BETWEEN PASS/FAIL TEST PATTERNS AND STANDARD SUBJECTIVE TESTING: A NEW METHODOLOGY TO DESIGN TEST SETS AND GUIDELINES FOR VISUAL QUALITY SCORING

J. E. Caviedes, M. Subedar, I.J. Ndiour, T. Lee, and N. Ahuja
Intel Corporation

ABSTRACT

This paper presents a methodology for designing perceptually relevant, subjective visual metrics used to assess and quantify the performance of video-processing algorithms. For a given video processing algorithm being evaluated, the methodology involves determining a set of key picture attributes, their priorities, and the algorithmic failure modes with associated visual impact. A minimal set of input test clips featuring such attributes in a differentiable manner must be chosen, along with clear guidelines on how to assign quality scores to the outputs. An important property of the test set and scoring options is the matching of the performance curve of the algorithm which combines the relative importance of the test cases, the number of them, and the ability to define minimum acceptable scores for different use cases. Both the selection of perceptually relevant test clips, as well as the construction of unambiguous scoring guidelines to match the performance curve of the algorithm will be addressed in this article.

1. INTRODUCTION

Video quality assessment approaches can broadly be classified into two categories - subjective methods and objective methods. With either type of approach, the intent is to obtain a single numeric value that is representative of the quality of the video. Subjective methods typically involve scoring of the video by multiple human observers and aggregating their scores into a single number. By contrast, objective methods measure quality by either comparing the video to a supposedly ideal "reference" video (full-reference video quality metrics), or by attempting to measure quality directly from the video based on a perceptual model of the human visual system (no-reference quality metrics) ([1]-[3], and references therein). Objective metrics have the distinct advantage of being efficient to implement, possibly in real-time. Unfortunately, scores obtained by such methods generally do not correlate well with the quality perceived by humans [4]. In subjective methods, however, because the video is being assessed and scored by human observers, the scores obtained are known to be the best predictors and indicators of perceived visual quality. An important consideration in the design of a subjective evaluation methodology is the repeatability of scores obtained when the evaluation is performed either by different sets of observers, or at different points in time. In simple MOS (mean opinion score) based approaches, it is nearly impossible to achieve similar scores under multiple trials. This drawback may be addressed by adopting the DSCQS (double stimulus continuous quality scale) method of performing quality tests. In this method, viewers are shown pairs of video sequences, one of which serves as a reference, in a randomized order and are asked to rate the quality of each sequence in the pair. The difference between these two scores is then used to quantify changes in quality. If a MOS score is assigned to each clip, the resultant score is the widely used differential MOS (DMOS) metric. This method, though widely accepted as an accurate test method with little sensitivity to context effects, suffers from a practical problem of requiring a reference clip for scoring [5, 6].

In this paper, a subjective approach to assess the quality performance of a video algorithm or video-processing pipe is presented that attempts to address the limitations of the prevalent VQA methods described above. The method involves choosing a minimal set of input test clips along with clear guidelines on how to assign quality scores to the outputs. This not only eliminates the need for reference outputs, but also minimizes the variability of scores obtained across multiple trials as the scores are assigned in accordance with the provided guidelines rather than arbitrarily by each individual observer.

The concept of using test sets to evaluate visual quality of video algorithms is, by itself, not new. Test-suites, such as HQV [7], exist and have been used for this purpose. Such method, however, does not have a clear model to combine pass/fail tests with continuous visual performance tests of varying criticality. What is addressed here is the methodology by which such tests and scoring guidelines should be designed such that the resultant score obtained by using this set is reflective of perceived visual quality. This is achieved by choosing each clip such that it will

produce a specific visual impact by exercising one or more key failure modes of the algorithm- or pipe-under-test. Details of this process are described in Section 2. While the total score serves as an indication of overall quality of algorithm being tested, examining the scores for each individual clip allows for a more fine-grained assessment of specific strengths or weaknesses of the algorithm. The method presented here can be used both for standalone quality evaluation of a single algorithm and for comparing quality of several algorithms attempting to perform the same processing task. A specific example of how to apply this process to design a test set for upscaling algorithms is described in Section 3. Conclusions are presented in Section 4.

