



Estimando qualidade de vídeo através de uma nova métrica de percepção de borramento sem referência

Jean Nunes Ribeiro Araujo¹, Claudio de Castro Monteiro²

¹Pós Graduando do Programa de Pós-Graduação em Telemática do Instituto Federal do Tocantins. e-mail: jean.to@gmail.com

²Prof. Doutor do Instituto Federal do Tocantins. e-mail: ccm@ifto.edu.br

Resumo: O aumento da demanda de serviços multimídia sobre redes de computadores tem despertado a atenção da comunidade de pesquisa nos últimos anos. As transmissões de conteúdo utilizando áudio e vídeo vêm ganhando espaço nas universidades, empresas, sistemas de ensino a distância, governo, além de alcançar a todos que possuem acesso a Internet. A grande aceitação e, muitas vezes, a preferência dos usuários, evidencia o crescimento do consumo deste tipo de conteúdo na busca por entretenimento e educação. No entanto, as transmissões multimídia acrescentam maior grau de sobrecarga em uma rede, principalmente quando se trata das populares redes WLAN, devido à maior probabilidade de interferências. Esta realidade justifica as diversas iniciativas de estudos e pesquisas que procuram desenvolver e propor melhorias em termos de qualidade de serviço e qualidade de vídeo nas transmissões de tráfego multimídia em redes IP. Para validar uma proposta de melhoria é necessário utilizar algum método confiável que permite encontrar a qualidade dos frames de vídeo recebidos pelos espectadores. Neste trabalho analisamos as principais métricas de qualidade de vídeo e propomos a utilização de uma nova métrica sem referência que estima a percepção de borramento de uma imagem, adaptando esta métrica em um ambiente real de transmissão multimídia.

Palavras-chave: qualidade, vídeo, borramento, psnr

1. INTRODUÇÃO

Sabemos que as transmissões de conteúdo multimídia em redes de computadores são muito sensíveis ao estado da rede, principalmente quando nos referimos à qualidade do vídeo percebido pelo usuário. As variáveis de atraso, jitter e perda indicam a real situação do meio de transmissão, portanto, os valores destas métricas exercem determinante influência na percepção da qualidade de um vídeo transmitido (Nunes, 2011).

Para estimar as métricas de QoS (jitter, atraso e perda) existem técnicas bem consolidadas. Muitos utilizam programas, como o *wireshark*, que farejam a rede e coletam os dados durante as transmissões, permitindo realizar os cálculos necessários para aferição do estado da rede. Outro método interessante é a utilização de programas que transmitem datagramas ICMP que capturam o tempo de ida e volta de cada datagrama, de acordo com as condições da rede, tornando possível também o levantamento das variáveis de QoS (Monteiro, 2012).

Para calcular a qualidade dos frames de vídeo transmitidos existem duas maneiras distintas. Através de teste subjetivo baseado na percepção visual e opinião do usuário receptor, e através de teste objetivo que procura estimar o grau de qualidade de um vídeo sem intervenção direta do espectador.

O teste subjetivo é conhecido como MOS (*Mean Opinion Score*) e possui um custo maior de implementação, pois exige espaço físico adequado, recursos tecnológicos específicos, planejamento de tempo, além da necessidade de verificar se as condições físicas dos espectadores são adequadas. Outro agravante é que a avaliação do usuário receptor é realizada analisando a sequência de vídeo como um todo, afinal, seria bastante oneroso avaliar um vídeo frame a frame. Esse método pode provocar algumas falhas de julgamento, onde determinada sequência de vídeo pode ser avaliada como ruim mesmo que apenas uma pequena parte tenha se mostrado incômoda.

A recomendação BT.500-11 da ITU-R apresenta procedimentos específicos para realização de testes subjetivos. Nesta recomendação são definidas características do ambiente, da tela de visualização, das imagens ou frames, do dispositivo utilizado para registro de opinião, dos



espectadores, entre outras definições. A Tabela 1 mostra as pontuações que podem ser atribuídas pelo espectador como opinião de qualidade da imagem (ITU-R, 2002).

