- Test the combinations of features used in the previous step with the RBF Kernel, with default parameters C , ε and γ.
- Select the combination of features and Kernel function that maximizes the PLCC between the DMOS and predicted DMOS.

After studying the SVR prediction performance for the different scenarios, with different subset of features and different Kernel functions, the combination of features and Kernel that provided the best results were found, and are presented in Table 4.

Table 4 - *Best training features and Kernel found for each scenario, Full Dataset.*

Scenario	Features	Kernel	PLCC Full Dataset	PLCC Partial Dataset
A	A2,A3	RBF	0.788	0.805
C	C1,C3	Linear	0.632	0.659
AC	A1,A3,C1,C3	Linear	0.810	0.837
ABC	**A1,A3,B1,C1**	**Linear**	**-**	**0.871**

The SVR parameters were then adjusted for the scneario ABC, reaching a PLCC equal to 0.876.

E. Final No-reference Metric Performance

The evaluation of the proposed solution was performed on the developed IST View Synthesis Image Quality Dataset using, as assessment measures, the PLCC, the Spearman rank-order correlation coefficient (SROCC), the root-mean-square error (RMSE) and the mean absolute error (MAE). A good metric is expected to have high PLCC and SROCC values, and low RMSE and MAE values. The comparison was done assuming all the features can be extracted (scenario ABC), where the SVR training and testing is performed with parameters C=0.25, ε=0.30. The following conventional 2D-IQA metrics were used as benchmark techniques: MSE, PSNR, SSIM [20], MS-SSIM [21] and VSNR [22]. As full-reference metrics, the original versions of the synthesized views under evaluation have to be available. The values resulting from the 2D-IQA metrics were mapped to the subjective scores using the following logistic function, outlined in [17]:

$$DMOS_p = \beta_1 \left(\frac{1}{2} - \frac{1}{1 + \exp(\beta_2 (S - \beta_3))} \right) + \beta_4 S + \beta_5 \qquad (10)$$

where S is the metric value, $DMOS_p$ is the mapped subjective score and $\beta_1, \ldots, \beta_1$ are the regression model parameters, obtained through a regression step which tries to minimize the error between the DMOS and $DMOS_p$.

The proposed metric is also compared with two objective quality metrics for synthesized images, whose source code is available online: the no-reference metric DSQM [17], described on the previous chapter, and the full-reference metric 3DSwIM [23]. The distortion values computed by the DSQM metric were mapped to subjective ratings using the same logistic function of [17], which is formulated by equation (10).The distortion values resulting from the 3DSwIM were mapped to the DMOS scores using the polynomial function formulated by (11):

$$\mathrm{DMOS_p} = a \cdot score^3 + b \cdot score^2 + c \cdot score + d \qquad (11)$$

where $score$ is the distortion value obtained from the metric and a, b, c and d are the parameters of the cubic function, obtained through a regression step to minimize the difference between the true DMOS values and the predicted DMOS.

The PLCC,SROCC, RMSE and MAE values are presented in Table 5. The proposed no-reference metric is superior to the other full-reference 2D and 3D IQA metrics, with higher PLCC and SROCC and lower RMSE and MAE.

Table 5 - *PLCC between true DMOS and predicted DMOS, for 2D-IQA metrics and proposed metric.*

Metric	PLCC	SROCC	RMSE	MAE
MSE	0.6869	0.7632	0.6960	0.5424
PSNR	0.7612	0.7632	0.6201	0.4647
SSIM	0.7846	0.7787	0.5929	0.4572
MS-SSIM	0.8547	0.8426	0.4964	0.3872
VSNR	0.6150	0.5699	0.7541	0.6229
DSQM	0.6975	0.7021	0.6853	0.5426
3DSwIM	0.6767	0.6236	0.7041	0.5597
Proposed	**0.8762**	**0.8724**	**0.4618**	**0.3637**

VII. CONCLUSIONS

A no-reference quality metric for synthesized images was proposed. Using a machine learning tool, it combines features that can be obtained before, during or after the synthesis procedure and, accordingly, it can be used in different application scenarios. The proposed metric outperforms the state-of-the-art metrics, being able to predict the images subjective scores with a PLCC close to 0.9.A new dataset was developed with synthesized images containing artifacts due to the texture and depth compression of the source views, as well as artifacts due to errors (or imperfections) in the synthesis process. It will be made available to the scientific community.

VIII. REFERENCES

[1] F. Dufaux, B. Pesquet-Popescu and M. Cagnazzo, Emerging Technologies for 3D Video, Jhon Wiley & Sons,Ltd, 2013.