2. DESIGN OF TEST-SET AND SCORING GUIDELINES

The proposed subjective benchmarking metric is described in this section. The subjective benchmarking metric includes a test suite, along with well-defined scoring guidelines. The steps involved in developing the metric are given in Figure 1.

2.1. Define video processing feature under test

The video processing feature under evaluation needs to be chosen. This feature can be one algorithm, or a combination of algorithms, or a complete system which includes all the processing blocks of a video chain. In order to allow fair comparison between different solutions, it is important to clearly specify the main processing blocks that can be enabled when scoring the proposed subjective benchmarking metric.

2.2. Define test environment

A good visual quality benchmarking metric needs to be repeatable and has very minor room for ambiguity. It is important to clearly define the test parameters for the feature being benchmarked. Some of the important test parameters are given below:

1) Input/output formats: The formats for the test inputs and outputs should be defined. This will include picture format (progressive/interlaced), resolution (480i, 480p, 1080p etc), frame rate (24fps, 60fps, 120fps etc), pixel format (yuv, rgb, hsv etc) and pixel packing format (yuv422, yuv444 etc). If a single processing block is being evaluated, a format that requires minimum pre- processing must be chosen.

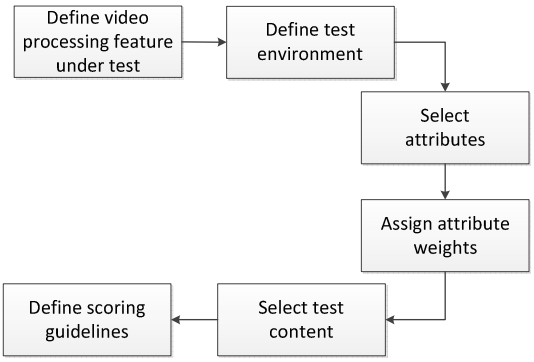

Figure 1: Steps to create subjective benchmarking metric.

2) Platforms: It is important to define the target platforms for the selected benchmarking tests. These can range from mobile phones and tablets to television sets with ultra-high definition formats. In the case of 3D tests, one has to consider passive/active displays, and stereoscopic/auto-stereoscopic displays. The evaluation display has to be representative of the target application.
3) Display calibration and viewing conditions: In order to get repeatable results with the benchmarking tests, one has to provide viewing conditions and appropriate display calibration steps. Some algorithms such as color and contrast enhancement are very susceptible to viewing conditions and display calibration. It is difficult to match different types of displays (LCD/Plasma) and different brands of displays. These constraints should be taken into account when designing the benchmarking metric.

2.3. Select attributes

In selecting test clips for the benchmarking metric, it is essential to take into account failure modes of the algorithm- or pipe-under-test. Failure modes are conditions under which the test algorithm can break down, resulting in visual impairments in the output. As an example, fast motion (out of search range) would be a failure mode of algorithms that rely on motion-estimation. Failure modes can typically be directly mapped onto characteristics or attributes of the input test clips. It is essential, therefore, that the test clips selected have relevant attributes such that the failure mode of the algorithm is exercised. Some examples of attributes which need to be commonly considered for video processing algorithms are:
1) Noise level: Noise level is an important attribute which needs to be considered for many of the video processing algorithms. Some algorithms will be very sensitive to noise level e.g. motion estimation, noise reduction, and sharpness/contrast enhancement.

2) Noise type: Along with noise level, noise type is an important attribute. Examples of the noise type are Gaussian noise, compression noise and salt & pepper noise.
3) Spatial frequency: Spatial frequencies which are challenging for the feature under test need to be considered. For example, noise reduction algorithm should include both smooth and textured regions. In the smooth regions noise is more visible and on the other hand any strong filtering will introduce softness in the textured regions. Another example will be motion estimation algorithms, which will have difficulty in finding good motion vectors for spatial frequencies with periodic structures.
4) Edge orientation: Edge orientation (diagonal, shallow edges) is an important attribute for interpolation algorithms, such as deinterlacing and image scaling. It is challenging to accurately calculate edge direction at shallow angles, which can manifest into incorrect interpolation and artifacts.
5) Motion type: Motion is an important attribute for several algorithms which depend on motion detection and estimation. Amount of motion (fast or slow) and type of motion (panning, zooming) should be considered in selecting the motion attribute.
6) Overlays/transparencies: Overlays and transparencies in the sequence can be challenging for some of the algorithms, especially which involve motion detection or estimation. These can be typically found in the content with waterfalls, fountains or video edits with fade-in and fade-outs.
7) Chroma information: Chroma can be an important attribute for some of the interpolation and motion detection related algorithms. The chroma related issues in the algorithm will be more apparent at higher output resolutions.