Qualidade	Valor da Métrica	Grau do incômodo
Excelente	5	Imperceptível
Boa	4	Perceptível, mas sem incômodo
Regular	3	Leve incômodo
Pobre	2	Incômodo
Ruim	1	Muito incômodo

Tabela 1. Escala de qualidade MOS

Outro meio de avaliar a qualidade dos frames de vídeo transmitidos em uma rede é utilizando testes objetivos. O maior interesse de uma métrica objetiva é ser capaz de substituir testes subjetivos que exigem tempo e meios adequados. No entanto, os testes subjetivos são necessários para validar a eficiência de uma métrica objetiva, correlacionando com o julgamento humano.

Um dos métodos de avaliação objetiva de qualidade de vídeo mais conhecidos é o PSNR (*Peak Signal Noise Ratio*). O PSNR é a relação entre a entrada e a saída de um processo de compressão com perdas, que avalia o quanto a compressão introduziu ruídos na imagem ou frame original. Portanto, para realizar o cálculo de PSNR é necessário realizar a comparação entre os frames do vídeo original e os frames do vídeo transmitido. Este tipo de método é conhecido como métrica de qualidade de vídeo com referência (Huynh-thu, 2012).

A métrica PSNR tem sido bastante utilizada em pesquisas e tem se mostrado útil em muitos trabalhos que visam analisar a qualidade de frames de vídeo. No entanto, devido a algumas particularidades, esta métrica possui dificuldades para relacionar o seu valor com o real estado de uma rede, principalmente quando há maior sobrecarga no meio (Monteiro, 2012). Isso se deve ao fato de que em uma rede com maior tráfego agregado a possibilidade de ocorrer perda de pacotes tende a aumentar. Se existir muita perda na rede, quando a métrica PSNR for comparar o frame original com o frame transmitido, há grande probabilidade da ocorrência de inconsistência entre os frames comparados, traduzindo em um valor de PSNR que pode ser inadequado (Huynh-thu, 2012).

Observando a dificuldade em se determinar coerentemente e de forma viável a qualidade de um vídeo, este trabalho visa apresentar uma nova métrica de determinação desta qualidade sem necessidade de um vídeo original como referência. Propomos uma adaptação da métrica sem referência que estima a percepção de borramento em uma imagem, a fim de utilizar esta métrica em um ambiente de transmissão de vídeo sob demanda. Após uma sequência de experimentos práticos, determinamos o grau de correlação entre esta métrica de borramento e as variáveis de determinação do estado da rede (jitter, atraso e perda).

Na seção 2 descrevemos alguns trabalhos de pesquisa realizados referente à estimação de qualidade de vídeo em transmissões multimídia em redes de computadores. Na seção 3 apresentamos uma descrição detalhada da métrica de estimativa de borramento e suas particularidades. Na seção 4 detalhamos o algoritmo utilizado para estimar o borramento em imagens ou frames de vídeo, bem como descrevemos as adaptações realizadas para utilizar a métrica no contexto de transmissões de vídeo em redes de computadores. Concluimos o trabalho na seção 5 com as considerações finais.

2. REFERENCIAL TEÓRICO

Existem fatores importantes que direcionam os pesquisadores no estudo de propostas que viabilizem a utilização de conteúdo multimídia em redes IP. Destes, podemos destacar como essenciais a maior oferta de banda larga, o aumento da demanda de serviços multimídia na Internet, a evolução dos *codecs* de vídeo e a padronização do protocolo IP.

