



UNIVERSIDADE ESTADUAL DE CAMPINAS

FACULDADE DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO

DEPARTAMENTO DE ENGENHARIA DE COMPUTAÇÃO E AUTOMAÇÃO INDUSTRIAL

Dissertação de Mestrado

Avaliação Objetiva de Codecs de Vídeo

Rangel Arthur

Orientador: Prof. Dr. Roberto de Alencar Lotufo

DISSERTAÇÃO DE MESTRADO
UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO
DEPARTAMENTO DE ENGENHARIA DE COMPUTAÇÃO E AUTOMAÇÃO INDUSTRIAL

Avaliação Objetiva de Codecs de Vídeo

Autor: **Rangel Arthur**

Orientador: **Prof. Dr. Roberto de Alencar Lotufo**

Banca examinadora:

Prof. Dr. Roberto de Alencar Lotufo – FEEC / UNICAMP

Prof. Dr. Yuzo Iano – FEEC / UNICAMP

Prof. Dr. Fernando José Von Zuben – FEEC / UNICAMP

Prof. Dr. Alexandre Xavier Falcão – IC / UNICAMP

Dissertação submetida à Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas, para preenchimento dos pré-requisitos parciais para obtenção do Título de Mestre em Engenharia Elétrica. Área de concentração: Engenharia de Computação.

17 de abril de 2002

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

Ar77a	<p>Arthur, Rangel</p> <p>Avaliação objetiva de codecs de vídeo / Rangel Arthur.-- Campinas, SP: [s.n.], 2002.</p> <p>Orientador: Roberto de Alencar Lotufo. Dissertação (mestrado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.</p> <p>1. Processamento de imagens. 2. Vídeodigital. I. Lotufo, Roberto de Alencar. II. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. III. Título.</p>
-------	---

À minha família

Agradecimentos

Ao Prof. Dr. Roberto de Alencar Lotufo, pela oportunidade de enaltecer meus conhecimentos através de sua orientação sempre perspicaz ao longo deste trabalho.

Aos meus amados pais Florevaldo e Zilda, ao meu irmão Wagner e à Angélica pelo incessante apoio e incentivo em todos os caminhos que percorri.

Aos meus amigos de república Alexandre, Luiz, Franklin, Maurício, Alex e André, pelo apoio nos momentos de dificuldades.

Aos sempre companheiros da Fundação CPqD, que me ajudaram e ajudam desde minha chegada a Campinas, em especial a Antônio Cláudio França Pessoa, pelas orientações e sugestões na elaboração deste trabalho.

Aos meus amigos do LCA, que sempre acrescentaram algo em minha vida.

Ao Prof. Dr. Fernando José Von Zuben, pela inestimável ajuda para o processo de otimização de descritores.

Ao Prof. Dr. Ariovaldo V. Garcia , Coordenador de Pós-Graduação da FEEC, e ao Prof. Dr. José Cláudio Geromel , Pró-Reitor de Pós-Graduação, pelo apoio ao agilizar o processo burocrático no final do curso.

À FAPESP e à FAEP, pelo indispensável apoio financeiro durante a realização deste trabalho.

A todos que me apoiaram nesta jornada, meu muito obrigado!

Resumo

Este trabalho baseia-se no estudo e desenvolvimento de metodologias para a avaliação objetiva de sistemas de transmissão unidirecional de vídeo. Particularmente, são tratados os sistemas de compressão de vídeo (denominados *codecs*) segundo o padrão MPEG, mas que podem ser aplicados à avaliação de outros tipos de sistema. O desenvolvimento de metodologias para avaliação objetiva de *codecs* de vídeo tem como meta substituir a avaliação subjetiva, feita com grupos controlados de observadores e que exige altos recursos. Dentro deste contexto, procuram-se métodos de avaliação objetiva que se aproximem da qualidade dos resultados encontrados para a avaliação subjetiva, e que reduzam consideravelmente os custos do processo. São apresentados métodos que podem ser aplicados em sistemas de avaliação com “Referência Total” e “Referência Reduzida”. Os algoritmos propostos são aplicados a um conjunto de cenas de vídeo MPEG-2 e cada parâmetro de qualidade é calculado. A estimativa da qualidade é realizada a partir de uma função interpoladora utilizando redes neurais, treinada com uma base de dados subjetiva das seqüências trabalhadas. Os resultados alcançados mostram valores elevados de correlação, os quais podem ser tomados como uma medida de qualidade dos métodos propostos.

Abstract

This work is based on the study and development of methodologies for objective evaluation of video unidirectional transmission systems. Particularly, the video compression systems (called codecs) are worked according to MPEG-2 standard, but they can be applied to evaluate other compression systems. The development of methodologies for video codecs assessment aims at replacing the subjective assessment, usually implemented with the help of a group of observers, demanding high costs. In this context, methods of objective evaluation are investigated in order to produce results close to the ones given by subjective evaluation, and also reducing considerably the cost of the process. The presented methods can be applied to two reference systems: full-reference and reduced-reference. The algorithms proposed are processed to a set of MPEG-2 video codec scenes and each quality estimator is computed. The quality estimative is done through an interpolation function using neural networks, trained with a subjective base of the worked sequences. The achieved results show high correlation values, demonstrating the quality of the proposed methods.

Sumário

1 Introdução	7
2 Codificação de Vídeo Digital	11
2.1 Esquema codificação-transmissão-decodificação	12
2.2 Fundamentos de compressão de imagens.....	13
2.3 Introdução ao MPEG2	16
2.4 Degradações.....	20
2.4.1 Blocação.....	21
2.4.2 “Blurring”	21
2.4.3 “Edge Business”	22
2.4.4 Ruído	22
2.4.5 “Jerkiness”.....	23
2.4.6 Blocos Errados	24
2.5 Descritores	24
3 Fundamentação Teórica	27
3.1 Sistema Visual Humano	29
3.1.1 Resposta em frequência Espacial de Visibilidade.....	29
3.1.2 Resposta em frequência de visibilidade dependente do brilho	30
3.1.3 Dependência visual sensitiva ao brilho	32
3.2 Fotorreceptores, Percepção de Cor e Conversão de Espaço de Cor	32
3.3 Avaliação Subjetiva	34
3.4 Avaliação Objetiva	35
3.4.1 Avaliação com Referência Total.....	36
3.4.2 Avaliação com Referência Reduzida	39
3.5 Parâmetros Objetivos vs. Parâmetros Subjetivos	41
3.6 Escala JND	42
3.7 Histórico do VQEG	46
3.8 Propostas de Avaliação (Grupos do VQEG).....	48
3.8.1 CPqD.....	48
3.8.2 Tektronix/Sarnoff.....	52
3.8.3 NHK	54
3.8.4 KDD	56
3.8.5 NASA.....	58
3.8.6 KPN.....	60
3.8.7 NTIA	61
3.8.8 PSNR usando Wavelet	64

4 Medidas Objetivas	67
4.1 Descritores implementados	68
4.1.1 PSNR - Pico da Relação Sinal Ruído	69
4.1.2 Regiões planas	72
4.1.3 Dimensão fractal	74
4.1.4 Descritor de Blocagem	76
4.1.5 Descritores de Fourier	80
4.1.6 Descritor de Sobel	83
4.1.7 Contraste de nível de cinza e Contraste Colorido.....	86
4.1.8 Histograma.....	89
4.1.9 Variância e Vida Média	92
4.1.10 Entropia	94
4.2 Métodos de comparação	96
4.3 Modelo de regressão por busca de projeção	97
5 Experimentos	103
5.1 Base de Dados	104
5.2 Descrição do Experimento	107
5.3 Resultados	110
6 Conclusões	119
7 Apêndice	125
8 Bibliografia	155

Índice das Figuras

Figura 2.1- Diagrama de blocos de um sistema de comunicação genérico	13
Figura 2.2- Estrutura de um codificador de vídeo.....	14
Figura 2.3- Estrutura de um decodificador de vídeo.....	14
Figura 2.4- Estrutura de um <i>codec</i> MPEG-2: codificador	16
Figura 2.5- Estrutura de um <i>codec</i> MPEG-2: decodificador.....	17
Figura 2.6- Imagens que exemplificam os tipos de degradações.....	23
Figura 2.7- Exemplo de efeito de informação espacial.....	25
Figura 2.8- Exemplo de efeito de informação temporal	26
Figura 3.1- Resposta em frequência espaço-temporal de visibilidade.....	30
Figura 3.2- Resposta em frequência espacial de visibilidade	31
Figura 3.3- Resposta em frequência temporal de visibilidade	31
Figura 3.4- Limites de percepção de ruído aleatório em um monitor de TV....	32
Figura 3.5- Processo de avaliação usando “Referência Total”	36
Figura 3.6- Ilustração de deslocamento de área de vídeo ativo	38
Figura 3.7- Processo de avaliação usando referência reduzida.....	39
Figura 3.8- Hierarquia de desenvolvimento de parâmetros objetivos	42
Figura 3.9- Fluxograma para o método EASE.....	44
Figura 3.10- Função por partes estimada da função escala.....	45
Figura 3.11- Espaço de medida de qualidade e pontos de teste	46
Figura 3.12- Segmentação por regiões	50

Figura 3.13- Fluxo do modelo para emular a visão humana	53
Figure 3.14- Estrutura do sistema de avaliação	54
Figure 3.15- Estrutura do sistema de avaliação	55
Figura 3.16- Modelo “três-camadas” para um sinal de vídeo	56
Figura 3.17- Esquema de avaliação	57
Figura 3.18- Passos para o processamento do DVQ	59
Figura 3.19- Esquema do modelo proposto.....	60
Figura 3.21- Filtros para detecção de bordas.....	63
Figura 3.22- Procedimento de avaliação objetiva usando wavelet.....	64
Figura 3.23- Decomposição três-escalas de wavelet.....	65
Figura 4.1- Imagens para cálculo de PSNR.....	72
Figura 4.2- Exemplo do descritor de regiões planas	74
Figura 4.3- Análise de imagens pela variância das colunas	76
Figura 4.4- Detecção de blocagem em imagens reais	79
Figura 4.5- Regiões para análise por descritores de Fourier	81
Figura 4.6- Exemplo de extração do descritor de Fourier	82
Figura 4.7- Aplicação do Descritor de Sobel.....	84
Figura 4.8- Exemplo prático do Descritor de Sobel com imagem real	85
Figura 4.9- Exemplo de contraste de nível de cinza de imagens reais	87
Figura 4.10- Exemplo de contraste colorido.....	89
Figura 4.11- Histogramas dos quatro tipos básicos de imagens.....	90
Figura 4.12- Histograma usado como descritor.....	92

Figura 4.13- Análise de imagens pela variância das colunas	94
Figura 4.14- Imagens e valores de entropia (imagem <i>Ball</i>)	95
Figura 4.15- Imagens e valores de entropia (imagem <i>Susy</i>)	96
Figura 4.16- Modelo de regressão por busca de projeção.....	99
Figura 5.1- Quadros de vídeo das seqüências trabalhadas.....	105
Figura 5.2 – Diagrama de blocos do Experimento.....	109
Figura 5.3 – Curvas com descritores finais calculados e dados subjetivos usando todas as imagens (Referência Total)	111
Figura 5.4 - Curvas com descritores calculados e dados subjetivos usando a imagem restante (Referência Total)	114
Figura 5.5 - Curvas com descritores extraídos e dados subjetivos usando a imagem restante (Referência Reduzida).....	117

Capítulo 1

Introdução

Um dos problemas que surge com a introdução de sistemas de compressão de vídeo digital nos mais diversos serviços (distribuições primária e secundária de televisão digital, vídeo sob demanda, videofone, videoconferência, etc) é o de avaliação de qualidade. A utilização de sistemas padronizados não garante homogeneidade de desempenho, dada a estrutura aberta desses padrões. Por outro lado, metodologias de avaliação são importantes ferramentas para a configuração adequada desses sistemas numa fase de grande diversidade de aplicações, que podem envolver cascata de sistemas, conversão entre formatos e utilização de configurações escalonáveis.

Hoje é possível prover recepção doméstica de diversos canais de televisão digitais, graças ao desenvolvimento de técnicas de transmissão bastante efetivas e à padronização de técnicas de compressão de áudio e vídeo. Embora a factibilidade destes novos serviços já esteja demonstrada, a maior questão neste cenário ainda permanece e consiste na satisfação do usuário com este tipo de serviço. Diversos fatores irão determinar seu sucesso, incluindo o preço inicial do serviço, interesse do usuário no conteúdo do canal e competitividade entre diferentes operadores. Costuma-se pensar que apenas o que se transmite e a que preço é que são fundamentais, e não aspectos técnicos envolvendo a qualidade da imagem. Entretanto, embora o fator econômico tenha um peso considerável, a importância da qualidade da imagem não deve ser menosprezada, já que o usuário certamente irá levá-la em consideração ao fazer sua escolha pelos novos serviços disponíveis.

