A Localized No-Reference Blurriness Measure for Omnidirectional Images and Video

Hannes Fassold, Stephanie Wechtitsch
DIGITAL - Institute for Information and Communication Technologies
JOANNEUM RESEARCH
Steyrergasse 17, 8010 Graz, Austria
Email: firstname.lastname@joanneum.at

***Abstract*—Blurriness is a defect commonly occurring in conventional video but also in omnidirectional video. In this work, we propose a novel no-reference blurriness measure for images captured with omnidirectional video cameras. These images present unique challenges for quality measures due to their size and due to the equirectangular projection which is commonly employed for them. We base upon a state of the art algorithm and adapt it for the specifics of omnidirectional images. Furthermore, we extend it with a coarse-scale blurriness map for measuring spatially varying blur. We present a novel ground truth dataset which was generated by adding spatially varying gaussian blur of different magnitude in a viewport-centric way. Experiments with the proposed algorithm on this dataset show a strong correlation of the localized blurriness measure with the ground truth.**

***Index Terms*—image quality measure, no-reference blur assessment, omnidirectional image, 360° video, VR**

I. Introduction

Omnidirectional (360°) video content recently got very popular in the media industry, because it allows the viewer to experience the content in an immersive and interactive way. Omnidirectional consumer video cameras like the Samsung Gear 360 or the Ricoh Theta V have multiple lenses and capture images which cover the whole viewing sphere, typically in 4K or UltraHD resolution. The whole viewing sphere is encoded in one 2D image for each timepoint, usually in equirectangular projection [4]. Omnidirectional videos are typically consumed with a head-mounted display (HMD), so that the user is free to choose the area (viewport) within the sphere he is currently interested in. A high-quality omnidirectional video, without major defects and distortions, is important in this context in order to provide the viewer with an optimal quality of experience.

Therefore, methods for measuring the quality of omnidirectional video automatically are very valuable during capturing and post-production. In this work we will focus on blurriness estimation, as it is a distortion commonly occurring in video, e.g. due to loss of focus or dirt on one the camera lenses of the 360° camera.

The paper is organized as follows. Section II gives information about related work in the literature. In Section III we present the proposed algorithm for localized blurriness. In Section IV we describe the novel ground truth dataset for the evaluation of the localized blurriness measure and give information how it was generated. In Section V we present the results of the evaluation of the proposed algorithm on the ground truth dataset, and Section VI concludes the paper.

II. Related work

Image quality metrics can be categorized either as full-reference, reduced-reference or no-reference, depending on the amount of information provided about the original image. For full-reference quality metrics, the original image (without defects) is provided. Reduced-reference quality metrics take into account partial information derived from the original image. For no-reference quality metrics, no information is available about the original image. In many important applications in media production and delivery, only no-reference quality metrics can be employed because only the distorted image is available. Therefore, we will focus in the following overview of related work on no-reference blurriness measures.

In [7] a spatial domain blurriness metric is proposed, based on the concept of just noticeable blur (JNB). An algorithm using the analysis of edges and adjacent regions in images is proposed in [5]. A transform based method [11] uses statistics of the discrete wavelet transform (DWT) coefficients in natural images to produce blurriness scores for compressed images. In [14] it is shown that local phase coherence (LPC) changes and that precisely localized features, e.g. sharp edges, cause a strong LPC in the complex wavelet transform domain.

Another blurriness metric which operates in the wavelet transform domain is proposed in [3]. The algorithm analyzes the local phase coherence of complex wavelet coefficients, assuming that blur causes a disruption of local phase near sharp image features. In [13] a method with low computational complexity is proposed, which measures the log-energy in high frequency discrete wavelet transform sub-bands. The authors of [1] present a blurriness measure which employs a combination of natural scene statistics, multi-resolution decomposition methods and machine learning techniques. In [2] a statistical analysis of the local edge gradient is done in order to form a Perceptual Sharpness Index (PSI). The algorithm proposed in [6] assesses the amount of blur using the fourier transform and spatial pyramids. The authors of [15] propose a deep learning based blur estimation method which employs a two-stage pipeline for blur classification and parameter estimation. A pre-trained deep neural network is used for classifying the blur

978-9-0827-9703-9/19/$31.00

Fig. 1. Illustration of the algorithm workflow. Top: input image, middle: binary edge image, bottom: blocks from which the blurriness is calculated. Best viewed in color.

type and a general regression neural network for estimating the blur kernel parameters.