Logo, diversas pesquisas vêm sendo publicadas com propostas de melhorias que proporcionam maior qualidade de vídeo em ambientes de transmissão de vídeo. Em Kim et al. (2011) é apresentado uma proposta que objetiva melhorar o desempenho fim a fim de transmissões de vídeo em tempo real



no modelo *broadcast* sobre redes sem fio WLAN 802.11 utilizando o protocolo HCCA e compressão de vídeo H.264/SVC. Mackenzie et al. (2009) apresenta um estudo para melhoria da qualidade de serviço em redes WLAN na distribuição de arquivos multimídia. Em Feng et al. (2009), utilizando o protocolo IEEE 802.11e, o autor procura apresentar uma estratégia para oferecer qualidade de serviço em transmissões multimídia sobre redes WLAN.

Para que estas e outras propostas desta natureza sejam validadas, é necessário utilizar alguma metodologia de verificação da qualidade da rede e da qualidade dos vídeos utilizados nas transmissões. A análise da qualidade da rede é realizada através das variáveis jitter, atraso e perda de pacotes, que podem ser aferidas utilizando métodos que permitem monitorar o estado do meio.

Existem também diversas metodologias que permite obter a qualidade do vídeo transmitido. A métrica subjetiva MOS é baseada na opinião do espectador, no entanto, possui custo elevado de implantação, portanto, a maioria das pesquisas fazem uso das métricas objetivas.

Uma das métricas objetivas mais utilizadas é o PSNR. Em Nunes et al. (2011) são apresentadas técnicas de análise de tráfego para aferição de QoS e QoV, sendo que a qualidade do vídeo foi obtida utilizando PSNR. Migliorini et al. (2011) e Vishwanth et al. (2011) avaliam a performance de *streams* de vídeo H.264/SVC sobre redes WiMAX tendo também como métrica de qualidade o PSNR. Forchhammer et al. (2011) e Eden et al. (2007) propõe soluções para estimar o PSNR baseado na decodificação dos frames MPEG após a transmissão, criando metodologias sem referência de estimativa de qualidade.

Mesmo sendo um método útil e bastante usado, a métrica PSNR possui algumas características que não são bem adaptáveis a ambientes de transmissão de vídeo. Monteiro et al. (2012) demonstra que esta métrica tem dificuldades para estimar o impacto que as condições de tráfego na rede exercem sobre a qualidade do vídeo, evidenciando que pode haver algumas incoerências em seu valor. Huynh-Thu et al. (2012) destaca que o PSNR trabalha comparando a imagem original e a imagem transmitida pixel a pixel, e pode ser utilizada para avaliar o desempenho de *codecs* de vídeo diferentes, no entanto, o autor conclui que esta métrica é pouco adequada em ambientes de tempo real, pois falha na representação da percepção humana da imagem.

Baseado nesta realidade, propomos uma adaptação da métrica de estimação de borrimento descrita em Crete (2007) para execução em ambientes de transmissão de vídeo sob demanda, a fim de retornar a qualidade dos frames sem necessidade de referência e com baixo custo de implementação.

3. DESCRIÇÃO DO EFEITO DE BORRAMENTO

O efeito de borrimento em uma imagem é geralmente causado pela perda do conteúdo de alta frequência. Essa perda pode acontecer quando se utiliza técnicas de correção de imperfeições ou ruídos, o que provoca decréscimo nos detalhes, gerando incômodo na percepção visual da imagem. O efeito de borrimento pode ocorrer também quando se tem seqüências de imagens em movimento (*blur motion*), o que caracteriza bem as transmissões de vídeo em um ambiente de rede, já que um vídeo é apenas um conjunto de imagens transmitidas seqüencialmente.

Portanto, a estimativa do grau de borrimento sofrido por uma seqüência de frames pode determinar as condições de qualidade do vídeo, assim como pode demonstrar o atual estado de uma rede através da relação com as métricas de QoS. Portanto, a recíproca é verdadeira se existir um limiar de correlação coerente entre a métrica de borrimento e as métricas de jitter, atraso e perda.