A economia em termos de meios de transmissão e capacidade de armazenamento, gerada pelo uso de compressão, depende da quantidade de informação presente no sinal original (não comprimido) e de quanto da qualidade original se está disposto a sacrificar. As degradações aparecem quando a quantidade de informação presente no vídeo é maior do que a capacidade de transmissão do canal. Os usuários de alguns tipos de serviço, entretanto, se dispõem a sacrificar parte da qualidade para economizar nos custos. Os equipamentos de codificação conseguem reduzir o número de bits para a representação da imagem, extraíndo a redundância da imagem, tanto quanto possível. Se uma redução adicional de bits é necessária, e isto é sempre verdade em aplicações para entretenimento, então alguma distorção deverá ser introduzida na imagem.

As degradações no vídeo digital são dificilmente mensuráveis objetivamente, embora sejam quase sempre óbvias para observadores humanos. Esta dificuldade deve-se, em parte, ao fato dos algoritmos de compressão introduzirem degradações numa escala que depende fortemente do conteúdo de detalhes e movimento das cenas. Por outro lado, a percepção dessas distorções também é altamente dependente deste contexto. Assim sendo, os métodos tradicionais de avaliação, baseados em sinais estáticos, são inadequados para quantificar a qualidade desses sistemas.

A avaliação de qualidade de vídeo digital tem sido objeto de estudo de diversos grupos de pesquisa, principalmente os pertencentes ao Grupo de Experts de Qualidade de Vídeo (VQEG) [1], que disputam atualmente o processo internacional de padronização para avaliação de vídeo digital. O grupo de Processamento de Sinais Audiovisuais da Fundação Centro de Pesquisa e Desenvolvimento em Telecomunicações (CPqD) participa deste processo de padronização, alcançando excelentes resultados na primeira fase de testes.

Este trabalho trata, no Capítulo 2, de uma descrição do esquema de codificação e decodificação em um sistema de transmissão digital, citando os tipos de compressão e os fundamentos para comprimir imagens. É dada também uma atenção especial ao processo de criação do formato MPEG-2 e são destacados os tipos de degradações mais comuns presentes em imagens decodificadas.

Uma metodologia para análise de qualidade de vídeo é mostrada no Capítulo 3, apresentando o comportamento do sistema visual humano (HVS) e os procedimentos básicos para uma avaliação subjetiva. São abordados os tipos de avaliação objetiva e suas aplicações para uma possível aplicação “Em-Laboratório” e “Em-Serviço”. A avaliação “Em-Laboratório”, ou com “Referência Total”, trabalha com uma grande quantidade de dados como referência, e só pode ser realizada em centros especializados. A avaliação “Em-Serviço”, ou com “Referência Reduzida”, permite o monitoramento em tempo real do codificador, permitindo ao usuário acompanhar a qualidade de serviço a todo momento. As principais técnicas propostas pelos grupos do VQEG em sua fase inicial são apresentadas, enfatizando-se a proposta do CPqD, que utiliza um processo de segmentação de regiões planas.

A principal contribuição deste trabalho refere-se às propostas de avaliação de qualidade aplicáveis a Referência Reduzida. A maioria dos métodos usados para Referência Total é baseada em técnicas presentes na literatura atual. Além disso, as técnicas de avaliação propostas utilizam somente as informações espaciais das seqüências de vídeo, não trabalhando, desta forma, com degradações temporais.

Apresenta-se, no Capítulo 4, um conjunto de métodos que podem ser aplicáveis ao processo de padronização de qualidade. Cada função proposta é fundamentada teoricamente, juntamente com as possíveis técnicas de comparação com bases subjetivas,

que fornecem subsídios para se medir a eficácia das implementações. Técnicas de treinamento para ponderar os descritores de qualidade são utilizadas para que sejam alcançados valores os mais correlacionados possíveis com as bases subjetivas.

No Capítulo 5, são realizadas medidas com as funções Matlab implementadas e que formam uma “Caixa de Ferramentas” para análise de qualidade de vídeo. Um conjunto de imagens degradadas com diferentes tipos de degradação é usado, e os resultados de cada função são comparados com uma base subjetiva fornecida pela Fundação CPqD. As conclusões obtidas são apresentadas no Capítulo 6.

Apesar da grande quantidade de propostas presentes na literatura e do aumento considerável dos recursos computacionais disponíveis atualmente, não se estabeleceram padrões abrangentes capazes de prever a qualidade subjetiva de sistemas digitais de transmissão de vídeo. Os *codecs* de vídeo representam os elementos-chave destes sistemas, onde o sinal de vídeo é comprimido para possibilitar sua transmissão. Este trabalho baseia-se no estudo e desenvolvimento de metodologias para a avaliação objetiva destes *codecs* de vídeo em processos com referência total e reduzida.

Capítulo 2

Codificação de Vídeo Digital

Neste capítulo, são descritos os principais elementos de um sistema de transmissão de vídeo digital, enfatizando-se o funcionamento dos equipamentos para compressão, denominados codificadores de vídeo. É importante obter a maior quantidade de informação possível sobre o equipamento para que se possa propor métodos para sua avaliação. São apresentadas também as ferramentas de codificação do consagrado formato MPEG-2, “Motion Picture Group”, utilizado neste trabalho.

Os tipos de degradação são descritos neste capítulo juntamente com alguns exemplos práticos. Cada tipo de degradação possui seus detalhes ou peculiaridades visuais, que podem ser fundamentais durante o processo de avaliação. Esta análise é imprescindível

para se verificar o potencial de aplicações e adquirir sensibilidade sobre os problemas e dificuldades na implementação desta tecnologia.

2.1 Esquema codificação-transmissão-decodificação

Os processos de digitalização e compressão de imagens oferecem muitas vantagens. A digitalização permite maior flexibilidade na manipulação dos dados e enriquecimento das possibilidades de tratamento do sinal. Os circuitos são mais estáveis e confiáveis, a imunidade ao ruído é maior e o transporte de sinais de áudio, vídeo e dados pode ser feito conjuntamente através de multiplexagem. Além disso, há a possibilidade de detecção e correção de erros pela utilização de códigos e melhores alternativas de criptografia e implementação de acesso condicional. A TV Digital adapta-se mais facilmente a automação, pois pode ser controlada por meio de computadores e, uma vez que carrega as informações dos componentes de cor separadamente, facilita também a troca de programas entre países.

A digitalização de imagens permite um aumento da robustez do sistema frente às distorções introduzidas pelo canal de transmissão e à integração de mídias e serviços. Além disso, em sua forma digital as imagens têm melhor qualidade, a manipulação e edição são mais fáceis e, quando associadas a técnicas de compressão, pode-se obter um aumento da eficiência espectral.

Compressão de imagens é o processo de redução do número de bits requeridos para se descrever uma imagem ou sequência de imagens, garantindo um nível de fidelidade em relação às imagens originais que depende fortemente do serviço e da aplicação. Estas técnicas sofreram considerável avanço na última década, motivada pela redução de custos que se pode obter por economia dos meios de transmissão e diminuição dos requisitos de armazenamento. Existem duas categorias de compressão. A compressão sem perdas, ou compactação, na qual o sinal reconstruído é exatamente igual ao sinal original, e a compressão com perdas, na qual o sinal reconstruído pode ser diferente do sinal original, mantido, porém, certo nível de fidelidade objetiva ou subjetiva.

Alguns aspectos da tecnologia de compressão de vídeo que podem implicar em perda de qualidade devem ser levados em consideração. O vídeo comprimido é fortemente susceptível a erros de canal, aumentando os requisitos por códigos corretores de erros, algoritmos de compressão mais robustos e técnicas de cancelamento eficientes. A compressão dificulta a concatenação de “streams” de vídeo de diversas fontes em tempo real, como por exemplo, a inserção de comerciais. A codificação e o empacotamento do vídeo para a transmissão implicam em um atraso que pode ser crítico dependendo da aplicação. Além disso, em aplicações em que há cascadeamento de diversas etapas de compressão é preciso cuidar para que os estágios iniciais sejam comprimidos com alta qualidade, como forma de garantir que não haja excessiva degradação de qualidade para o usuário.

Um sistema de comunicação digital pode ser representado genericamente pelo diagrama de blocos da Figura 2.1.

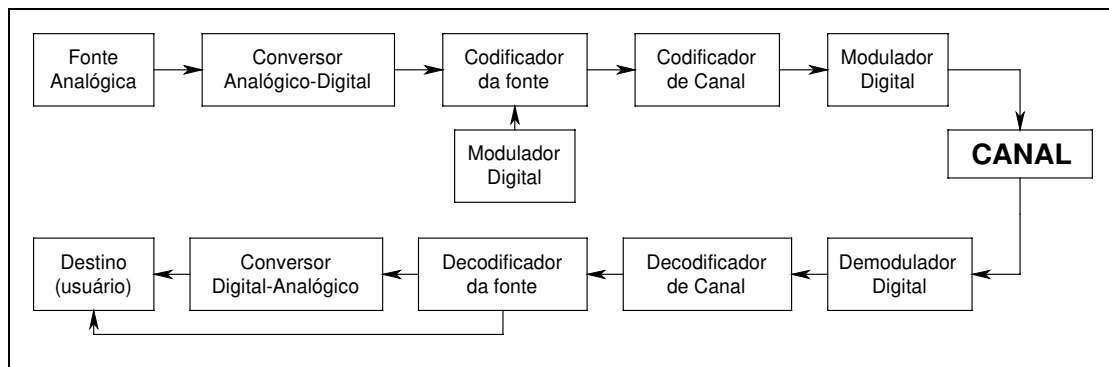


Figura 2.1- Diagrama de blocos de um sistema de comunicação genérico

2.2 Fundamentos de compressão de imagens

Os elementos chaves de um sistema de compressão são os codificadores (Figura 2.2), que realizam a compressão, transformando a informação visual original em uma sequência de dados codificados (“bitstream”) e os decodificadores (Figura 2.3), responsáveis pela reconstrução do material visual a partir da “bitstream”. Em geral, os

sistemas de compressão são assimétricos, de tal forma que a complexidade do codificador é muito maior do que a do decodificador, sendo a qualidade da imagem definida pelo codificador.

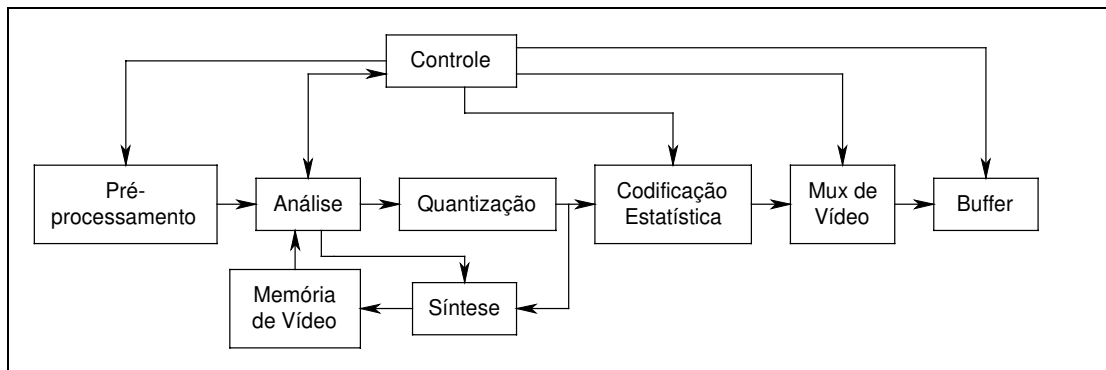


Figura 2.2- Estrutura de um codificador de vídeo

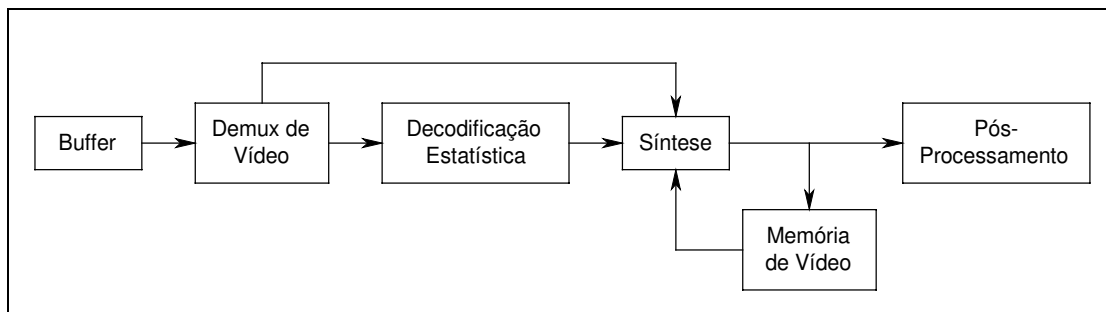


Figura 2.3- Estrutura de um decodificador de vídeo

Uma das primeiras ferramentas de transformação da fonte foi a análise preditiva [2,3]. O descritor utilizado neste caso é o erro de predição da imagem. Esta ferramenta tem a vantagem de ser bastante simples, porém os ganhos de compressão alcançados são baixos. Em codificadores práticos, utilizam-se preditores pré-definidos, pois os preditores otimizados dependem da estatística do sinal e a diferença de desempenho entre os dois não justifica o aumento de complexidade. Os preditores de vídeo se classificam, segundo a posição espaço-temporal das amostras, em preditores intra-campo, inter-campos e inter-

quadros, se as amostras estão no mesmo campo, ou em campos ou em quadros consecutivos, respectivamente.