All the sequences in the test database should be classified according to the selected attributes. It is expected that some sequences will have more than one attribute, in which case one can include relative weight of each attribute.

2.4. Assign weights to attributes

In this step, priority or weights are assigned to the attributes included in the benchmark metric in accordance with their perceptual relevance. The relevance of various attributes can be obtained by domain experts directly from the processing requirements of the algorithms involved, and by studying their failure modes under a large and diverse set of test inputs. The set of sequences which can stress the functionality of the feature or algorithm will be candidates to be included in the benchmark set.

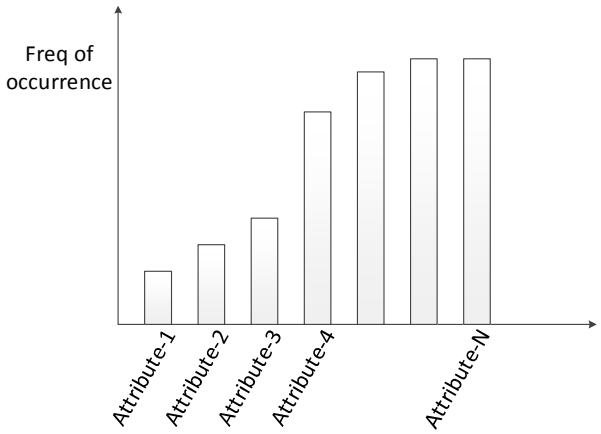

Figure 2: Histogram showing frequency of occurrence of failure modes for selected attributes.

It is important to study at least two different solutions or algorithms to reliably predict weights for the attributes. In many cases, prior knowledge of the working of the algorithm can be helpful in determining the weights for the attributes. After studying the failure modes for the selected attributes, a histogram of frequency of occurrence of the failure modes is plotted, as shown in Figure 2. The frequency of occurrence of attributes is normalized by total number of occurrences of all attributes, to obtain the attribute weight. The weight assignment step is revisited in the event new failure modes are identified. New attributes are added to comprehend the additional failure modes, and the weights are reassigned based on the new histogram.

2.5. Select test content

In this step, test content for the subjective metric is selected based on the chosen attributes. The final number of test clips depends on how many attributes need to be included and their relative weights (Table 1). Each test clip can have more than one attribute, and the column corresponding to the given attribute is marked in the table. After selecting a minimal set of test clips that cover all the relevant attributes, the frequency of each attribute is calculated by summing the attribute columns.

Sequence Names	Attr. 1	Attr. 2	Attr. 3	…	Attr. m	# of Attr.
Seq-1	x		x			2
Seq-2		x	x			2
Seq-3	x					1
…						1
Seq-n		x	x		x	3
Freq. Of Occurrence	**4**	**6**	**10**		**25**	
Attr. Weight	**0.1**	**0.12**	**0.20**		**0.45**	**1.0**

Table 1: Illustrative example of attribute weight selection.

The normalized frequency of occurrence of attributes provides the attribute weight. The attribute weights thus calculated should resemble the relevance curve of the attributes. Hence, the sequence selection may need to be iterated till the relative weights of different attributes matches the one defined after studying the failure modes for the algorithm under test.

Based on the attributes present and their weight, the researchers will create a scoring guideline for each sequence. Typically a discrete range between 1-5 is used, with variable points reduction for failure or deficiencies related to each attribute.