3.1 Considerações sobre a percepção do efeito de borrimento

Para encontrar esta correlação precisamos primeiramente entender melhor como o efeito de borrimento é percebido pelo homem. Em primeiro lugar, quando se compara uma imagem nítida com esta mesma imagem borrada, a percepção humana detecta uma grande diferença em termos de perda de detalhes entre a primeira e a segunda imagem. No entanto, se compararmos uma imagem já borrada e a mesma imagem ainda mais borrada a diferença será pequena entre as duas. Esta realidade é descrita na Figura 1.



Figura 1. da esquerda para direita: frame original, frame transmitido com limitação de 100Kbps, frame transmitido com limitação de 50Kbps

Na realidade, quanto mais se borrar uma imagem, mais os pixels vizinhos irão convergir para o mesmo nível de cinza. Se borrarmos uma imagem nítida, os níveis de cinza dos pixels vizinhos irão se alterar com uma variação maior. Porém, se borrarmos uma imagem já borrada, os níveis de cinza de pixels vizinhos ainda vão se alterar, mas com menor variação. A idéia principal da estimativa de borrimento é analisar o comportamento da variação dos pixels vizinhos para determinar o grau de borrimento sofrido por um frame em comparação com um frame anterior (Crete, 2007).

Outro detalhe relevante é que ao observar uma imagem que contenha uma pequena parte borrada sobre uma área homogênea, a percepção humana identifica que aquela imagem não está nítida, mesmo se apenas uma menor parte estiver borrada. Por esta razão, na análise da variação dos pixels vizinhos, a métrica de borrimento considera apenas os pixels que mudaram após o processo que provoca o incômodo na imagem (Crete, 2007).

A última consideração é feita em relação a imagens que apresentam borrimento de movimento, típico de transmissões de vídeo em ambientes de rede. Ao extrair uma imagem a partir de uma sequência em movimento podemos observar que em uma imagem borrada em uma direção, o olho humano percebe o efeito de borrimento sobre esta direção, mas não percebe a nitidez sobre as outras direções. Por esta razão, a métrica calcula o borrimento nas direções horizontais e verticais selecionando o valor de borrimento mais incômodo como valor de borrimento final (Crete, 2007).

3.2 Descrição do princípio de estimação de borrimento

Considerando os fenômenos explicados na seção anterior, é possível então quantificar o incômodo causado pelo efeito de borrimento em um frame de vídeo. A primeira etapa de cálculo da métrica consiste em determinar as variações de intensidade entre os pixels vizinhos de um frame inicial. Em seguida, calculam-se as variações entre os pixels vizinhos do frame seguinte. Então, a comparação entre estas variações de intensidade, do frame inicial e do frame seguinte, permite avaliar o incômodo do efeito de borrimento entre os dois frames.

Assim, uma maior variação entre o frame inicial e o frame seguinte significa que o frame inicial estava nítido. Porém, se existir uma menor variação entre o frame inicial e o frame seguinte significa que o frame inicial já estava borrado. Esta descrição é resumida na Figura 2.