All existing no-reference blurriness metrics in the literature are designed for image content taken with a conventional camera and therefore do not take into account the specifics of omnidirectional image content. E.g. due to the equirectangular projection which is commonly employed, the areas of the sphere which are away from the equator are stretched in the image. Furthermore, blurriness can change across the captured image (e.g. as the properties of the different lenses of the 360° camera are different), so the algorithm should be able to measure blur locally and consequently deliver a coarse-scale blurriness map instead of a single value.

In the following, we will describe the first (to our knowledge) no-reference blurriness measure which takes into account the specifics of omnidirectional images and which is able to measure spatially varying blur.

III. Proposed algorithm

The proposed algorithm for measuring blurriness is based on the work presented in [2]. In Section III-A, a brief description of the base algorithm from [2] is given, Section III-B describes the necessary algorithm extension for omnidirectional images and Section III-C describes the algorithm extensions for spatially varying blur.

A. Base algorithm

In the first phase, a binary edge image is calculated from the input image. For this, gradient images are calculated from the input image and the average gradient magnitude is calculated. From this, an adaptive threshold is calculated which is used for calculating the binary edge image. An edge thinning is applied then, yielding the edge image.

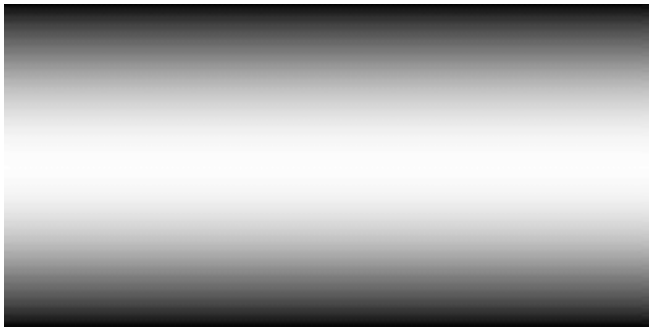

Fig. 2. Distortion map for vertical edges.

In the second phase, the edge width image is calculated, which gives us for a certain pixel in the edge image the width (in pixels) of the corresponding image edge. Only edges with approximately vertical orientation are taken into account. For an edge pixel, its edge width is now calculated by tracing along the edge gradient until a minimum or maximum intensity is encountered. From the pixel-wise edge width measure, now a block edge width measure is calculated. For this, the image is divided into blocks of size 32×32 pixel. For each block, we calculate the edge width for the block as the average of the per-pixel edge width measurements.

Finally, a global edge width measure for the whole image is calculated as the average of a certain quantile of the block edge width measures. A global blurriness value is calculated from the global edge width measure by applying a nonlinear function. A visualization of some stages of the algorithm is given in Figure 1.

B. Extensions for omnidirectional image content

Some general extensions have been done to the base algorithm in order to cope better with omnidirectional image content. Firstly, the phases where the edge width image is calculated has been extended so that the edge width can be measured also for pixels lying on approximately horizontal edges. This is important for the subsequent localization, in order to have more per-pixel measurements available. Especially in indoor scenes, there are naturally a lot of approximately horizontal edges in the content (e.g. from tables or other furniture). Furthermore, several parameters have been adapted in order to account for the size of the input images. E.g., an Ultra HD image captured by an Insta360 Pro camera has a size of 3,840 x 2,120 pixel. In order to compensate for the doubling of the image size (from Full HD to Ultra HD), the block size has been increased also from 32×32 pixel to 64×64 pixel, and all thresholds relying on the block size have been adapted accordingly.

