Objective metrics for measuring perceptual video quality

Claudio Miceli de Farias
PPGI
IM/NCE- UFRJ
Rio de Janeiro, Brazil
claudiofarias@nce.ufrj.br

Paulo Henrique de Aguiar Rodrigues
PPGI
IM/NCE-UFRJ
Rio de Janeiro, Brazil
aguiar@nce.ufrj.br

Abstract—Assessing video quality is one of the most challenging problems in real time environments. Real time environments usually use codecs defined by ITU-T and MPEG. In these codecs, spatial compression is represented by key frames while other frames represent temporal compression. Video quality is affected by losses, delays and distortion in the frames of a stream. We focus on simple perceptual video quality metrics to be used in low-end systems. We present three algorithms. The algorithms take advantage of the fact that if distortion is concentrated in a determined region, the negative impact in visual perception is higher, and, if a region has greater luminance, the impact is even higher. The first one divides a frame in well defined geometric regions and measures distortion in each region, using luminance as a relevant factor. The second algorithm uses a filter to determine edges in a frame and locate regions. Finally, the third one uses luminance variation between frames for evaluating distortion and appears to be the best. Our metric is compared with PSNR, a well established objective metric. Although our metric is as simple as other objective metrics, it achieves more accurate perceptual quality results, when human visual system metrics (subjective measurements) are taken as reference.

Keywords-video quality assessment; objective video metrics; NR algorithm; low power consumption metrics..

I. INTRODUCTION

In the last few years, there has been a great improvement in digital video quality assessment techniques. Though HVS *(*Human Visual System*)* metrics usage has resulted in more accurate results from human perspective, mostly of these algorithms are related to Full (FR) and Reduced (RR) Reference metrics.

In real time scenarios, having the original video as a parameter to measure transmission quality is extremely difficult, inadequate or impossible. In these scenarios, it is important to use NR metrics (No Reference), which is based on the received video stream only. Recent research has proposed various NR metric algorithms to assess quality [1,2,3,4,6,7], but failing to provide applicable solution to scenarios like IMS (IP multimedia subsystem), where devices would be extremely limited in computing power.

We address two FR simple algorithms and a NR algorithm, all with low resource consumption. By resource consumption, it is meant memory consumption and processor use.

The rest of the paper is organized as follows. In section 2, the algorithms are described. Section 3 presents results of performed experiments. And finally, conclusions are presented in section 4.

II. PROPOSED ALGORITHMS

Assessing real time videos through FR metrics is extremely difficult or inadequate, as it is necessary to get a reliable sample of the source to compare to the streamed video [15]. In recent years, we have seen many NR metrics proposals. Even though delivering good results, these metrics require resources (computing power and memory) not available in very restricted devices, meaning devices that use processors up to ARM 920 series and 64 Mb of RAM. Typical cell phones are in this range. Restricted devices require that assessment algorithms be simple and have implementations able to perform in real time.

We target real time scenarios, as seen in IMS (IP Multimedia Subsystem) and NGN (Next Generation Networks), where simple and fast objective metrics, like PSNR (Peak Signal to Noise Ratio), are widely used. However, PSNR does not consider human characteristics. It uses only pixel information, which does not correlate well with human perception [9].

Our goal is to construct an objective metric that uses some HVS (Human Visual System) features in order to enhance metric performance and it is simple enough to be used in restricted real time communication environment (video calls). We use luminance, instead of channel colors, because the human eye is rather more sensitive to luminance than to colors [9] and, also, because there is less data to evaluate using only luminance (1 channel rather than 3 channels).

II.I First Algorithm

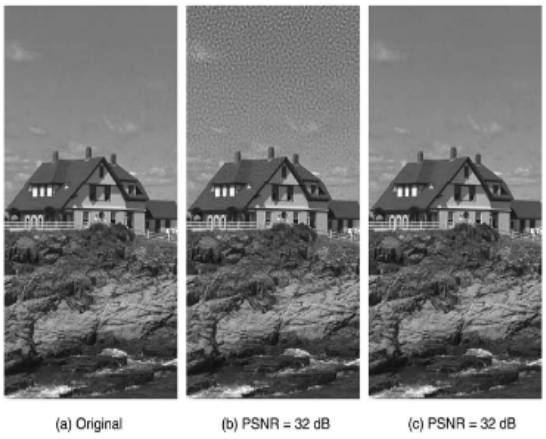

Figure 1. Differences in distortion perception [9]

Regions with low luminance values have greater tolerance to distortion than regions with high luminance values [9]. Figure 1 shows an example of this problem. Images (b) and (c) have same PSNR score, but distortion can be differently perceived in each image. It is easy to see that there are differences in perception impact depending on the affected region. If the affected region has a greater luminance value, as in image (b), the distortion is more perceptible. As we can see, images (a) and (c) have similar quality perception, but quite different PSNR scores. Our first algorithm overcomes this deficiency.