Uma melhora significativa de desempenho é obtida pela incorporação de técnicas de estimação de movimento à análise preditiva. Os vetores de movimento, obtidos por técnicas de casamento de blocos, compensam eventuais movimentos de translação da imagem. Esta informação deve ser transmitida ao decodificador como informação lateral, que é conhecida previamente pelo equipamento.

Uma técnica mais eficiente é a descrição da imagem através dos coeficientes de uma transformada discreta aplicada sobre os pixels da imagem. A transformada escolhida deve ter características de descorrelação dos coeficientes e concentração da energia sobre alguns dos coeficientes. A transformada ótima, que garante total descorrelação dos coeficientes, é a transformada de *Karhunen-Loève*. Entretanto, esta transformada tem, em geral, grande complexidade, e por isso nos codificadores práticos faz-se opção por transformadas determinísticas, não dependentes do conteúdo da imagem, que praticamente atingem o desempenho da transformada ótima para a maioria das cenas. Algumas das transformadas determinísticas mais importantes são a transformada discreta de *Fourier* (DFT), a transformada discreta de cossenos (DCT) e a transformada de *Hadamard* (DHT).

A transformação determinística adotada em praticamente todos os *codecs* padronizados é a DCT. São tipicamente utilizados blocos de 8x8 pixels. A escolha do tamanho do bloco representa um compromisso entre complexidade e adaptabilidade. Cada coeficiente transformado carrega informações do conteúdo da imagem em alguma porção do espectro de frequências. Isto permite a priorização dos coeficientes no sentido de obter melhor casamento com a acuidade do sistema visual humano.

A aplicação da transformada proporciona “descorrelacionamento” espacial entre as amostras, porém persiste forte correlação entre campos e quadros consecutivos. Esta correlação temporal é explorada fazendo-se uma análise híbrida. A ferramenta de análise híbrida compreende a predição com estimação de movimento, seguida da aplicação da transformada sobre o erro de predição. A análise híbrida pode ser também bidirecional no tempo, e neste caso, a predição utilizada é progressiva e/ou regressiva, ou seja, na predição regressiva a ordem de codificação das seqüências precisa ser diferente da ordem de

exibição. Apesar das vantagens de desempenho em compressão e em facilidade de edição, a análise bidirecional aumenta a complexidade e os requisitos de armazenamento do sistema.

À etapa de análise, segue-se a etapa de quantização onde se dá efetivamente a compressão e onde são introduzidas distorções. Os principais fatores no controle da quantização são: a estatística dos descritores da fonte, as características do sistema visual humano e os dados do sistema (transporte). As informações de sistema descrevem as facilidades permitidas em aplicações escalonáveis, cifragem, acesso condicional e múltiplos programas.

2.3 Introdução ao MPEG2

O padrão de compressão de imagem especifica funções para dar suporte à rede de pacotes, para operar em canais com taxa de bits variável e garantir robustez frente a erros de bits e perda de células. O MPEG-2/vídeo tem funções particulares para vídeo entrelaçado e não entrelaçado.

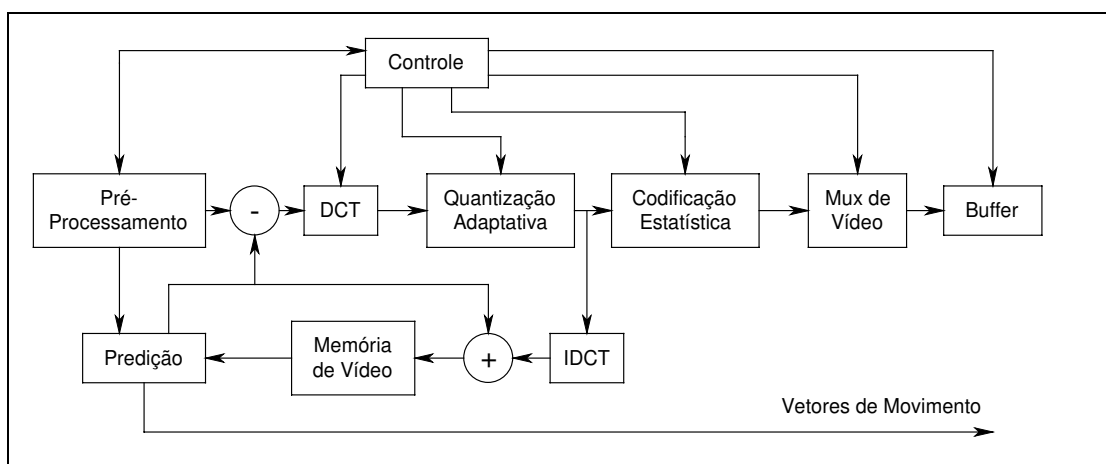


Figura 2.4- Estrutura de um *codec* MPEG-2: codificador

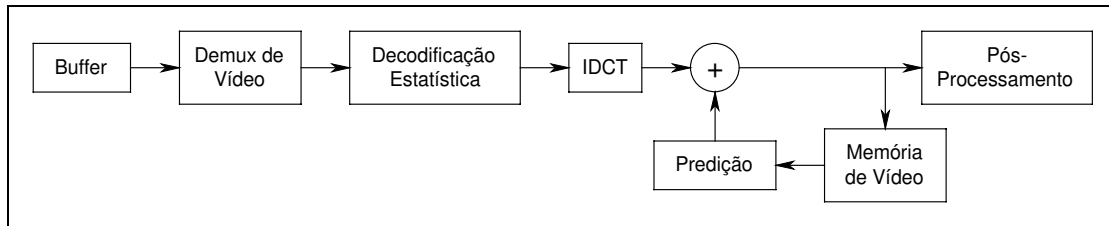


Figura 2.5- Estrutura de um *codec* MPEG-2: decodificador

O sinal de entrada deve ser vídeo digital em componentes, com um componente de luminância Y e dois componentes de croma Cb e Cr. Os formatos de codificação utilizados são YCbCr 4:2:0, YCbCr 4:2:2 e YCbCr 4:4:4 [4]. A menor estrutura definida é o bloco, um conjunto de 8 linhas de um mesmo campo, ou quadro por 8 amostras de uma mesmo componente. A unidade básica de codificação, entretanto, é o macrobloco, que consiste num bloco formado por quatro blocos de luminância, 4 blocos de Cb e 4 de Cr, no caso de codificação YCbCr 4:4:4, 2 blocos de Cb e 2 de Cr, no caso de codificação YCbCr 4:2:2 e 1 bloco de Cb e 1 de Cr no caso de YCbCr 4:2:0. Um conjunto de macroblocos forma uma fatia. Cada fatia tem, em seu início, uma palavra de sincronismo para os dados codificados. Portanto, na codificação, a imagem é tomada como um conjunto de fatias [5,6].

Dependendo da forma como são codificadas, as imagens podem ser classificadas em três tipos. Imagens tipo I, codificadas independentemente de outras imagens, imagens tipo P, codificadas com referência a uma imagem anterior no tempo, e imagens tipo B, codificadas com referência a uma imagem anterior e a uma imagem posterior no tempo. As letras I, P e B são tomadas para codificação Intraquadro, Predição e Interpolação Bilateral, respectivamente. As imagens tipo I constituem pontos de acesso do vídeo codificado. As imagens tipo B nunca são usadas como referência na codificação de outras imagens. Para uso como referência é dada preferência a imagens tipo I e P, por minimizar o efeito de propagação de erros. Como regra geral em caso de imagens campo, os dois campos da imagem são codificados segundo o mesmo tipo, com exceção feita a imagens tipo I, na qual um campo é tipo I e o outro é do tipo P. Um conjunto finito de imagens, que normalmente contém um ponto de acesso dado por uma imagem tipo I e que pode ser decodificada

independentemente de outras imagens, é chamado de grupo de imagens (GOP). As seqüências de vídeo, estruturas de maior hierarquia, são, dependendo do contexto, conjuntos de GOPs formados por campos ou quadros consecutivos.

Na etapa de pré-processamento, faz-se a necessária conversão de formato de codificação de croma, que corresponde a uma filtragem e/ou interpolação, seguida de subamostragem e reordenação dos quadros. Esta última etapa é necessária quando se utilizam imagens do tipo B. Na etapa de pós-processamento do decodificador, realizam-se os processos de reordenação e conversão inversos.

A codificação de imagens empregadas em MPEG-2/vídeo faz uso de técnicas de predição bidirecional, onde a imagem predita é a média entre a imagem anterior e a imagem posterior. A utilização de imagens com codificação bidirecional diminui o ruído proporcionando ganhos de desempenho significativos em baixas taxas, mas tem as desvantagens de aumento de complexidade, do atraso de codificação e também dos requisitos de “buffer”. O esquema básico de codificação é o algoritmo híbrido combinando DCT e predição com compensação de movimento. É, porém, um sistema amplamente adaptativo e que emprega codificação baseada em contexto: há uma variedade de modos de predição e de codificação que são transmitidos ao receptor como informação lateral.

Na etapa de análise preditiva, é feita a escolha do macrobloco de referência para o macrobloco atual, a ser codificado, com base no macrobloco que minimiza a diferença dos níveis de intensidade entre os blocos. Os candidatos a macrobloco de referência dependem do tipo de imagem a ser codificada. Em imagens tipo I não há predição. Nas imagens P, o único candidato na predição é o macrobloco de predição, o qual pode pertencer à imagem anterior, ou à posterior ou ser a média entre essas imagens.

O algoritmo de estimação de movimento define um vetor de deslocamento do macrobloco de referência mais casado com o macrobloco atual, segundo alguma função de erro. Em MPEG-2, utiliza-se DCT baseada em quadro, no caso de imagens com pouco movimento, e baseada em campos, no caso de imagens com muito movimento. Para cada macrobloco, é definido um índice de quantização (M_{quant}) que depende do controle de taxa e da importância relativa do macrobloco. Existem duas tabelas de codificação básica, uma para macroblocos intra (intraquadro), outra para macroblocos com codificação não intra

que são projetadas de acordo com a importância relativa do coeficiente transformado. Multiplica-se ao valor da tabela o parâmetro M_{quant} , que pode assumir valores de 2 a 62 em escala linear, em situações de taxas baixas ou moderadas, e valores de 1 a 112, não linearmente distribuídos no caso de altas taxas. A etapa seguinte consiste em codificar os coeficientes quantizados por corridas de zeros. Se um macro bloco tiver todos os seus coeficientes quantizados iguais a zero, então não é transmitido.

Muitas informações são codificadas por palavras de tamanho variável. Entre elas temos: símbolos produzidos pela codificação de corrida de zeros efetuada nos coeficientes quantizados, vetores de movimento, tipo de macrobloco e mudanças no valor do parâmetro M_{quant} . As tabelas contendo as palavras de comprimento variável são fixas e não podem ser alteradas pelo usuário.