[2] S. Muddala, M. Sjöström, R. Olsson , "Virtual view synthesis using layered depth image generation and depth-based inpainting for filling disocclusions and translucent disocclusions",Journal Visual Communication Image Representation.38 (2016) 351–366, 2016.

[3] F. Battisti, E. Bosc, M. Carli, P. Callet and S. Perugia, "Objective image quality assessment of 3D synthesized views", Signal Processing: Image Communication, 2014.

[4] S. Muddala, "Free View Rendering for 3D Video", Edge-Aided Rendering and Depth-Based", Doctoral Thesis No. 226 Sundsvall, Sweden 2015.

[5] E. Bosc, R. Pépion, P. Callet, Martin Köppel, P. Ndjiki-Nya, M. Pressigout, and L. Morin, "Towards a New Quality Metric for 3-D Synthesized View Assessment", IEEE Journal of Selected Topics in Signal Processing, vol. 5, no. 7, November 2011.

[6] X. Liu, Y. Zhang, S. Hu, S. Kwong, C.Kuo, and Q. Peng, "Subjective and Objective Video Quality Assessment of 3D Synthesized Views With Texture/Depth Compression Distortion", IEEE Transaction on Image Processing, vol. 24, no. 12, December 2015.

[7] M. Farid, M. Lucenteforte, M. Grangetto, "Depth Image Based Rendering with Inverse Mapping", in Proc. Multimedia Signal Processing (MMSP), 2013 IEEE 15th International Workshop on, pp. 406–411, Sep. 2013.

[8] International Organisation For Standardization, "Test Model under Consideration for HEVC based 3D video coding," ISO/IEC JTC1/SC29/WG11 MPEG2011/N12559, February 2012.

[9] "Fujii Lab's sequences Download lists" , [Online]. Available: http://www.fujii.nuee.nagoya-u.ac.jp/multiview-data/ , [Accessed September 2017].

[10] "SVM-Light Support Vector Machine" , [Online]. Available: http://svmlight.joachims.org/. [Accessed September 2017].

[11] ISO/IEC 23001-10:2015 "Information technology – MPEG systems technologies - Part 10: Carriage of Timed Metadata Metrics of Media in ISO Base Media File Format".

[12] G. Chaple, R.Daruwala and M. Gofane, "Comparisions of Robert, Prewitt, Sobel operator based edge detection methods for real time uses on FPGA", International Conference on Technologies for Sustainable Development (ICTSD), 2015.

[13] C. Gonzalez, R. Woods, S. Eddins, Digital Image Processing Using MATLAB, Gatesmark Publishing, 2009.

[14] M. Dubuisson and A. Jain, "A Modified Hausdorff Distance for Object Matching", Processing International Conference on Pattern Recognition, Jerusalem, Israel, pp 566-568, 1994.

[15] K. Wegner, O. Stankiewicz, T. Grajek, M. Domański, "Depth map formats used within MPEG 3D technologies", ISO/IEC JTC1/SC29/WG11, Geneva, January 2017.

[16] C. Gonzalez, R. Woods, S. Eddins, Digital Image Processing Using MATLAB, Tata McGraw Hill, 2011.

[17] M.S. Farid, M. Lucenteforte, M. Grangetto, "Perceptual Quality Assessment of 3D Synthesized Images," in IEEE International Conference on Multimedia and Expo (ICME), Hong Kong, 2017, pp. 505-510.

[18] L.Zhang, X.Mou, D.Zhang, FSIM: a feature similarity index for image quality assessment, IEEE Transactions Image on Processing. 20 (8) (2011) 2378–2386.

[19] A. Marques, "Automatic Road Pavement Crack Detection using SVM", IST, Dissertation submitted for obtaining the degree of Master in Electrical and Computer Engineering, 2012.

[20] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," IEEE Transaction on Image Processing, vol. 13, no. 4, pp. 600–612, April 2004.

[21] Z. Wang, E.P. Simoncelli, and A.C. Bovik, "Multiscale structural similarity for image quality assessment," in Processing IEEE Asilomar Conf. on Signals, Systems, and Computers, 2003, vol. 2, pp. 1398–1402.

[22] D.M. Chandler and S.S. Hemami, "VSNR: A Wavelet Based Visual Signal-to-Noise Ratio for Natural Images," IEEE Transaction on Image Processing, vol. 16, no. 9, pp.2284–2298, Sept 2007.