To introduce HVS characteristics in our first algorithm, we divide the image in *N* geometric regions, as follows:

First step: Divide the image in *N* regions.
Second step: Calculate PSNR in each region.
Third step: For each region, calculate the average luminance.
Fourth step: The distortion of a region will be PSNR times the average luminance. Do it for every frame.

The above algorithm would approximately score the same distortion (in db) for images (a) and (c), as desired.

II.II Second Algorithm

Humans focus their attention on objects in scene. If distortion is concentrated in the objects, it has a more severe impact in quality perception than if distortion is spread over the image [4].

In the first algorithm, we divided the scene geometrically; in the following one, we divide an image using its object borders (applying a filter like Sobel [14]). This approach tries to focus on scene objects.

The use of filters generates noise in frames. In order to avoid that filter generated noise is evaluated as a region by the algorithm, a maximum likelihood [10] algorithm between region sizes is used to determine a region size threshold above which a region is considered for evaluation.

First step: Apply the Sobel filter over the image and select the regions it has delimited. Apply the maximum likelihood algorithm in [10] to select the regions for evaluation.
Second step: Calculate PSNR for each region.
Third step: For each region, calculate the average luminance.
Fourth step: The distortion of a region will be PSNR times the average luminance. Do it for every frame.

The two algorithms described previously are still FR metrics. In some video conferencing systems, the video is fed back by the system to the source, allowing for comparison. But, in a regular video call, there is no sending video back for reference. The third algorithm provides a NR metric.

II.III Third Algorithm

Pixel luminance deviation between frames is used as a factor in this algorithm. If the luminance deviation of a pixel is similar to deviation of the region around it, it is considered object movement; on the contrary, it is considered distortion. We also consider the influence of fast-moving regions. Humans can tolerate distortion in a fast-moving region to a considerable extent. [2]

First step: Take frame *n* and apply Sobel filter over the image and select the regions it has delimited. Apply the maximum likelihood algorithm in [10] to select the regions for evaluation. Inside a region, determine *(x,y)* for each pixel.
Second step: For each *(x,y)* pixel inside a region, determine the pixels *(x+1, y), (x-1,y),(x,y+1), (x,y-1).*
Third step: Take frame *(n+1),* the next frame.
Fourth step: Take the average luminance of pixels *(x+1,y),(x-1,y), (x, y+1), (x,y-1)* and *(x,y)* of frame *n* (we call it *Lum(n)*).
Fifth step: Calculate *Lum(n)* and *Lum(n+1).*
Sixth step: If $|(Xn,Yn)-(Xn+1, Yn+1)| <= 2*|Lum(n)-Lum(n+1)|$ where (*Xn*, *Yn)* are the pixel *(x,y)* in frame *n*, we consider that there is distortion and, to weight movement, distortion is considered as $|(Xn,Yn)-(Xn+1,Yn+1)|/max(Lum(n), Lum(n+1))$.
Seventh step: Repeat it for every pixel of a frame and sum all distortions. Do it for every frame.

Distortion is given in cd/m2.

III. RESULTS

The experiments for quality assessment consisted of presenting a set of 100 videos. All video sequences are CIF format (352 X 288), with a frame rate of 25 Hz. We compared our metric with PSNR and subjective results. We followed the guidelines specified by VQEG [8] for quality tests and the results for 25 non expert viewers were selected. The viewers evaluated the video quality in real time using a continuous scale marked with “Very Good”, “Good”, “Average”, “Bad” and “Very Bad”.

Subjective scores were quantized on a scale of [0 -100]. There were four kinds of video: movement, conference, still image and disconnected scenes.