A função do “buffer” é acomodar a taxa variável de codificação à taxa fixa do canal. O tamanho do “buffer” depende da configuração MPEG-2 adotada. No nível principal (“Main Level”) o tamanho máximo do “buffer” é de 1,8 *Mbits*. Para controle do “buffer” são estabelecidas médias de complexidade para cada tipo de imagem a partir de imagens anteriores. No início da codificação da próxima imagem, define-se uma meta em número de bits para codificação da mesma com base na informação de complexidade e no número de bits disponíveis até o final do GOP, objetivando manter a qualidade constante em todo o GOP. A estratégia para alocação dos bits consiste em verificar se há uma diferença significativa entre o número de bits já codificados e a quantidade de bits planejada. Se esta diferença é positiva, o valor do parâmetro M_{quant} é aumentado e, se é negativa, diminuído. Esta estratégia visa manter constante o número de bits por GOP, para que antes da codificação da primeira imagem tipo I do próximo GOP, o “buffer” de transmissão esteja praticamente vazio. Este equilíbrio é obtido se o número médio de bits codificados por intervalo de quadro é igual ao número de bits lidos do “buffer” no mesmo intervalo.

Diferenças na implementação das funções de codificação faz com que as diferenças de desempenho sejam significativas. De fato, o padrão não garante qualidade. As principais funções na determinação da qualidade são: a estimação de movimento, tanto o algoritmo como a área de busca; o esquema de controle de taxa; e as etapas de pré e pós-processamento.

2.4 Degradações

Em sistemas de compressão podem ser exploradas tanto as propriedades psicovisuais do sistema visual humano como a redundância estatística dos dados da imagem, para conseguir redução da taxa. Entretanto, alguns tipos de degradação podem ser introduzidos no processo de compressão [7]. Estas degradações são, em alguns casos, perceptíveis e incomodam o usuário, embora um dos objetivos dos sistemas de compressão seja minimizar seus efeitos.

Os sistemas de vídeo digital produzem degradações que são fundamentalmente diferentes das produzidas pelos sistemas analógicos. Estas degradações podem ser introduzidas através de filtragem, conversão, transformação, quantização e/ou transmissão. Perda de resolução, efeito de bloco, ruído de quantização e erros de bloco são alguns dos indesejados efeitos do processo de compressão. Essas degradações são influenciadas pelo algoritmo de compressão utilizado, conteúdo da imagem, origem do material-fonte e condições do canal de transmissão. Um aspecto que torna estas degradações tão diferentes das distorções analógicas é o fato de que estes são influenciados pelo conteúdo da imagem e, portanto, são variantes no tempo fazendo com que algumas cenas sejam mais difíceis de serem comprimidas do que outras.

Vale observar que, embora o vídeo comprimido seja geralmente degradado em relação ao original, alguns processos podem reduzir o nível de degradação existente no material. Isto significa, portanto, que para alguns tipos de degradação a codificação pode significar uma melhora na imagem. É o caso em que o sinal original já está degradado e pode acontecer, por exemplo, do material comprimido ser menos ruidoso do que o sinal original.

2.4.1 Blocagem

A degradação conhecida como efeito de bloco ou blocagem é uma deterioração na qual a imagem recebida apresenta padrões retangulares que não estavam presentes na imagem original. Isto pode ser observado na Figura 2.6b [8]. As discontinuidades se tornam visíveis quando a quantização é pouco refinada. Este tipo de degradação está presente em cenas complexas ou quando elevadas taxas de compressão são necessárias. Nestas situações, o “buffer” do codificador tende a encher e a quantização é ajustada para ser menos refinada e evitar o transbordo do “buffer”. Para uma dada granulidade de quantização, a blocagem é mais perceptível em áreas mais planas da figura.

2.4.2 “Blurring”

A perda de resolução é uma degradação caracterizada pela perda, ou quantização pouco refinada, dos componentes de alta frequência. Os componentes de alta frequência têm, usualmente, menos energia e são menos perceptíveis ao olho humano. Esta informação é descartada para que se atinja a taxa desejada. Na prática, em uma dada situação de taxa, há um compromisso entre alta resolução e baixo ruído. Este ajuste é subjetivo e depende da aplicação.

O efeito conhecido como “blurring” (Figura 2.6c) é a redução da definição em bordas e áreas com muitos detalhes espaciais. Em cenas de alta resolução espacial este efeito é consequência do compromisso na alocação de bits para a descrição dos detalhes de alta resolução e na descrição do movimento. No caso geral, é a etapa de pré-filtragem a maior responsável pela perda de resolução. Se a imagem for convertida de um formato com menor número de linhas para um formato com mais linhas, então, ela aparecerá borrada, antes mesmo da compressão.

Perda de resolução ou erros de crominância em vídeo digital comprimido podem ser mais perceptíveis que os demais tipos de degradação, pois a maioria dos sistemas emprega uma subamostragem na crominância. Na presença de objetos coloridos em movimento

rápido, a subamostragem da cromaância pode levar a um descasamento entre luminância e cromaância. Essa perda de resolução torna as outras deteriorações mais visíveis.

2.4.3 “Edge Business”

A degradação conhecida como “edge business” é causada, geralmente, pela quantização de um bloco contendo pixels que delimitam áreas planas com diferença significativa entre os níveis médios em cada uma das bordas. O resultado é o aparecimento de uma distorção concentrada nas bordas dos objetos, caracterizada pela variação temporal ou espacial da borda.

2.4.4 Ruído

Há uma grande variedade de degradações associadas ao ruído produzido pelos algoritmos de compressão. Os dois tipos mais importantes são o ruído de quantização (Figura 2.6d) e o ruído mosquito. O ruído de quantização é uma das deteriorações predominantes. Na compressão, a quantização é parte fundamental do processo e, em muitos sistemas, a quantização é o único processo que introduz distorções. Este ruído tende a ser descorrelacionado do sinal, mas não é uniformemente distribuído ao longo da imagem. No caso da utilização adaptativa por blocos, como é o caso de MPEG-2, ele pode estar associado ao conteúdo local da imagem.

O ruído mosquito é um tipo de “edge business” caracterizado por tipos de degradação em movimento ou padrões de ruído em blocos superpostos aos objetos. Na recomendação ITU-T P.930 é utilizado o termo *ruído correlacionado* para designar o efeito resultante da combinação de “edge business” e ruído mosquito, já que, em geral a caracterização das diferenças subjetivas entre os dois efeitos é confusa. A distorção do ruído correlacionado é, portanto, tipicamente visualizada ao redor de bordas em movimento.

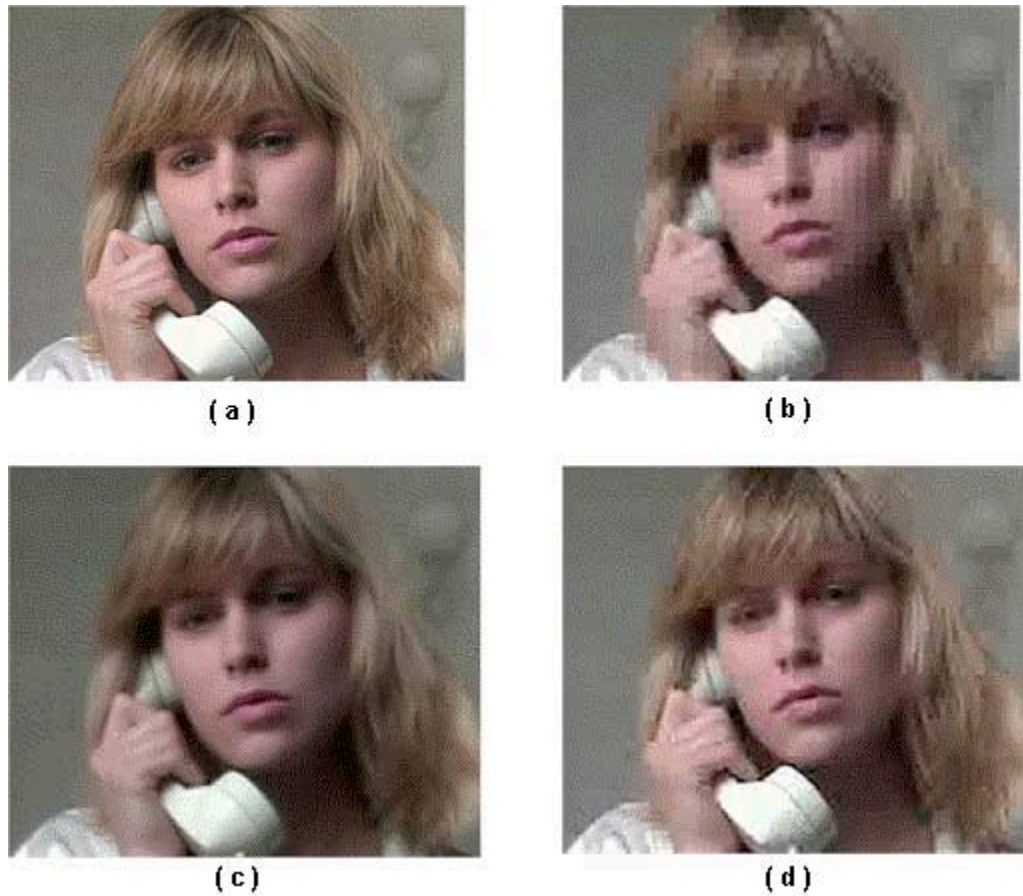


Figura 2.6- Imagens que exemplificam os tipos de degradações (**a**) imagem original (**b**) imagem com blocagem (**c**) imagem com “blurring” (**d**) imagem com ruído de quantização

2.4.5 “Jerkiness”

“Jerkiness” é uma degradação no qual o movimento originalmente suave e contínuo é percebido com uma sequência de cortes bruscos. No caso de sobrecarga no codificador, a imagem pode até mesmo ficar congelada. Tipicamente o “jerkiness” pode ser observado em *codecs* operando a taxas muito baixas. Este efeito é resultado da repetição de quadros para reduzir a quantidade de informação de vídeo. A baixa taxa de quadros resultantes é indicativa do nível desta degradação.

2.4.6 Blocos Errados

Devido a erros do canal, pode-se ter no vídeo reconstruído a presença de blocos errados. O vídeo digital comprimido é muito susceptível a erros de canal, devido à utilização de ferramentas de codificação preditiva. Dependendo do tamanho do GOP e do tipo de imagem afetada, um bloco errado pode se propagar por diversos quadros. Este espalhamento temporal é mais severo em imagens tipo I e em estratégias de codificação que utilizam GOPs grandes. Também dependendo da estratégia de predição intraquadros e da compensação de movimento, a transmissão de um bloco errado pode contaminar uma grande porção dentro de um mesmo quadro (espalhamento espacial).

2.5 Descritores

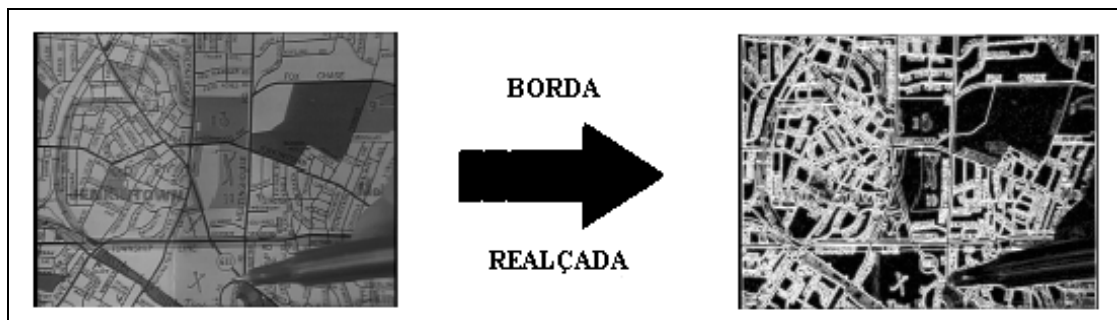
O propósito da avaliação objetiva de *codecs* é medir automaticamente a qualidade de vídeo em diversas condições, obtendo parâmetros de imagens de testes a partir do conhecimento de resultados padronizados. A escolha das cenas e dos parâmetros analisados representa uma característica muito importante da metodologia, pois o nível de detalhes e as formas de cada quadro de teste irão repercutir em parâmetros de maior ou menor qualidade.

Para obter os descritores de qualidade, são definidos parâmetros relacionados aos sinais de entrada e de saída, utilizando estimadores de baixa complexidade aplicados a regiões distintas da cena. Os valores obtidos para cada parâmetro podem variar fortemente em função da cena aplicada, produzindo diferentes níveis de degradação.

Existem duas categorias de análise: 1) as espaciais, que são medidas feitas em cada quadro do vídeo e 2) as temporais, que são medidas feitas entre quadros. Das características espaciais, destacam-se duas baseadas na magnitude da detecção de borda pelo gradiente de Sobel: energia total RMS e energia total RMS das bordas verticais e horizontais.