3. CASE STUDY: IMAGE UPSCALING METRIC

In this section, we apply the presented methodology to the particular case of an image upscaling metric. The goal of the exercise is to demonstrate the practical use of the methodology to construct an image upscaling metric by following the different steps described in Section 2, from characterizing the principal failure modes for scaling algorithms to building the minimal test set with associated guidelines. In line with the processing requirements, the metric evaluates the upscaling technique's capability of preserving the input picture quality across different scales. When the upscaling technique introduces distortions along a given picture attribute vector (e.g. high-contrast edges), these manifest in the form of undesired image artifacts on the upscaled picture (e.g. jaggies).

The first step is to define the perimeter of scaling solutions being evaluated. We are interested in a scaling metric targeted at standalone single-frame upscaling methods. Since scaling is the feature of interest, typical picture enhancements such as noise reduction or sharpness enhancement need to be disabled whenever present. Temporal super-resolution and other methods using multiple input images are beyond the scope of techniques being evaluated.

Secondly, related to the metric application, we need to define the test environment. Techniques under evaluation include software algorithms (e.g. Photoshop®, and SmartEdge [8]) but also hardware capabilities in CE devices such as HDTV, tablets, and phones. Quality performance is sought for both still and video content. Performance on interlaced content is deemed not relevant in order not to contaminate the scaling quality with deinterlacing quality. Fair and good quality test content is considered, with input video resolutions of 720x480, 1280x720 and 1920x1080 and output video resolutions of 1920x1080 and 3840x2160. It is advised to perform visual quality evaluation on calibrated 1080p and 4K QFHD professional displays. In the case where professional displays are not available, regular HDTV or 4K TV can be used after calibration (with all picture enhancement options disabled).

Subsequently, the key picture attributes for scaling are defined and prioritized. The process to define these was based on extensive testing, using multiple scaling algorithms on a large data set to identify challenging attributes for scaling.

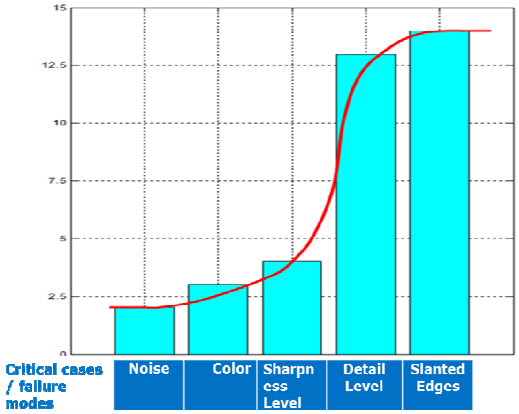

Figure 3: Attributes distribution in the test set: actual (bars) vs. theoretically defined (red).

Sequence Name	Output Res	Slanted Edges	Spatial HF	Color	Sharpness level	Noise
Video1-720p	2160p	3	3			2
Still1 (720p)	2160p	3	2		1	
Video2 (480p)	1080p	3		3		
Still2 (720p)	2160p	2	3		1	
Video3 (480p)	1080p	2	2	1	1	
Still3 (480p)	1080p	1	3			
Total		**14**	**13**	**4**	**3**	**2**

Table 2: Selected test clips with attribute weights.

Expert knowledge derived from the literature is also used for the attributes selection. Following this process, five attributes were selected. For each attribute, we also specify the corresponding observable degradations in the upscaled pictures. By order of priority, these attributes are classified as follows:

1. Slanted edges (high-contrast edges, thin edges) → Jaggies and Roping.
2. Detail level (spatial high frequencies such as natural texture, geometric patterns, vertical/horizontal edges) → Blur, loss/degradation of details, Ringing, Moiré, flicker.
3. Sharpness level → Blur.
4. Noise level (analog, compression) → Noise amplification, flicker.
5. Color (all of the above) → Color artifacts (all of the above).

Following the attributes definition, we selected a minimal set of clips to form the scaling test set. The clips are selected so that the frequency of occurrence of attributes

present in the scenes match the defined priority (obtained following Section 2 instructions). Figure 3 illustrates such matching. Both natural content and graphics imagery were used to populate the test set. Table 2 provides frequency of occurrence of each attribute in the selected test clips, with totals along the columns highlighting the relative importance of a given attribute in the entire test set.