Movement videos are videos that have scene objects in movement (sport videos, vehicles and action videos). Conference videos are videos similar to video calls and classes. Still Image videos are videos that repeat a picture. Disconnected scenes are videos that change scenarios repeatedly.

We normalized all results in order to compare with the output given by our algorithms. The subjective measurements were our goal (in terms of quality) and we expected that the performance of our algorithms would exceed PSNR (in terms of quality) and to be close in terms of resource consumption.

Comparisons between the proposed algorithms and PSNR have the goal of providing a trustful base for extending the comparison to other metrics. Although there are metrics with better performance than PNSR [4], comparison with PSNR is important for estimating the advantage of using HVS characteristics.

As subjective metrics is the only one representing adequately human perception, comparison with it is essential, although subjective measurement may be not applicable to real time environments. As seen in the last session, distortion and perceptible error are not necessarily associated.

All algorithms were programmed in JAVA. Processor usage was measured in MIPS and PSNR implementation scored 5017 MIPS.

The first algorithm had similar results as PSNR in terms of resource consumption (5107 MIPS), but it showed slightly better quality results than PSNR (around 5% of all samples were better related to subjective results than the PSNR measurement).

The second algorithm had higher resource consumption (10% over for processor, 5473 MIPS) and better quality performance than PSNR (9% of all samples were better related to subjective results). On the other hand, when a video had a large number of small details, the metric mixed up noise with regions and generated results over 30% worse than PSNR. It happened with all four video kinds.

The third algorithm had resource consumption a little higher than the second (around 20% over for processor, 6117 MIPS), but had excellent results in quality assessment performance (almost 20% of samples were closer to subjective results). As PSNR has very low resource consumption, having a cost 20% higher still fits in our proposed scenario.

Videos with fast changing scenes (disconnected scenes) were the ones with worse algorithm III scores. Coincidently, these videos were the ones that had major disagreements in subjective quality scores.

Table 1 shows the results with major differences in subjective assessment confidence interval. First column enumerate the videos. Second column presents subjective scores. Third column shows 90% confidence intervals. Fourth column presents PSNR results. In the last column, we present the difference between the subjective scores and PSNR scores.

The worst results for the third algorithm happen when evaluating disconnected scenes. The reason is that this algorithm uses the last frame as reference to the next one.

Table 2 compares our algorithm and PSNR. First column shows the evaluated algorithms. Second column represents the mean difference between the evaluated algorithm and the subjective score and the standard deviation is showed in the third column. The last column represents (Y-X)/X, where X is PSNR processor usage (in MIPS) and Y is proposed algorithm processor usage. Memory consumption was not considered, because 64 Mb proved to be enough for all algorithms and cell phone usage.

As Table 2 shows, the third algorithm achieves a result that is much closer to the subjective metric than the others, including PSNR, which performs worst. Though not shown, PSNR outperformed our third algorithm for videos with frequently changed scenes. This occurred because the third algorithm loses references of sequential frames at the moment of a scene change.

TABLE I. Evaluation of "Disconnected scenes" videos

Video	Subjective score	Confidence Interval	PSNR	Difference
31	17	26,91	38	21
32	8	12,43	5	3
40	67	16,52	50	17
43	90	13,88	79	11

TABLE II. Comparison between the proposed algorithms and PSNR.

	Mean	Standard Deviation	Processor Usage
First Algorithm	6.95	10.65	0%
Second Algorithm	6.09	5.81	10%
Third Algorithm	3.18	2.19	20%
PSNR	7.04	5.69	-

The algorithms 2 and 3 have higher values of standard deviation than PSNR. For algorithm 1, this occurred because distortion did not concentrate in a single region, most of the time. For algorithm 2, the algorithm mixed up noise and regions in videos with small details in scene.

In order to apply our algorithms to colored videos, one might think of applying it to different color channels. However, results would not be good, because color channels are not correlated. Wandell and Poirson [11] have suggested a new color space in order to solve this problem. Van den Branden [12] has adopted this color space in his image quality metric.

IV. CONCLUSIONS

We have observed that Full Reference metrics are not suited to real time media, as there is no reference sample to evaluate the stream. No Reference metrics use only the received stream as input for evaluation.

We have proposed 3 algorithms, using some HVS characteristics in order to improve quality assessment. The first two algorithms are full referenced and use characteristics of the image to assess quality. The third algorithm is a No Reference algorithm and it uses the deviation of luminance between frames to assess quality.