A Figura 2.7 mostra o processo usado para extrair informação espacial (SI) de um quadro de vídeo amostrado. Um gradiente ou algoritmo para realçar bordas é aplicado ao

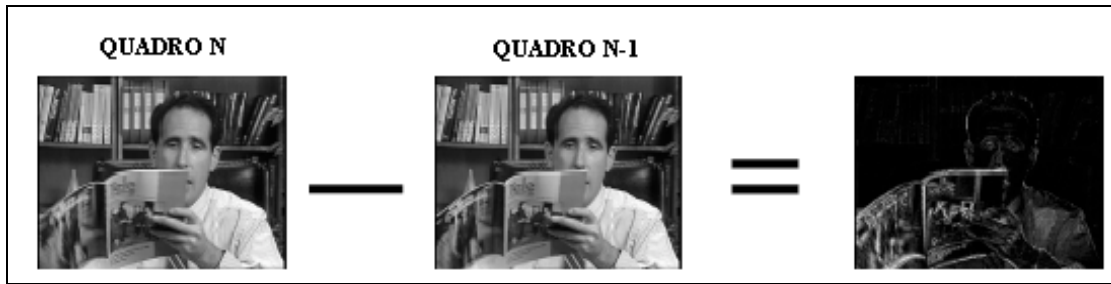
quadro de vídeo. Para cada pixel da imagem, dois operadores de gradiente são aplicados para aumentar as diferenças vertical e horizontal. Uma estatística é então calculada sobre a subregião selecionada do gradiente espacial da imagem para produzir uma quantidade escalar. Parâmetros para detecção e quantificação de sinais de vídeo degradados como “*blurring*”, “*blocking*” e “*edge busyness*” são medidos usando a análise espacial.



extraído de [9]

Figura 2.7- Exemplo de efeito de informação espacial

A Figura 2.8 mostra o processo usado para extração de informação temporal (TI) de um quadro de vídeo amostrado no tempo n . Primeiramente, os gradientes temporais são calculados, para cada pixel da imagem, pela subtração do *quadro n* pelo *quadro $n-1$* . Na Figura 2.8, aparece a magnitude absoluta do gradiente temporal e o gradiente temporal devido ao movimento. Um processo estatístico, calculado sobre a subregião do gradiente temporal da imagem, é usado para produzir o parâmetro escalar. Parâmetros como “*jerkiness*”, quantização de ruído, e erro de bloco são medidos utilizando técnicas de informação temporal.



extraído de [9]

Figura 2.8- Exemplo de efeito de informação temporal

Apesar da grande importância da informação temporal nos processos de compressão do sinal de vídeo, este trabalho não propõe técnicas de avaliação de qualidade que utilizam degradações temporais. Um dos motivos para isso foi o conhecimento, obtido através da Fundação CPqD, de que o problema de avaliação de qualidade pode ser resolvido, em boa parte, através da detecção de degradações espaciais.

A metodologia de avaliação objetiva pode utilizar estimação de degradação baseada em regiões, classificadas como regiões planas, de bordas e de textura [10]. Assim, uma distorção como bloqueio pode ser medida através de um detector de bordas aplicado às regiões homogêneas de uma cena de vídeo.

Os resultados de comparação devem ser correlacionados, através de um processo de treinamento, com a degradação associada à percepção visual, e esta é a maior dificuldade dos métodos de avaliação objetiva. A percepção visual é maior quando o ruído aparece na região homogênea em relação ao ruído na região de textura.

Capítulo 3

Fundamentação Teórica

A avaliação de qualidade de vídeo digital tem sido objeto de estudo de muitos grupos, oriundos de várias partes do mundo. A necessidade de ferramentas eficazes e confiáveis propiciou a criação de diversos métodos com variadas técnicas de processamento.

Um fator muito importante a ser tratado (Seção 3.1) é a capacidade de resposta do sistema visual humano diante de variações de intensidade de brilho e ruído em uma imagem. Estes parâmetros devem estar em uma faixa de níveis aceitáveis para que o observador tenha acesso a todas as informações e detalhes contidos nas cenas. São apresentados também às características dos fotoreceptores do sistema óptico humano e o fenômeno de percepção de cores.

Para a criação de métodos de avaliação eficientes é de extrema necessidade que se tenha uma base de dados subjetiva confiável, pois é praticamente impossível disponibilizar um grande grupo de voluntários adequadamente preparados para atuar junto a cada parâmetro de avaliação proposto. A avaliação subjetiva exige altos recursos e um rígido processo desde a seleção das seqüências de vídeo e iluminação da sala de testes até o melhor coeficiente estatístico das notas que representam a qualidade das cenas. Na Seção 3.2, são apresentados alguns dos procedimentos necessários para a avaliação subjetiva e são referenciadas as normas ITU que devem ser seguidas para extração destes dados.

Na seção 3.3 a definição de avaliação objetiva é apresentada, especificando as formas ou tipos de referência. São apresentadas as aplicações de cada um destes tipos de referência e comentadas as dificuldades de implementação com a diminuição dessas taxas de referência.

Um dos grandes avanços nas pesquisas referentes a esta área foi a criação de uma unidade padrão para medir os níveis de percepção humana diante de degradações (Seção 3.4). Esta escala, denominada escala JND (“Just Noticeable Difference”), possibilitou a comparação direta dos níveis de degradação com o que pode ser percebido pelo olho humano. Isto fez com que alguns dos grupos pertencentes ao VQEG, utilizassem a medida de JND nos algoritmos dos descritores de qualidade.

Como este trabalho está fortemente ligado ao VQEG, apresenta-se na Seção 3.5 um breve histórico de como começou e tudo que já foi feito nesta disputa pela padronização de avaliação de vídeo digital. São comentados os resultados encontrados na fase inicial do processo, e os possíveis avanços futuros.

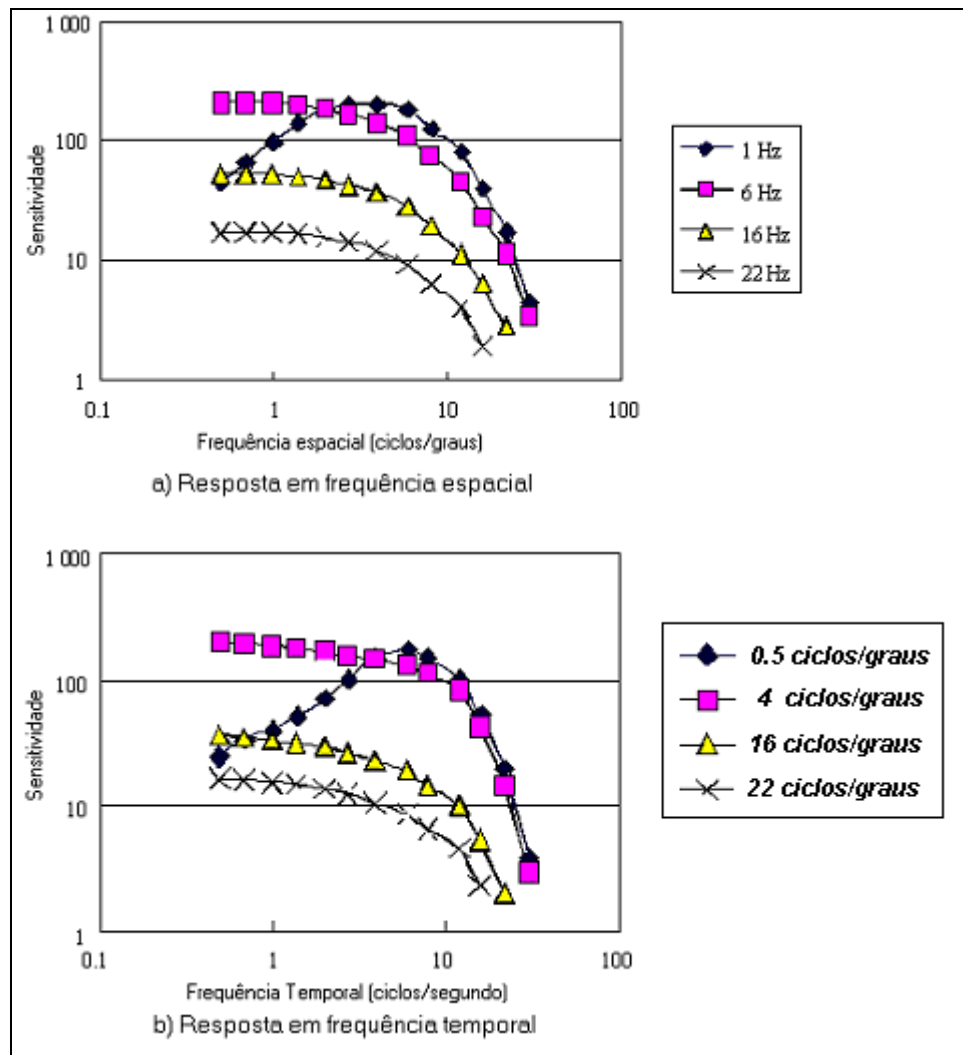
A Seção 3.6 segue com as propostas dos grupos que submeteram ao VQEG funções para avaliação objetiva com referência total. Estas propostas são descritas sem muita profundidade, já que cada grupo procurou publicar apenas as idéias centrais das implementações. Já a proposta da Fundação CPqD é discutida com mais detalhes, já que todo o material utilizado pelo grupo foi disponibilizado.

3.1 Sistema Visual Humano

Um ponto inicial muito importante para o estudo de sistemas de avaliação de qualidade de vídeo digital é entender, mesmo que superficialmente, as curvas de resposta do Sistema Visual Humano. O olho humano possui muitas limitações referentes às respostas ao brilho, à frequência espacial temporal e à percepção de ruído aleatório de um monitor. Assim, uma vez conhecidas estas limitações, pode-se determinar o quanto da imagem está sendo captada pelo observador e qual a influência do ruído aleatório no processo de inteligibilidade.

3.1.1 Resposta em frequência Espacial de Visibilidade

A resposta em frequência espaço-temporal de visibilidade humana foi medida por J.G. Robson [11] e outros. A resposta em frequência espacial mostra uma característica seccional de perpendicularidade para a resposta em frequência temporal, correspondente a uma simetria rotacional com os eixos ópticos no centro.



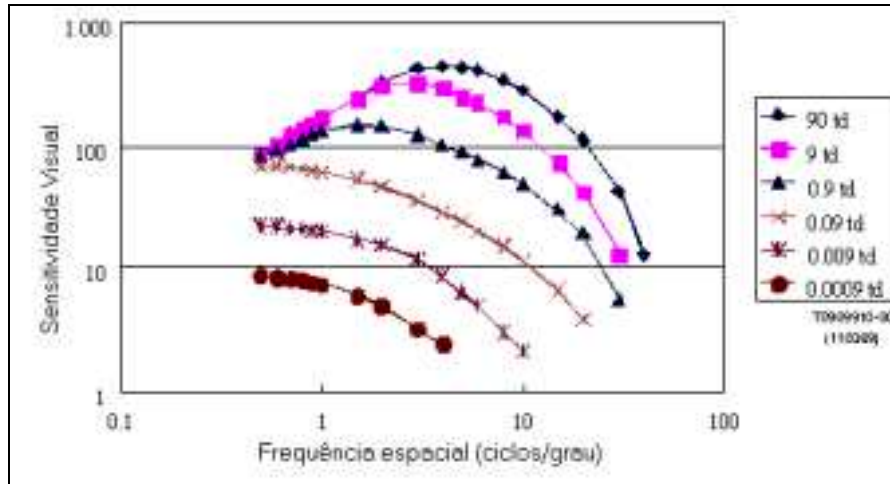
extraído de [12]

Figura 3.1- Resposta em frequência espaço-temporal de visibilidade

3.1.2 Resposta em frequência de visibilidade dependente do brilho da imagem

Tanto a resposta em frequência espacial como a resposta em frequência temporal dependem do brilho presente nas imagens de teste. A Figura 3.2 indica a dependência da resposta em frequência espacial de visibilidade do brilho de uma imagem com uma

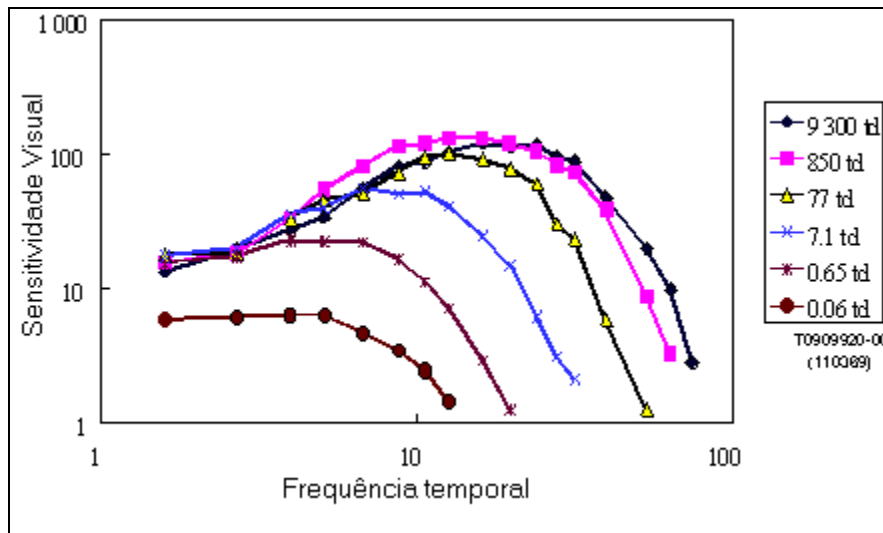
freqüência temporal menor que 4 Hz para a sensibilidade visual. “td” é uma unidade para a luminância de uma imagem “eyeground”.



extraído de [12]

Figura 3.2- Dependência da resposta em frequência espacial de visibilidade no brilho

A Figura 3.3 mostra a dependência da resposta em frequência temporal de visibilidade com o brilho no caso de uma imagem uniforme. Os olhos humanos são tipicamente sensíveis ao flicker de aproximadamente 10 Hz quando o brilho da imagem é alto. Quando o brilho da imagem é muito baixo, o flicker é invisível.