The input image of the blurriness algorithm is one frame of the omnidirectional video in equirectangular projection (for runtime considerations, we do not measure the blurriness in the viewports as rendering many viewports is computationally

Fig. 3. Visualized coarse-scale blurriness map (4×2 matrix). The blurriness value is overlaid in red tones (more intense red means higher blurriness).

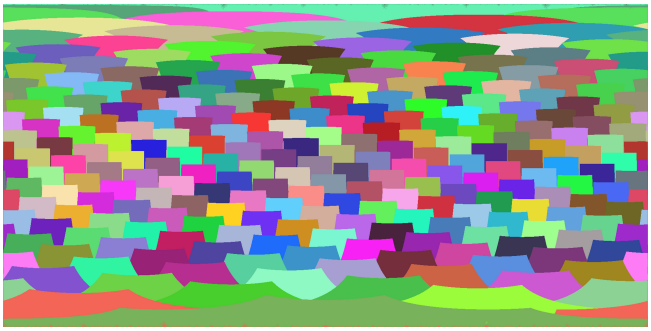

Fig. 4. Visualization of the 240 overlapping viewports employed in viewport-centric image transformation.

expensive). Consequently, this means that the measured per-pixel edge widths (in the second phase) are biased. E.g., vertical edges are stretched (enlarged) out due to the employed projection.

In order to correct for this bias, we calculate distortion maps for horizontal and vertical edges, which give us for each pixel the correction factor C by which the measured edge width has to be multiplied in order to correct for the bias. It is possible to pre-calculate these distortion maps, as they depend only on the resolution of the input image and not on the actual content. In the following, the calculation of the distortion map for vertical edges is briefly sketched (for horizontal edges, the procedure is quite similar):

- For each n^{th} pixel P_i in x and y direction, a viewport V is generated. The viewport has a horizontal field of view of 75° and an $16:9$ aspect ratio. The viewport width is set as a fraction $(1/m)$ of the input image width, which ensures that the blurriness measure is independent of the resolution of the input image. We set the sampling distance n to 10 and m to 5.
- The point Q_i is calculated as the horizontal neighbor of P_i via the formula $Q_i = P_i + (0,\, \delta)$. The value δ is chosen as 4–5 pixel.
- Both points P_i and Q_i are mapped into the viewport V, yielding the mapped points P_v and Q_v .
- The correction factor C is now calculated via the formula $C = \frac{\|P_v - Q_v\|}{\|P_i - Q_i\|}$.
- From the sparse map of correction factors, a dense map is generated for each pixel via the spatial interpolation method given in [9].

The distortion map for vertical edges is visualized in Figure 2.

C. Extensions for localization

The base algorithm calculates only one global blurriness value for the whole input image. So in order to calculate a coarse-scale blurriness map for measuring spatially varying blur, the last phase of the algorithm had to be modified as follows.

From the block edge widths, the goal is to calculate a coarse-scale blurriness map, e.g. as a 4×2 matrix of blurriness values. This corresponds to the classical problem of interpolating function values at regular spaced points of a grid from known function values at irregularly sampled points, namely from the block edge widths which are the result of the intermediate phase of the algorithm. In order to calculate the function values at the points of the regular grid, we employ interpolation based on radial basis functions [10]. Specifically, we employ thin-plate splines

$$\tau\,(r) = r^2 \log\,(r)$$

as radial basis functions and employ Tikhonov regularization [8] in order to avoid overfitting. The visualized coarse-scale blurriness map can be seen in Figure 3.

IV. Evaluation Dataset

For an objective evaluation of the proposed algorithm, a ground truth dataset of omnidirectional images affected by spatially varying blur of different magnitude is needed. Unfortunately, such a dataset is not available, therefore we had to generate it synthetically. For this, we propose in Section IV-A a novel *viewport-centric* method for transforming an omnidirectional image in a certain way (e.g., add synthetic blur in the viewport). In Section IV-B we utilize this method for the generation of the ground truth dataset used for evaluation.