It is possible to notice that the three algorithms have increasing complexity. The higher complexity indicates a better video quality assessment, but it also indicates increasing resource consumption.

There are tradeoffs in the algorithms that could have been used to improve assessment or reduce resource consumption. In the second and third algorithms, we could have used a better filter in order to reduce noise effect, at the expense of increasing power consumption. Nevertheless, in first and second algorithms, we could have used just some frames, a key frame for instance, to assess quality. In the third algorithm, we could have used just some pairs of frames to measure deviation. It would have reduced our quality performance, but it would have improved resource consumption performance.

As future work, the study of a multimedia semantic model can be achieved, once a video quality metric has been chosen. This ontology could evolve into a semantic network management scenario as we could use the video quality results as input to this ontology.

The third algorithm presents an interesting feature. Luminance is a continuous characteristic in videos. There will not be outlier values. It could have recovered from losses through interpolation of frames based on its luminance.

Another interesting application would be video's dynamic adaptation to network condition using the defined ontology. Through the use of some computational intelligence mechanism (such as Fuzzy logic or a neural network) evaluate network conditions and trying to anticipate problems.

Extending E Model [13] to include video characteristics is an extremely important work. If possible, it would allow the development of a multimedia quality assessment model.

The use of proposed algorithms in conjunction with audio metrics could be used to assess multimedia quality [5]. The problem of synchronization between audio and video is complex [9] and the lack of lip sync certainly impacts quality. All present quality metrics do not take audio and video sync in consideration and some novel contribution could be achieved on this matter.

REFERENCES

[1] H. Liu, N. Klomp, and I. Heynderickx, "A No-Reference metric for perceived ringing" Fourth International Workshop On Video Processing and Quality Metrics, Arizona, 2009.

[2] F. Yang, S. Wan, Y. Chang, and H.R. Wu, "A Novel Objective No-Reference Metric for Digital Video Quality Assessment", IEEE Signal Processing Letters, Vol.12, No10, 2005

[3] N. Suresh, P. Mane, and N. Jayant, "Real-Time Prototype of a Zero-Reference Video Quality Algorithm" International Conference on Consumer Electronics, Las Vegas, 2008.

[4] N. Suresh, P. Mane, and N. Jayant,"Testing of a No-Reference VQ Metric: Monitoring Quality and Detecting Visible artifacts", Fourth International Workshop On Video Processing and Quality Metrics, Arizona, 2009.

[5] D. S. Hands, "A Basic Multimedia Quality Model", IEEE Transactions on Multimedia, Vol 6, No6, 2004

[6] A. Neri, M. Carli, M. Montenovo, and F. Cuomi, "No Reference Quality Assessment of Internet Multimedia Services",14th European Signal Processing Conference, Florence, 2006.

[7] R. Palaniappan, N. Suresh, and N. Jyant, "Objective Measurement of Transcoded Video Quality in Mobile Applications", World of Wireless, Mobile and Multimedia Networks, NewPort Beach, 2008.

[8] VQEG Report., "Final Report From Video Quality Experts Group on the Validation of Objective Models of Video Quality Assessment", [Online]. Available: http://www-ext.crc.ca/vqeg/frames.html, 2000.

[9] S. Winkler. Digital Video Quality Visions, Models and Metrics. John Wiley. 2005

[10] A. R. Alves, E. M. Lapolli,R., Bastos, and L. Bastos. Classificacao de imagens pelo método da verossimilhanca - uma nova abordagem. 7 Simpósio Brasileiro de Sensoriamento Remoto. 1993

[11] A. B. Poirson and B. A. Wandell. "Pattern-color separable pathways predict sensitivity to simple colored patterns". volume 36. Vision Research. 1996

[12] C. J. van den Branden Lambrecht . "Color moving pictures quality metric". volume 1. IEEE International Conference on Image Processing. 1996

[13] ITU-T Recommendation G.107; The E-Model, a computational model for use in transmission planning. Genève, 2003

[14] Gonzalez, R. E. W. R. C. Digital Image Processing. Prentice Hall, 2008.

[15] S. Winkler and F. Daufax, "Video Quality evaluation for mobile application" in Proc. SPIE/IS & T VCIP, vol 5150, Lugano, Switzerland, 2003.