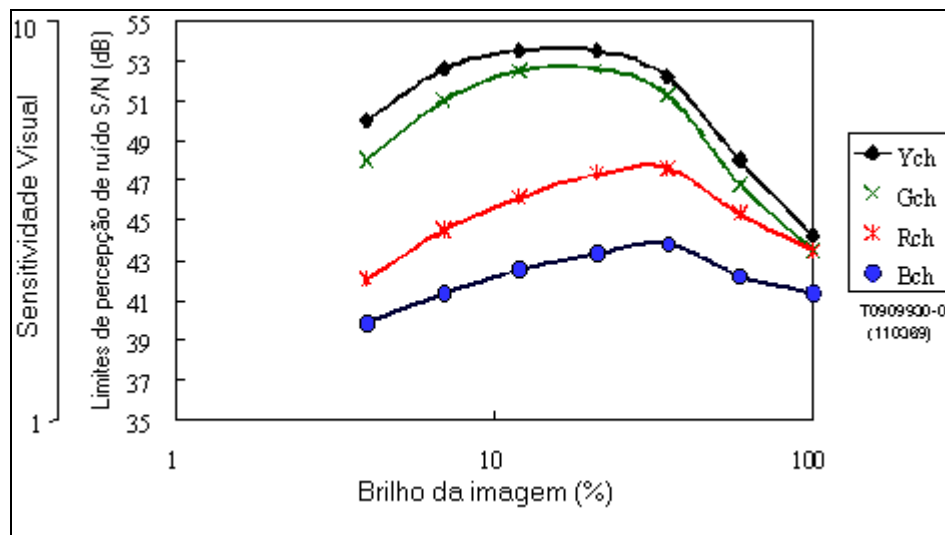


extraído de [12]

Figura 3.3- Dependência da resposta em frequência temporal de visibilidade no brilho

3.1.3 Dependência visual sensitiva ao brilho

A Figura 3.4 mostra os limites de percepção de ruído aleatório de um monitor de TV, sob diferentes níveis de brilho. Podem ser observados na figura, os comportamentos das curvas com os componentes R, G e B, e do componente de luminância Y. Como se pode perceber, a sensibilidade visual aumenta gradativamente à medida que o brilho aumenta. Quando é atingido um certo valor de brilho, a relação se inverte, passando a diminuir a sensibilidade visual a partir de um nível de brilho muito alto.



extraído de [12]

Figura 3.4- Limites de percepção de ruído aleatório em um monitor de TV

3.2 Fotorreceptores, Percepção de Cor e Conversão de Espaço de Cor

Através do sistema óptico do olho humano, toda a luz captada é projetada sob a retina, onde fisicamente está uma fina camada de tecidos neurais de fotorreceptores. Existem dois tipos de fotorreceptores: os bastonetes e os cones [13]. Os bastonetes iniciam

a visão com níveis de baixa iluminação, sendo assim insignificantes para aplicações de métricas de qualidade, no qual são usualmente empregadas condições normais de iluminação. Os cones são responsáveis pela percepção visual sob condições de níveis de luz mais altos, e podem ser caracterizados em três tipos: L-cones, M-cones e S-cones, já que eles são sensíveis a comprimentos de onda longos, médios e pequenos, respectivamente. Experimentos com cores de alta luminosidade tem revelado alta característica de absorção dos cones [14].

Por outro lado, a taxa absoluta de absorção dos cones não pode explicar a absorção de cor. Isto tem sido observado a partir de alguns pares de cores que podem coexistir em uma sensação simples de cor enquanto outros não. Este fato conduz a teoria de “cores oponentes”, e a existência desses caminhos visuais separados são sustentados por experimentos fisiológicos e psicológicos. Os três componentes de cor principais são preto e branco (B-W), vermelho e verde (R-G) e azul e amarelo (B-Y). Um trabalho recente de Poirson e Wandell propõe um padrão de cor com espaço de cor separável [15], que é de grande interesse para o desenvolvimento de métricas de qualidade de vídeo e imagem digital.

Em muitos dos sistemas de codificação de vídeo digital, os sinais estão no espaço de cor YCbCr, e necessitam ser convertidos em um espaço de cores oponentes, considerando que os estudos de contraste sensitivo são baseados em estímulos de cores oponentes. Na técnica de avaliação de qualidade criada em [16], adota-se a teoria de cores oponentes e converte-se as cores YCbCr para RGB [4], em seguida para o espaço de cor LMS e finalmente para o espaço de cores separáveis mencionado anteriormente.

No modelo proposto em [17], por exemplo, um espaço de cor oponente similar é usado. Os sinais de entrada são transformados de RGB para coordenadas CIE XYZ [18]. As coordenadas CIE são convertidas em respostas LMS dos cones. As coordenadas de cores oponentes, Luminância (preto e branco), azul e amarelo, e vermelho e verde são derivados das coordenadas L, M, S e Y.

Apesar de não existir ambigüidades entre os espaços de cor LMS e os caminhos de cores oponentes, a forma exata do espaço de cor está ainda sujeita a debates [19].

3.3 Avaliação Subjetiva

Na implementação de sistemas de compressão digital para contribuição, distribuição e armazenamento, atenção deve ser dada à avaliação de seus desempenhos. O estudo de técnicas de processamento de sinais deve levar em consideração o usuário final do serviço, a quem compete julgar a qualidade final do sinal. Para que se tenha melhor garantia de sucesso em termos subjetivos, as características dos sistemas de compressão e os tipos de degradação associados devem ser cuidadosamente considerados e avaliados. A caracterização de parâmetros que afetam a qualidade perceptível do sinal pressupõe resultados subjetivos confiáveis e reproduzíveis.

Os procedimentos de teste diferem bastante se o objetivo é testar a qualidade ou a inteligibilidade do sinal digitalizado [20]. A classificação de um *codec* utilizado em aplicações de distribuição se refere à classificação feita sobre a imagem decodificada após uma passagem pelo *codec*. Para qualidade de contribuição, entretanto, deve-se analisar a qualidade básica após vários *codecs* colocados em série (configuração em cascata) de forma a simular uma aplicação de contribuição típica. Em geral, os problemas de especificação de qualidade de imagem têm sido considerados separadamente para sistemas analógicos e digitais. Se de fato existir uma independência psicológica entre as degradações, então elas poderão ser julgadas como degradações causadas pela seção digital, pela seção analógica, ou ainda degradações causadas simultaneamente pelas duas seções. Testes subjetivos são parte importante também na etapa de projeto de codificadores. Mesmo em situações em que medidas objetivas que apresentam boa relação com a qualidade percebida da imagem estão disponíveis, os testes subjetivos são usados para complementar, verificar ou calibrar tais medidas objetivas.

O objetivo destes testes é estabelecer, através de métodos empíricos, bases para os julgamentos. É essencial que o método de medida usado leve a resultados válidos e representativos das opiniões. A Recomendação ITU-R BT. 500-7 [21] é largamente usada

para o projeto de avaliação de sistemas de compressão de vídeo. Embora a mesma metodologia possa ser usada para os diferentes tipos de *codecs*, que usam codificação intra-campos ou inter-quadros, usando palavras de comprimento fixo ou variável, as imagens de teste escolhidas devem ser diferentes e isso terá grande influência sobre o resultado final do teste. Os métodos existentes não estabelecem níveis absolutos de qualidade. Ao contrário, os resultados obtidos são sempre influenciados pela escolha das imagens de referência.

Além da Recomendação ITU-R BT.500 que se aplica à televisão digital de resolução convencional, existe um conjunto de normas do ITU-R que provêem instruções e métodos disponíveis para o acesso à qualidade da imagem em um ambiente de laboratório controlado. A Recomendação ITU-R BT.710 [22] trata especificamente de televisão de alta definição e a Recomendação ITU-T BT.910 [23] faz referência à avaliação subjetiva em aplicações multimídia.

Por não ser o enfoque principal deste trabalho, tanto as seqüências de vídeo originais e degradadas como a base de dados subjetiva utilizada foram disponibilizadas pelo grupo de Processamento de Sinais Audiovisuais da Fundação CPqD.

3.4 Avaliação Objetiva

As atividades em avaliação objetiva de qualidade de vídeo comprimido se restringiram, inicialmente, à aplicação de sinais de teste artificiais, para medidas de ganho, deslocamento, resposta em frequência 2D e área ativa. A natureza custosa e demorada dos testes subjetivos, aliada ao fato de estarem baseados somente nos tipos de degradações visíveis, motivou a elaboração de métodos de desempenho de *codecs* baseados em medidas objetivas. Assim, embora se acredite que a melhor avaliação da qualidade de vídeo é feita por observadores humanos, é necessário o desenvolvimento de medidas objetivas de qualidade de vídeo comprimido. Medidas que são necessárias para organizações de padronização, usuários finais e provedores de serviços. Medidas objetivas imparciais, confiáveis, repetíveis e de baixo custo, tanto de vídeo como de desempenho dos sistemas de transmissão de imagem, aumentam a competição entre os provedores de serviços e

equipamentos e, também, permite melhor especificação e avaliação de novos sistemas. A possibilidade de realização de medidas objetivas em tempo real abre a perspectiva de monitoração contínua da qualidade de vídeo. Além disso, o crescente número de novos algoritmos de codificação de imagens faz com que a realização de testes subjetivos não seja factível.

Necessita-se, assim, de medidas com boa correlação com os resultados subjetivos, de forma que se possa prever a aceitação dos usuários frente aos artefatos de compressão. Deseja-se que estas medidas forneçam valores absolutos dos erros de codificação e que se desenvolvam regras para sua concatenação, possibilitando assegurar o nível de qualidade desejado ao longo da cadeia de produção.

As formas de análise objetiva da qualidade de *codecs* de vídeo estão divididas em dois tipos: 1) Avaliação com "Referência Total", onde o descritor objetivo de qualidade é obtido a partir de uma grande quantidade de informação, e 2) Avaliação com "Referência Reduzida", onde está disponível uma quantidade limitada de informação.