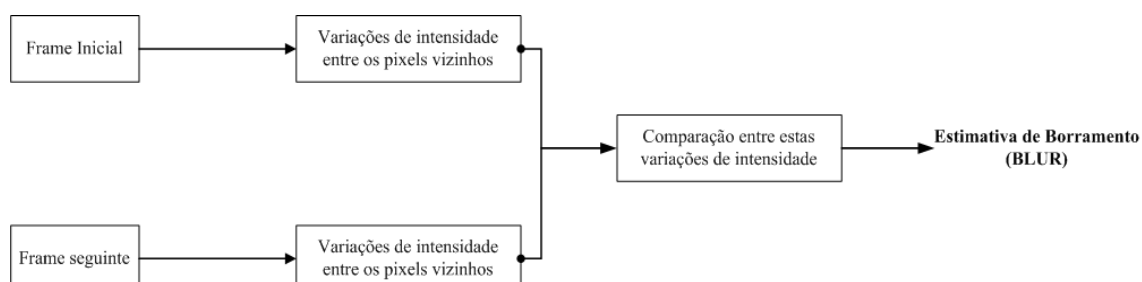


Figura 2. Fluxo simplificado do princípio de estimação de borrimento

É importante destacar que a métrica utiliza os frames que foram transmitidos do servidor para o cliente, não sendo necessária a utilização do vídeo original para analisar a qualidade do vídeo, ao contrário da métrica PSNR que necessita da referência dos frames originais. Portanto, esta métrica de estimativa de borramento é caracterizada como um método de aferição de qualidade sem referência. Esta característica elimina o problema de inconsistência que pode ocorrer com o uso do PSNR, que pode ser causado por perdas de frames na transmissão, o que provoca em muitos casos a perda da referência no cálculo da qualidade de determinados frames (Crete, 2007).

A proposta visa, portanto, adaptar a métrica de estimativa de borramento para utilização em um ambiente de transmissão de vídeo sob demanda, bem como analisar detalhadamente sua eficiência de atuação através da correlação entre as variáveis de QoS, que determinam o estado da rede.

3.3 Descrição das equações de determinação de borramento

Sabe-se que a nitidez de uma imagem é determinada pelo seu componente de cinza. Esta hipótese, que é verificada com testes subjetivos, justifica que a estimativa do incômodo de borramento pode ser verificada apenas através do componente de luminância (Boult, 1999). A Figura 3 descreve a sequência geral da execução do algoritmo e as referentes equações.

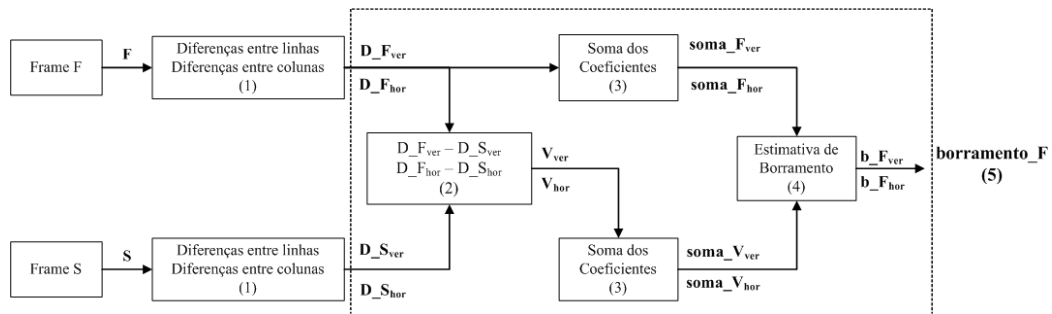


Figura 3. Fluxo do algoritmo de estimativa de borramento com referência às equações

Seja F um componente de luminância de um frame de vídeo do tamanho de pixels $m \times n$. Para estimar o incômodo do borramento é necessário estimar a variação dos pixels vizinhos do frame F e do frame que vem imediatamente após ao frame F na sequência do vídeo. Este frame seguinte é denominado nas equações de frame S . Portanto, a fim de estudar as variações dos pixels vizinhos dos dois frames, a métrica computa a diferença absoluta em cada frame na horizontal e na vertical, gerando os valores ilustrados por $D_{F_{ver}}$, $D_{F_{hor}}$, $D_{S_{ver}}$ e $D_{S_{hor}}$ como mostra em (1):

$$\begin{aligned}
 D_{F_{ver}}(i, j) &= Abs(F(i, j) - F(i - 1, j)) \rightarrow \text{para } i = 1 \text{ até } m - 1, \text{ para } j = 0 \text{ até } n - 1 \\
 D_{F_{hor}}(i, j) &= Abs(F(i, j) - F(i, j - 1)) \rightarrow \text{para } j = 1 \text{ até } n - 1, \text{ para } i = 0 \text{ até } m - 1 \\
 D_{S_{ver}}(i, j) &= Abs(S(i, j) - S(i - 1, j)) \rightarrow \text{para } i = 1 \text{ até } m - 1, \text{ para } j = 0 \text{ até } n - 1 \\
 D_{S_{hor}}(i, j) &= Abs(S(i, j) - S(i, j - 1)) \rightarrow \text{para } j = 1 \text{ até } n - 1, \text{ para } i = 0 \text{ até } m - 1
 \end{aligned} \tag{1}$$