Using the test set, we provide clear guidelines for scoring the performance of scaling algorithms. First for each test clip, we consider the cumulative weights of all attributes present in said clip. This number is obtained by totaling along the rows of the previous table and represents the overall scaling complexity for the test clip. The maximum score should directly derive from this complexity number and reflects how challenging a particular test clip can be to upscale. Here, we apply simple thresholds to the complexity numbers in order to allocate a maximum scaling score to each clip in the set. The maximum score for each clip also represents its relative importance within the test set. Then for each test clip, we examined the visual artifacts obtained with poor scaling performance and determined clear, unambiguous rules instructing subjects on how to deduct points from the maximum score. One such example is shown in Figure 4. Since jaggies are observed at different edge orientations for different scaling methods, the addition of visual aids and color coded regions onto the test picture enable simple, yet strict instructions for scoring, based on observed quality performance in each region.

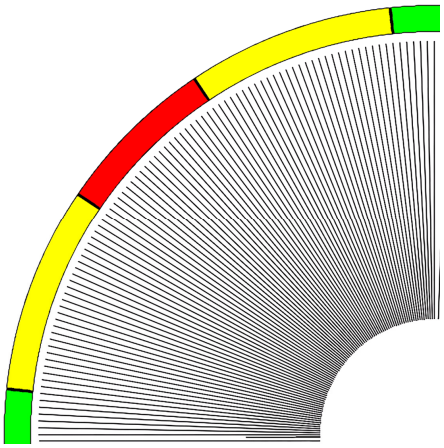

Figure 4: Example simple scoring guidelines. 7: No jaggies at all angles. 5: Jaggies only in green regions. 2: Jaggies in yellow regions. 0: Jaggies in red regions or objectionable quality.

The definition of the scoring guidelines for all test clips completes the metric. For this example of a scaling metric, we tested it using 4 different algorithms (bicubic, SmartEdge, linear polyphase, enhanced linear polyphase) and 5 subjects with expertise in visual quality evaluation. Equipment used includes a calibrated professional 55inch 4K display and a real-time playback system. Table 3 summarizes the total score assigned to each scaling solution by the subjects when following the guidelines.

Subject	Bicubic	Smart Edge	Poly phase	Enhanced Polyphase
1	9	12	15	23
2	9	12	12	23
3	12	12	15	25
4	9	12	15	23
5	9	12	15	23

Table 3: Subjective Scoring results of upscaling solutions.

4. CONCLUSIONS

A new methodology to design test sets and guidelines for visual quality scoring of video algorithms was proposed. This was tested by designing sets for various video processing tasks (scaling, deinterlacing, etc.). The scores obtained with the methodology are consistent with the preferences of human observers. The variability of scores obtained during multiple trials and between multiple observers was also found to be small. The proposed methodology was thus demonstrated to be effective in producing perceptually relevant quality scores in a consistent, repeatable manner.

5. REFERENCES

[1] S. Winkler, "Issues in vision modeling for perceptual video quality assessment," *Signal Processing, Elsevier*, Vol. 78, No. 2, October 1999, pp. 231–252.

[2] Z. Wang, A. C. Bovik, et al, "Image Quality Assessment: From Error Visibility to Structural Similarity," *IEEE Transactions On Image Processing*, Vol. 13, No. 4, April 2004, pp. 600 – 612.

[3] A. B. Watson, "Toward a perceptual video-quality metric," *Proc. SPIE 3299, Human Vision and Electronic Imaging III*, 139, July 1998.

[4] Z. Wang, A. C. Bovik, L. Ligang, "Why is image quality assessment so difficult?", *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vol. 4, May 2002.

[5] ITU-R Recommendation BT.500-11, "Methodology for the subjective assessment of the quality of television pictures," Geneva, 2002.

[6] M. H. Pinson, S. Wolf, "Comparing subjective video quality testing methodologies," *Visual Communications and Image Processing 2003, Proceedings of the SPIE*, Volume 5150, pp. 573-582, 2003.

[7] HQV Video Processing, Link: http://www.qualcomm.com/solutions/multimedia/video-processing.

[8] SmartEdge: http://graphicon.ru/oldgr/en/research/scaling/