Coordinates on the viewing sphere are usually given in a longitude-latitude representation. In the following, the longitude is always denoted by ϕ and has the range $[-180, 180]$. The latitude is always denoted by θ and has the range $[-90, 90]$. All angles are in degrees.

A. Viewport-centric image transformation

The method we propose for viewport-centric transformation of an omnidirectional image I (in equirectangular projection) consists of three main steps. We first decompose the viewing sphere into a set of overlapping viewports $\{v_i\}$ so that the full sphere is covered and render a viewport image r_i for each viewport. Each viewport image r_i is then transformed in a certain way depending on the application (e.g. add blur or add noise) which gives its transformed version $\widetilde{r_i}$. In the last step, the transformed omnidirectional image $\widetilde{I}$ is rendered as an aggregation of the transformed viewport images $\widetilde{r_i}$. In the following, we will outline the first and last step more in detail.

Fig. 5. Some of the references images used for generation of the ground truth dataset.

Viewing sphere decomposition For the decomposition of the viewing sphere, we generate a set of points $\{c_i\}$ which are approximately uniformly distributed on the sphere. These points will serve then as the center of the respective viewport v_i. We employ the *Vogel method* [12] for generating approximately uniformly distributed points. It basically wraps a planar sunflower-like spiral around a sphere, employing the golden angle during the construction of the spiral. For each point c_i, we create a quadratic viewport v_i with c_i as its center.

We employ 240 viewports, with each viewport having a horizontal and vertical FOV of 24 degrees. The FOV has been determined heuristically in a way so that each point of the viewing sphere is contained in at least one viewport. For each viewport, a viewport image r_i is now rendered from the omnidirectional image I employing bilinear interpolation. A visualization of the viewports (back-projected to the image in omnidirectional projection) can be seen in Figure 4.

Aggregated rendering For rendering the transformed omnidirectional image $\widetilde{I}$ from the set of transformed viewport images $\widetilde{r_i}$, we employ a method which is similar in spirit to the image warping algorithm described in [9]. For the aggregated rendering, we employ an accumulator image A and a weight image W. Both are floating-point images of the same size as the image I and are set to zero initially.

For each transformed viewport image $\widetilde{r_i}$, we iterate over all the source pixel p_v of it and map them to the corresponding position p_i in the omnidirectional image I. The position p_i has typically non-integer coordinates, so we cannot write the source pixel intensity directly. We instead write into the four surrounding pixels of the position p_i via a sort of 'bilinear writing'. For that, we increment the four surrounding pixels of p_i in the accumulator image A and in the weight image W. The amount of increment depends on the distance of p_i to the specific pixel neighbor. After processing all viewports, the transformed omnidirectional image $\widetilde{I}$ is calculated via $\widetilde{I} = A/W$ (pixel-wise division).

B. Ground truth dataset generation

For the generation of the ground truth dataset used for evalution of the localized blurriness algorithm, we extracted a set of representative reference images from several 360° videos which were considered as sharp by a human. Gaussian blur of varying magnitude and with different spatial distribution was then added to these reference images with the method described in Section IV-A, which gives us the ground truth dataset. In the following, the procedure is described more in detail.

TABLE I
PERFORMANCE METRICS FOR THE WHOLE DATASET AND SEPARATELY FOR THE SUBSETS WITH NO, MEDIUM OR STRONG SPATIAL VARIATION.

metric	spatial variation			
	combined	no	medium	strong
PLCC	0.8512	0.8852	0.8688	0.8264
SRCC	0.8413	0.8520	0.8511	0.8211

Specifically, we extracted 60 reference images (see Figure 5 for some examples) from 12 different omnidirectional videos. The video sequences are quite diverse and contains both indoor and outdoor scenes with different motion characteristics. The resolution of the reference images is Ultra HD (3,840 x 2,120 pixel).