Como descrito na seção anterior, é preciso analisar a variação dos pixels vizinhos. Se esta variação entre os dois frames for alta, o frame inicial possui maior grau de nitidez. Se a variação for pequena o frame inicial já apresenta um estado de borramento. A variação é obtida pela equação (2):

$$\begin{aligned}
 V_{ver} &= Max(0, D_{F_{ver}}(i, j) - D_{S_{ver}}(i, j)) \rightarrow \text{para } i = 1 \text{ até } m - 1, \text{ para } j = 1 \text{ até } n - 1 \\
 V_{hor} &= Max(0, D_{F_{hor}}(i, j) - D_{S_{hor}}(i, j)) \rightarrow \text{para } i = 1 \text{ até } m - 1, \text{ para } j = 1 \text{ até } n - 1
 \end{aligned} \tag{2}$$

Então, a fim de comparar as variações do frame inicial, calculamos a soma dos coeficientes de $D_{F_{ver}}$, $D_{F_{hor}}$, $D_{S_{ver}}$, $D_{S_{hor}}$ como segue na equação (3):

$$\begin{aligned}
 soma_F_{ver} &= \sum_{i,j=1}^{m-1,n-1} D_F_{ver}(i,j) & soma_F_{hor} &= \sum_{i,j=1}^{m-1,n-1} D_F_{hor}(i,j) \\
 soma_V_{ver} &= \sum_{i,j=1}^{m-1,n-1} D_V_{ver}(i,j) & soma_V_{hor} &= \sum_{i,j=1}^{m-1,n-1} D_V_{hor}(i,j)
 \end{aligned} \tag{3}$$

Finalmente, o resultado é normalizado dentro do intervalo definido entre 0 e 1 como mostra a equação (4):

$$b_F_{ver} = \frac{soma_F_{ver} - soma_V_{ver}}{soma_F_{ver}} \quad b_F_{hor} = \frac{soma_F_{hor} - soma_V_{hor}}{soma_F_{hor}} \tag{4}$$

Então, selecionamos o borramento mais incômodo entre a vertical e a horizontal como o valor de borramento final através da equação (5).

$$borramento_F = Max(b_F_{ver}, b_F_{hor}) \tag{5}$$

Portanto, a métrica obtém uma estimacão sem referência da percepção de borramento variando de 0 a 1, que são respectivamente a melhor e a pior qualidade em termos de percepção de borramento.

4. ALGORITMO DE ESTIMACÃO DE BORRAMENTO

Baseado nos conceitos e equações descritas foi possível adaptar a métrica em forma de algoritmo que execute dentro de um ambiente de transmissão de vídeo sob demanda, retornando o valor de borramento de cada frame, permitindo a obtenção do limiar de qualidade do vídeo como um todo.