3.4.1 Avaliação com Referência Total

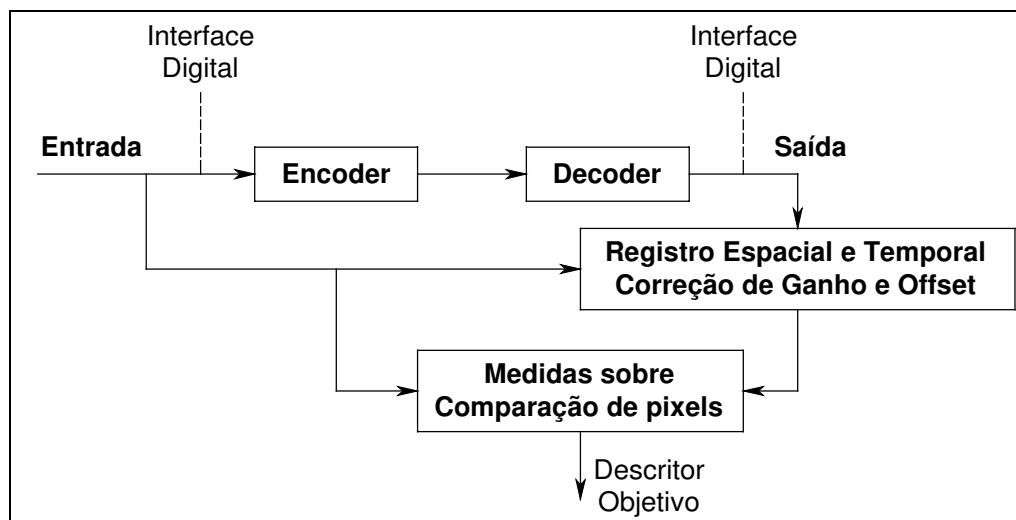


Figura 3.5- Processo de avaliação usando "Referência Total"

A Figura 3.5 mostra a configuração do processamento computacional usado para obtenção dos parâmetros objetivos e da qualidade de vídeo para o caso com “Referência Total” ou “Em Laboratório” [12]. Este tipo de avaliação permite o uso de uma quantidade ilimitada de referência, já que é possível acessar a imagem original durante todo o processo. As medidas são realizadas sobre comparações pixel-a-pixel entre imagens original e degradada. Por isso, o objetivo deste parâmetro é reunir a maior quantidade de informação possível e útil, e que possam repercutir em um parâmetro de qualidade robusto e eficaz.

Como pode ser observada na Figura 3.5, a imagem de entrada é utilizada em três partes do processo. Num primeiro estágio, a imagem original sofre um processo de degradação, sendo levada ao codificador e depois decodificada, criando-se assim uma versão degradada. Num segundo momento, a imagem original é processada por métodos ou algoritmos de alta ou baixa complexidade, podendo envolver, por exemplo, processos de segmentação, que é a técnica utilizada pelo CPqD. Finalmente, a imagem de entrada pode ser usada sem modificações para gerar parâmetros comparativos com a imagem degradada sem processamento.

Uma etapa fundamental no processo de avaliação é aquela que envolve o registro temporal e espacial e a correção de ganho e offset da imagem degradada. O registro temporal é necessário para a identificação de qual quadro original representa o quadro degradado gerado. Isto evita confusões devido a tempos de espera criados durante a codificação e decodificação. O registro espacial faz com que possíveis deslocamentos da área de vídeo ativo sejam corrigidos, e que os pixels referentes à imagem degradada sejam combinados com os respectivos pixels da imagem original. A Figura 3.6 ilustra este deslocamento, sendo a região em destaque, com dimensões N_v e N_h , a área de vídeo ativo do quadro processado.

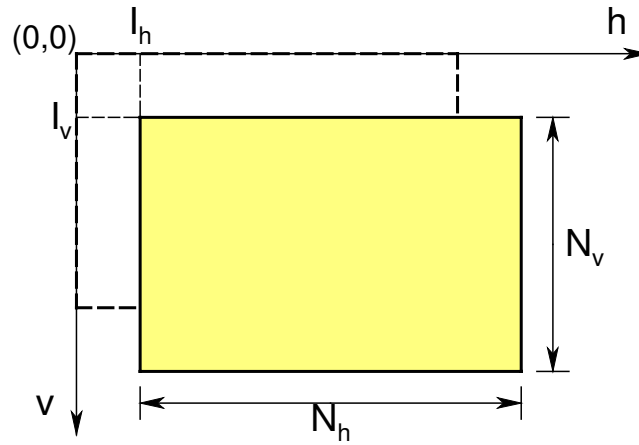


Figura 3.6 - Ilustração de deslocamento de área de vídeo ativo

A área de vídeo ativo representa a porção retangular do sinal de vídeo de entrada que não é eliminada pelo sistema sob avaliação, ou seja, que é efetivamente codificada. O offset representa um acréscimo constante nos níveis de intensidade dos pixels nos quadros de vídeo processados. O ganho é um crescimento proporcional de todos os valores de intensidade de luminância ou cromaância na imagem, como pode ser observado nas equações a seguir.

$$\begin{aligned} Y_{out} &= g_Y \cdot Y_{in} + I_Y \\ Cb_{out} &= g_{Cb} \cdot Cb_{in} + I_{Cb} \\ Cr_{out} &= g_{Cr} \cdot Cr_{in} + I_{Cr} \end{aligned} \quad (3.1)$$

sendo: g_Y , g_{Cb} e g_{Cr} os ganhos respectivos aos campos Y, Cb e Cr

I_Y , I_{Cb} e I_{Cr} os offsets respectivos aos campos Y, Cb e Cr

Os ganhos e offsets podem ter valores diferentes para os componentes de Luminância e Crominância. Em [24] estão detalhadas as técnicas utilizadas pelo CPqD para correção de ganho e offset.

A etapa de processamento de sistemas de avaliação objetiva será tratada mais adiante com a apresentação dos vários métodos propostos pelos grupos participantes do VQEG.

3.4.2 Avaliação com “Referência Reduzida”

Na Figura 3.7 é apresentado o modelo de referência usado em medidas de desempenho com “Referência Reduzida” ou “Em Serviço” [9,25,26], que são sensíveis às distorções introduzidas pelo codificador ou decodificador de vídeo. Neste esquema, têm-se dois equipamentos, um no transmissor e outro no receptor, que extraem dos sinais de vídeo características que, transmitidas por um canal auxiliar de baixas taxas, podem ser usadas para monitorar a qualidade do serviço disponível. Neste canal auxiliar, é enviada informação de sincronismo e as medidas objetivas da seqüência de vídeo original até a saída do decodificador. Em seguida, cada quadro de vídeo degradado é sincronizado, sendo realizadas as mesmas medidas objetivas da seqüência original. Os dois conjuntos de medidas são então comparados para gerar o descritor de qualidade.

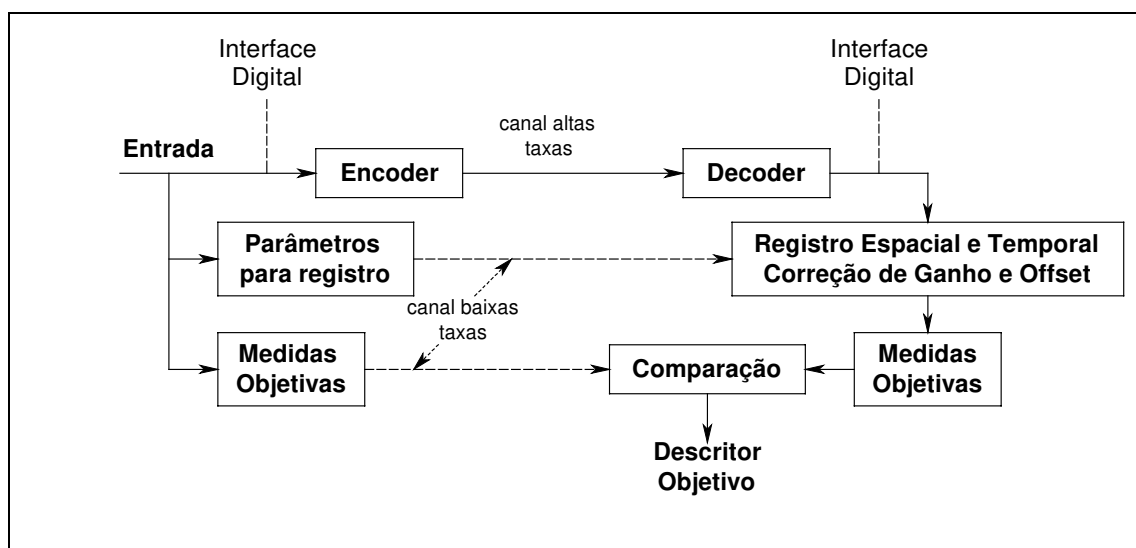


Figura 3.7- Processo de avaliação usando referência reduzida

O grande desafio deste sistema é o de identificar características úteis para a avaliação de qualidade de vídeo, que podem ser extraídas em tempo real e reduzidas a poucos valores, permitindo a transmissão no canal auxiliar de baixas taxas. O mais importante é que a comparação destas características medidas no transmissor e no receptor seja fortemente correlacionada com a degradação visual introduzida pelo sistema de transmissão.

O VQEG estabeleceu os limites com as taxas de referência para os sistemas com referência reduzida em 10, 56 e 256 kb/s, sendo que os dados de avaliação podem ser transmitidos juntamente com cada quadro de vídeo codificado. Isto facilita a sincronização entre as imagens e seus dados de referência.

Desta forma, pode-se calcular as taxas mínimas e máximas de referência para cada quadro de seqüências de imagens típicas. Considerando a existência de 30 quadros por segundo, e ainda que estes dados possam ser comprimidos, e que um compactador comum de dados reduza de 8 a 10 vezes o tamanho do arquivo, obtém-se as taxas mostradas a seguir.

$$\text{Taxa mínima de referência} = \frac{10 \text{ Kb/s}}{30} \cdot 10 = 3,33 \text{ Kb/quadro}$$

$$\text{Taxa máxima de referência} = \frac{256 \text{ Kb/s}}{30} \cdot 10 = 85,3 \text{ Kb/quadro}$$

Além dos sistemas com referência total e referência reduzida, existe um outro tipo de avaliação para sistemas sem referência (NR), em que somente as características do sinal degradado são levadas em consideração. Atualmente, existem poucos trabalhos publicados envolvendo este tipo de avaliação. Em [27], utiliza-se uma análise da quantidade de blocagem presente na imagem degradada através dos componentes harmônicos da transformada de Fourier. A quantidade de blocagem é proporcional a intensidade dos harmônicos encontrados na imagem codificada. Já em [28], é proposta a medição de três parâmetros de degradação diferentes: a blocagem, a detecção de bordas suavizadas e a detecção de bordas falsas. As bordas falsas são trechos ou pixels das bordas da imagem

original que não estão presentes na imagem codificada, apresentando níveis muito mais claros ou muito mais escuros quando comparados aos pixels vizinhos. Neste caso, é usada uma base de dados, criada a partir de seqüências degradadas, com cada um dos três tipos de degradação considerados. Cada um dos parâmetros encontrados recebe uma ponderação para a extração de um único descritor de qualidade.

3.5 Parâmetros Objetivos vs. Parâmetros Subjetivos

Os trabalhos de predição de qualidade subjetiva com base em medidas objetivas têm dois objetivos teóricos básicos: encontrar o melhor conjunto de medidas objetivas para prever o resultado subjetivo e determinar quão perto do ótimo estão os descritores.

Como desenvolver os parâmetros de desempenho de vídeo e como eles devem ser relacionados à percepção humana pode dar margem a considerações filosóficas. Considere um tipo de degradação que não seja detectado por nenhum observador. Neste caso, as medidas objetivas devem ou não ter sensibilidade suficiente para detectar este tipo de deficiência do sinal? Se as medidas objetivas forem tomadas como substitutivas das medidas subjetivas, então os parâmetros objetivos devem se comportar de forma similar e serem insensíveis a este tipo de degradação. Entretanto, o conhecimento de que a distorção existe é desejado, tanto para o conhecimento do usuário quanto para o isolamento e manutenção do sistema defeituoso.

Estas considerações mostram a necessidade do desenvolvimento de uma hierarquia de medidas objetivas. Esta hierarquia deve levar em conta o usuário final bem como o provedor do serviço e garantir flexibilidade na especificação dos parâmetros de desempenho de um grande número de sistemas e aplicações. A Figura 3.8 ilustra este raciocínio e evidencia a necessidade de desenvolvimento de parâmetros para quantificar cada uma destas dimensões. Supõe-se, na figura a seguir, que as dimensões de qualidade de vídeo podem ser independentemente quantificadas por parâmetros objetivos. Sempre existe, no entanto, certo nível de dependência entre os parâmetros objetivos.