From each reference image, we generate a set of 80 synthetically blurred images together with their corresponding ground truth blurriness map. For this, we first define a set of 9 values $\sigma_i = 1.25^i, i = 0 \dots 8$. These give us the base values for the amount of gaussian blur to be added, which are then multiplied with a spatially varying function $\rho(\phi, \theta)$.

We employ 3 models for the spatial variation of the synthetic blur: no spatial variation, medium spatial variation and strong spatial variation. For medium and strong spatial variation, the variation is either in ϕ or in θ (not in both), and it can be either linearly ascending or descending (linear ramp) in the range $[a, b]$. For medium spatial variation, we define this range as $[0.8, 1.2]$, whereas for strong spatial variation it is defined as $[0.6, 1.4]$. This gives us 9 different functions $\rho_j(\phi, \theta), j = 0 \dots 8$ for modeling the spatial variation - one for no spatial variation and four functions for medium as well as strong spatial variation. So we generate 81 spatially varying functions

$$s_{ij}(\phi, \theta) = \sigma_i \cdot \rho_j(\phi, \theta)$$

which for a certain position (ϕ, θ) give us the amount of blur (as the sigma of the gaussian blur kernel which will be applied to the viewport image) to be added there.

By transforming each reference image with all 81 blur functions s_{ij}, we finally produce a dataset of 4,860 synthetically blurred omnidirectional images. For each synthetically blurred image, the corresponding ground truth blurriness map is also calculated from the blur function s_{ij}. We assume that the blurriness algorithm is calculating a coarse-scale blurriness map with dimensions 4×2.

V. EXPERIMENTS AND RESULTS

For evaluating the performance of the proposed localized blurriness algorithm on the ground truth dataset, two different metrics were employed: Pearson linear correlation coefficient (PLCC) and Spearman rank order correlation coefficient

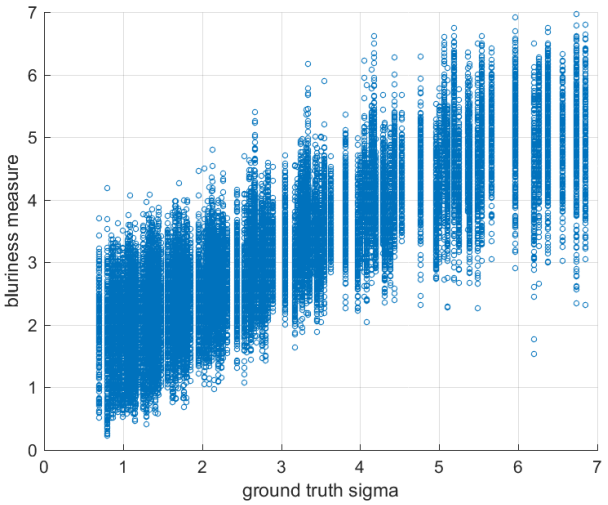

Fig. 6. Scatter plot of ground truth sigma (x-axis) versus the blurriness measure reported by proposed algorithm (y-axis).

(SROCC). The blurriness algorithm calculates for each image a coarse-scale blurriness map with matrix dimension 4×2 which is compared to the respective ground truth blurriness map. By accumulating over all 4,860 synthetically blurred images of the dataset, we obtain a set of $38,880$ tuples, from which the specific performance metric is calculated.

The results are presented in table I. We can see that the results of the proposed localized blurriness algorithm are highly correlated with the ground truth, with a Pearson linear correlation coefficient of 0.8512 and a Spearman rank order correlation coefficient of 0.8413. The high correlation can also be observed from Figure 6. We calculated the performance metrics also for subsets of the whole dataset, specifically for the subsets corresponding to one of the three spatial variation models of synthetic blur: no spatial variation, medium spatial variation and strong spatial variation. The results which are also given in table I indicate that the performance of the algorithm is still very good for all subsets, but decreases somewhat when the spatial variation of the synthetic blur is larger.