Para tanto, utilizamos a linguagem computacional GNU Octave, desenvolvida para computação matemática. O Octave se destaca por ser um software livre sob os termos da licença GPL e que possui compatibilidade com o MATLAB tendo grande número de funções semelhantes. Inicialmente, utilizamos uma implementação da percepção de borramento desenvolvida em MATLAB descrita em Crete (2007) e adaptamos a sua execução em Octave, como mostrado no código abaixo:

```

01 function blur = blurMetric(imagem_original, imagem_borrada, mediablur)
02     f = imread(imagem_original);
03     b = imread(imagem_borrada);
04     [y x] = size(f);
05     D_F_Ver = abs(f(:,1:x-1) - f(:,2:x));
06     D_F_Hor = abs(f(1:y-1,:) - f(2:y,:));
07     D_S_Ver = abs(b(:,1:x-1) - b(:,2:x));
08     D_S_Hor = abs(b(1:y-1,:) - b(2:y,:));
09     T_Ver = D_F_Ver - D_S_Ver;
10     T_Hor = D_F_Hor - D_S_Hor;
11     V_Ver = max(0, T_Ver);
12     V_Hor = max(0, T_Hor);
13     S_D_Ver = sum(sum(D_F_Ver(2:y-1,2:x-1)));
14     S_D_Hor = sum(sum(D_F_Hor(2:y-1,2:x-1)));
15     S_V_Ver = sum(sum(V_Ver(2:y-1,2:x-1)));
16     S_V_Hor = sum(sum(V_Hor(2:y-1,2:x-1)));
17     blur_F_Ver = (S_D_Ver - S_V_Ver) / S_D_Ver;
18     blur_F_Hor = (S_D_Hor - S_V_Hor) / S_D_Hor;
19     blur = max(blur_F_Ver, blur_F_Hor);
20     blurstr = num2str(blur);
21     mediablur = blurstr;
22 endfunction

```



Esta função retorna o valor de borramento comparando apenas a imagem original com a imagem borrada. Como em nosso modelo desejamos executar em um ambiente de transmissão de vídeo, foi necessário adaptar a métrica para funcionar dentro de uma estrutura de repetição que chamasse a função para cada frame recebido pelo cliente. Em primeiro lugar, inicializamos o *vlc server* como servidor multimídia em uma ambiente qualquer de rede.

Após a transmissão do vídeo na rede, precisamos transformar o vídeo *mp4* recebido pelo usuário em imagens *png* para então executarmos o programa que calcula o borramento. Utilizamos novamente o *vlc* para esta tarefa utilizando o seguinte comando:

```
vlc <video_recebido>.mp4 --rate=1 --video-filter=scene --vout=dummy --scene-format=png --scene-ratio=1 --scene-prefix=video --scene-path=video/vlc://quit
```

Através deste comando é gerado em formato de imagens os frames do vídeo transmitido. Com estes frames armazenados em uma pasta podemos alterar o algoritmo de percepção de borramento a fim de obter a qualidade de cada frame. Deste modo propomos um algoritmo que percorre esta pasta e a cada frame encontrado chamamos a função que calcula o borramento comparando o frame atual com o frame seguinte. Após o cálculo, armazenamos o valor obtido em um arquivo *txt* para posterior análise. Estes procedimentos são descritos no algoritmo abaixo:

```
01  y = 0;
02  q = 1000; % quantidade de frames do vídeo
03  for i=0:q
04      seq = num2str(y);
05      imgborrada = strcat (url, seq, ".png");
06      y = y + 1;
07      seq2 = num2str(y);
08      imgnormal = strcat (url, seq2, ".png")
09      try
10          mediablur = blurMetric(imgnormal, imgborrada, mediablur);
11      catch
12          break;
13      end_try_catch
14  endfor
15  arq = strcat ("blurmetric.txt");
16  filename = strcat (pasta, arq);
17  media = num2str(mean(mediablur));
18  fid = fopen (filename, "a");
19  fputs (fid, "\n");
20  fputs (fid, ind);
21  fputs (fid, " - ");
22  fputs (fid, media);
23  fclose (fid);
```

O algoritmo gera um arquivo com todos os valores da métrica de borramento para cada frame. Com base nos resultados encontrados é possível verificar a qualidade do vídeo com coerência, sendo uma métrica que pode ser utilizada como alternativa à métrica PSNR.

5. CONCLUSÃO

Evidenciando a necessidade de se ter uma alternativa a métrica PSNR para levantamento da qualidade de vídeo, propomos uma nova métrica que estima o grau de borramento que um frame sofre durante o processo de transmissão multimídia em uma rede de computadores.