The runtime of the proposed algorithm for one omnidirectional image in Ultra HD resolution is approximately 300 milliseconds on a Xeon QuadCore 3.5 Ghz CPU. This allows for real-time processing of a $360°$ video in Ultra HD resolution when temporal subsampling (by taking every 10^{th} frame) is employed.

VI. Conclusion

In this work, we presented a novel no-reference blurriness measure for omnidirectional images and video. It builds on a state of the art algorithm and adapts it for the specifics of omnidirectional images like the equirectangular projection. Extensions for localization are added in order to measure blur which is spatially varying. For evaluation of the algorithm, a novel ground truth dataset was generated by adding spatially varying gaussian blur of different magnitude in a viewport-centric way. The novel method for viewport-centric image transformation could be used also for other image transformations (e.g. for adding gaussian noise). Experiments with the proposed algorithm on the ground truth dataset show a strong correlation of the localized blurriness measure with the ground truth.

Acknowledgment

This work has received funding from the European Union's Horizon 2020 research and innovation programme, grant n° 761934, Hyper360 ("Enriching 360 media with 3D storytelling and personalisation elements"). Thanks to the reviewers and Werner Bailer for the helpful suggestions.

References

[1] M. Chen and A. Bovik. No-reference image blur assessment using multiscale gradient. *Journal on Image and Video Processing, EURASIP*, pages 1–11, 2011.

[2] C. Feichtenhofer, H. Fassold, and P. Schallauer. A perceptual image sharpness metric based on local edge gradient analysis. Signal Processing Letters, IEEE, pages 379–382, 2013.

[3] R. Hassen, Z. Wang, and M. Salama. No-reference image sharpness assessment based on local phase coherence measurement. *IEEE ICASSP*, pages 2434–2437, 2010.

[4] H. Lee, Y. Tateyama, and T. Ogi. Realistic visual environment for immersive projection display system. In *2010 16th International Conference on Virtual Systems and Multimedia*, pages 128–132, Oct 2010.

[5] P. Marziliano, F. Dufaux, S. Winkler, and T. Ebrahimi. Perceptual blur and ringing metrics: Applications to jpeg2000. Signal Proc.: Image Comm., 19:163–172, 2004.

[6] E. Mavridaki and V. Mezaris. No-reference blur assessment in natural images using fourier transform and spatial pyramids. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 566–570, Oct 2014.

[7] N. D. Narvekar and L. J. Karam. A no-reference image blur metric based on the cumulative probability of blur detection (CPBD). *Image Processing, IEEE Transactions on*, 20(9):2678–2683, 2011.

[8] M. J. L. Orr. Regularisation in the selection of radial basis function centres. *Neural Computation*, 7:606–623, 1995.

[9] J. Rosner, H. Fassold, P. Schallauer, and W. Bailer. Fast GPU-based image warping and inpainting for frame interpolation. *International Conferences on Computer Graphics,Vision and Mathematics*, 2010.

[10] R. Schaback. A practical guide to radial basis functions. *Electronic Resource*, 2007.

[11] H. Sheikh, A. Bovik, and L. Cormack. No-reference quality assessment using natural scene statistics: Jpeg2000. *Image Processing, IEEE Transactions on*, 14(11):1918–1927, 2005.

[12] R. Swinbank and J. Purser. Fibonacci grids. *AMS 13th Conference on Numerical Weather Prediction*, 1999.

[13] P. V. Vu and D. M. Chandler. A fast wavelet-based algorithm for global and local image sharpness estimation. *IEEE Signal Processing Letters*, 19(7):423–426, 2012.

[14] Z. Wang, E. P. Simoncelli, and H. Hughes. Local phase coherence and the perception of blur. *Conference on Neural Information Processing Systems*, pages 786–792, 2003.

[15] R. Yan and L. Shao. Blind image blur estimation via deep learning. *IEEE Transactions on Image Processing*, 25(4):1910–1921, April 2016.