Destacamos que a percepção de borramento da visão humana possui algumas características que precisam ser emuladas pelo algoritmo, onde a diferença entre uma imagem nítida e uma imagem borrada é bem perceptível pelo olho humano, no entanto, a diferença entre uma imagem borrada e uma



imagem ainda mais borrada não é tão aparente. Outra característica percebida é que o grau de borramento pode ser definido apenas pelo fator de luminância.

Portanto, a função de borramento apresentou baixo custo de implementação e execução, visto que a programação utiliza métodos matemáticos simples de cálculo de absoluto, soma, média e máximo. Outro fator importante é que a métrica não precisa necessariamente analisar cada pixel da imagem, já que a percepção humana de borramento define a qualidade de toda a imagem baseado em apenas alguns trechos que possam estar desfocados. A não necessidade de referência também colabora para o baixo custo desta métrica.

Concluimos que a estimativa de borramento se apresentou uma metodologia eficiente para demonstrar a qualidade de um vídeo transmitido em uma rede de computadores, podendo ser utilizada como alternativa eficiente em relação à métrica PSNR.

REFERÊNCIAS

- BOULT, B. E.; CHIANG, M. C. **Local Blur Estimation and Super-Resolution**. IEEE, Computer Society Conference on Computer Vision and Pattern Recognition, pg. 821-826 – 1997.
- CRETE, F.; DOLMIERE, T.; LADRET, P.; NICOLAS, M. **The Blur Effect: Perpection and Estimation with a New No-Reference Perceptual Blur Metric**. SPIE Electronic Imaging Symposium Conf Human Vision and Electronic Imaging – 2007.
- EDEN, A. **No-reference estimation of the coding PSNR for H.264-coded sequences**. IEEE Trans. Consumer Elec. 53, pg. 667-674 – 2007.
- FENG, Z.; WEN, G.; ZOU, Z.; GAO, F. **Red-txop Scheme for Video Trasmission In IEEE 802.11E EDCA WLAN**. IEEE Communications Society – 2009.
- FORCHHAMMER, S.; Li, H.; ANDERSEN, J. D. **No-reference analysis of decoded MPEG images for PSNR estimation and post-processing**. Jornal of Visual Communication and Image Representation, pg. 313-324 – 2011.
- HUYNH-THU, Q.; GHANBARI, M. **The accuracy of PSNR in predicting video quality for different video scenes and frame rates**. Telecommunication Systems – Vol. 49 Pg. 35-48 – 2012.
- KIM, Y. H.; PARK, H. R.; SUK, J. B. **A Scheme of H.264/SVC Video Broadcasting Over IEEE 802.11 WLAN**. IEEE Communications Society – 2011.
- MACKENZIE, R.; HANDS, D.; O'FARREL, T. **QoS of Video Delivered over 802.11e WLANs**. IEEE Communications Society – 2009.
- MIGLIORINI, D.; MINGOZZI, E.; VALLATI, C. **Performance evaluation of H.264/SVC video streaming over mobile WiMAX**. Computer Networks – Vol. 55 Pg. 3578–3591 – 2011.
- MONTEIRO, C. C. **Um ambiente para apoio à integração de redes sem fio heterogêneas**. Tese (Doutorado em Engenharia Elétrica) - Universidade de Brasília, Brasília – 2012.
- NUNES, J.; MONTEIRO, C. **Recepção de Vídeo em Redes 802.11: Utilizando Técnicas de Análise de Tráfego para Aferição de QoS e QoE**. VI Congresso de Pesquisa e Inovação da Rede Norte Nordeste de Educação Tecnológica, Natal – 2011.
- VISHWANATH, A.; DUTTA, P.; CHETLUR, M.; GUPTA, P.; KALYANARAMAN, S.; GHOSH, A. **Perspectives on Quality of Experience for Video Streaming over WiMAX**. Mobile Computing and Communications Review, vol. 13, n. 4 – 2011.