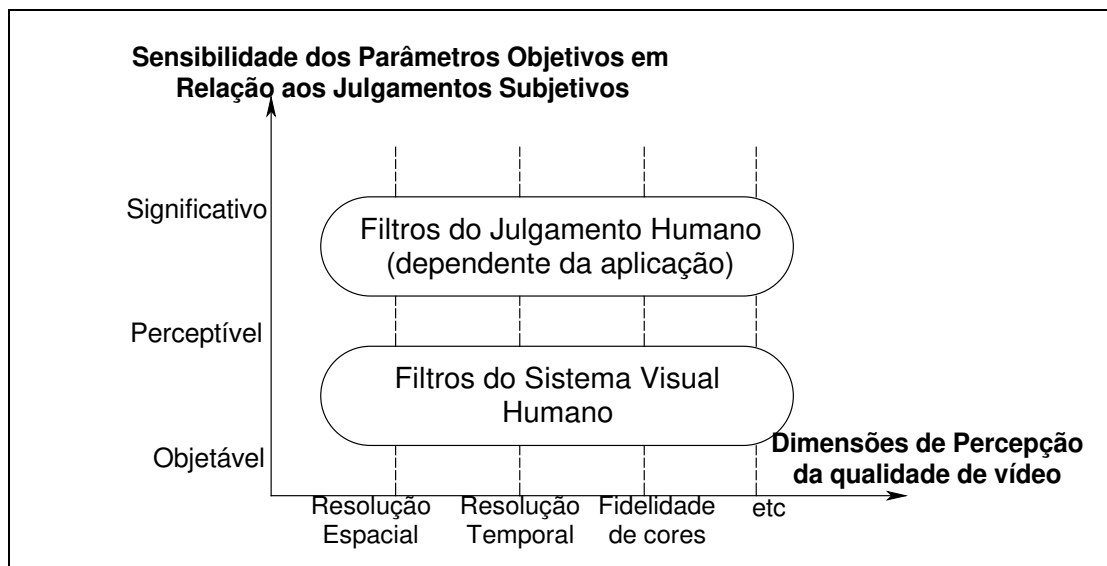


Figura 3.8- Hierarquia de desenvolvimento de parâmetros objetivos

Esta abordagem, contrariamente às tentativas de desenvolvimento de um único parâmetro, garante máxima flexibilidade na especificação dos atributos do sistema e no casamento destes atributos à qualidade. Imagens médicas, por exemplo, requerem alta resolução espacial, enquanto que imagens para aplicações em teleconferência necessitam apenas de uma resolução espacial moderada para serem julgadas “excelentes”. Pode ser observado (Figura 3.8) no eixo vertical os graus de sensibilidade dos parâmetros objetivos em relação aos julgamentos subjetivos. No nível mais baixo, as medidas objetivas detectam distorções que não são visualmente perceptíveis. O terceiro nível leva em consideração as medidas que afetam o julgamento humano; este é o nível mais difícil de ser quantificado. Os filtros de julgamento humano devem transformar as distorções percebidas em medidas de distorção significativas, considerando a aplicação e a população envolvidas.

3.6 Escala JND

No campo da psicofísica, que estuda a sensação e percepção humana, sempre se buscou medidas que pudessem quantificar a reação humana diante de variações de

qualidade de imagem. A idéia era estabelecer uma unidade que representasse exatamente quanto uma imagem pode ser melhor do que outra. Tendo este problema em foco, o IEEE G-2.1.6 *Subcomitê sobre Medida de Compressão de Vídeo*, juntamente com grupos de pesquisa ligados ao VQEG, chegou a uma eficiente técnica para medir unidades de sensibilidade da visão humana diante de degradações. Constatou-se, então, que esta medida poderia ter um papel importante durante processos de avaliação de qualidade de vídeo. Esta técnica [29] é resumidamente tratada a seguir.

A cada instante de uma seqüência de vídeo é admitida no observador uma magnitude de degradação percebida. As magnitudes percebidas são tomadas com ter variância unitária. Uma escala de intensidade física, diante de uma quantidade de deterioração adicionada, conduz a uma escala de deterioração percebida, nas chamadas unidades de *JND* (“just noticeable difference”), ou *função escala*.

Em experimentos que variam a degradação de uma seqüência de vídeo original, a medida física de degradação possui ponderação linear. O topo da escala (peso=1) tem um valor de JND que é chamado de JND total, sendo a medida de JND para a condição de compressão testada. É importante, portanto, obter as escalas JND e o número total de JNDs das condições utilizadas.

Inicialmente foi proposto um método baseado em limiares concatenados, que gerava a escala de qualidade a partir de um conjunto de medidas sucessivas de limiares discriminados. Porém, foi constatado que tal método não gerava uma escala confiável. A partir disso, foi criado o método de “Estimação eficiente de Escala Adaptativa” (EASE).

A Figura 3.9 mostra o procedimento do método EASE para estimação da escala JND. Uma formulação matemática é adotada para a função escala e os parâmetros iniciais são arbitrários. Baseado na função, um conjunto de pares de ponderações são selecionados para especificar aproximadamente um número de JNDs. Uma tentativa é então conduzida para cada um destes pares. Todos os dados são então usados para estimar os parâmetros da função. Se os dados coletados forem suficientes, o procedimento pára, e o dado deve ser usado para estimar os JNDs para a condição. Se mais tentativas são necessárias, um novo conjunto de pares é derivado da função novamente estimada, e o procedimento continua.

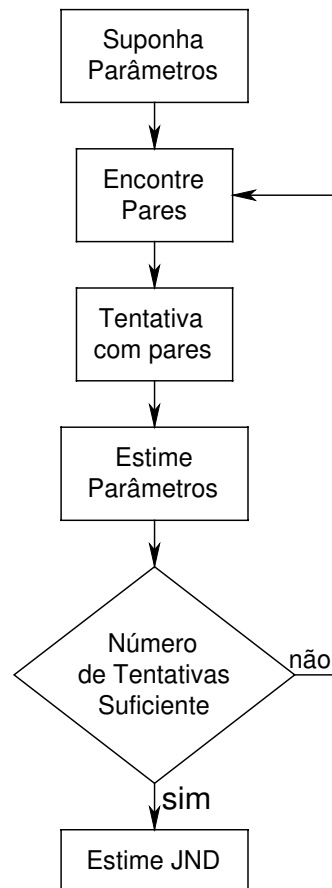
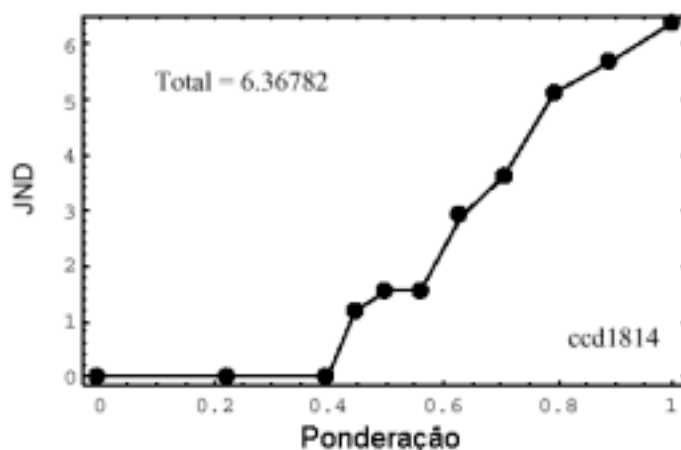


Figura 3.9- Fluxograma para o método EASE

Várias funções matemáticas podem ser consideradas para a função escala durante o método EASE. Tipicamente, a função terá um parâmetro que é o valor da escala quando a ponderação é 1, que representa o JND total ou parâmetro genérico *max*. Como regra de terminação, pode ser testado se $tentativas/Max \geq tentativas/jnd$, onde 32 representa um valor confiável de *tentativas/jnd*.

Uma vez que o EASE está completo, o dado pode ser analisado para derivar uma função escala e estimar o JND total para a condição. Um método de análise muito eficaz é a *estimação por partes da função escala*. Neste método, ajusta-se uma função, definida como

função por partes, pelos valores de cada uma das ponderações usadas na sessão. Os parâmetros são simplesmente os valores da função naqueles pontos.



extraído de [29]

Figura 3.10- Função por partes estimada da função escala

Os parâmetros são estimados pela minimização do logaritmo negativo da função de verossimilhança. O JND total é o último ponto na função, com peso 1. A Figura 3.10 mostra um exemplo dos resultados deste procedimento. Cada ponto na Figura corresponde a uma ponderação usada na sessão. A coordenada vertical é o número estimado de JNDs para cada ponto.

Em [28], utiliza-se uma técnica similar para obter as bases de dados JND, mas com algumas variações importantes. No caso anterior, as degradações inseridas nos quadros sob análise são geradas por vários codificadores, colocados em cascata ou não. Isto gera imagens codificadas com diversos tipos de degradações ao mesmo tempo. Neste caso, procura-se analisar 3 tipos de degradações, consideradas mais importantes, separadamente. Cada imagem original é degradada apenas com bloqueio (blocking), apenas com “ringing” e apenas com inserção de bordas falsas (corner outlier). A partir da Figura 3.11, podem ser vistos os três eixos de degradação, “blocking” B, “ringing” R e “corner outlier” C, com os respectivos valores de JND.

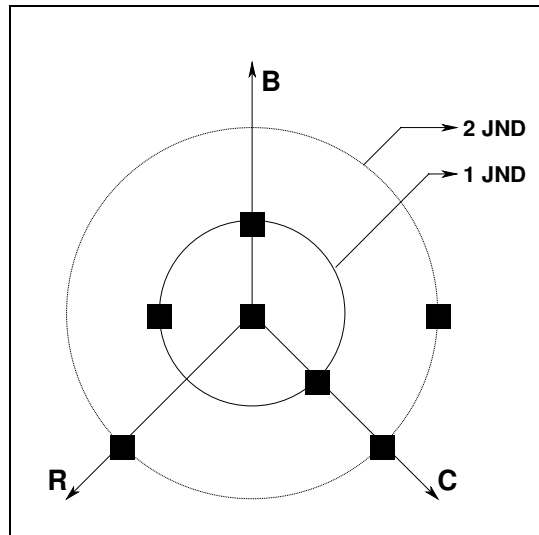


Figura 3.11- Espaço de medida de qualidade e pontos de teste

A degradação sob teste é incrementada até o ponto onde um observador possa notar a diferença. Assim, cada uma das imagens degradadas é processada pelo método EASE descrito anteriormente, até que se alcance uma base JND para cada um dos tipos de deterioração.

Um grande problema é que a sensibilidade humana às degradações varia muito de um parâmetro para outro, sendo necessária a criação de uma escala relativa a cada parâmetro. Assim, pouca quantidade de bloqueio, por exemplo, inserida na imagem original, pode criar um mesmo efeito de degradação visual que uma quantidade muito maior de “ringing” na mesma imagem.

3.7 Histórico do VQEG

O Grupo de Especialistas de Qualidade de Vídeo (VQEG) reúne pesquisadores e especialistas de várias afiliações, incluindo participantes de grandes organizações de reconhecimento internacional. Foi criado em 1997 em um encontro de especialistas de

qualidade de vídeo em busca de um objetivo comum: a padronização de sistemas de avaliação de qualidade de vídeo digital.

Existe, atualmente, um interesse crescente dos pesquisadores e profissionais da área de vídeo digital pela garantia e manutenção de um nível aceitável da qualidade do serviço de distribuição. O VQEG foi criado para tratar especificamente de edições de qualidade de vídeo, propondo métodos para prever a qualidade da imagem. Os sistemas de medida tradicionais, quando ainda necessários, não são adequados para medir a qualidade de sistemas digitais que sofreram processos de compressão. Com a passagem da tecnologia analógica para a digital, ocorreu uma grande mudança nos tipos de degradações visuais. A fim de avaliar corretamente estes novos tipos de deterioração, novos métodos objetivos necessitam ser desenvolvidos e padronizados.

Historicamente, a forma mais usada para a avaliação das imagens de vídeo foi a avaliação subjetiva. Por mais de 20 anos, os pesquisadores de todo o mundo usaram métodos esboçados na recomendação 500 da ITU-R [30] para avaliar a qualidade de vídeo em serviços de televisão. Mais recentemente, o ITU-T desenvolveu a recomendação P.910 [31] para padronizar métodos de avaliação de qualidade multimídia. A grande inviabilidade da avaliação subjetiva é o uso de observadores humanos para julgar as seqüências de vídeo (geralmente curtas). Desta forma, apresenta-se como um método pouco prático e impossível para avaliação contínua em tempo real. Os métodos objetivos que emulam o sistema visual humano são extremamente necessários.

A fase inicial do VQEG baseou-se em propostas para avaliação objetiva em sistemas com referência total (FR). Em março de 1999, 10 grupos de pesquisa submeteram seus modelos de avaliação para julgamento. Foram utilizadas bases de dados subjetivas de 8 laboratórios independentes, organizadas em seqüências de baixa e alta qualidade. As taxas de bits das seqüências de baixa qualidade variaram entre 768 kbits/s e 4.5 Mbits/s, e as seqüências de alta qualidade entre 3 Mbits/s e 50 Mbits/s. Nesta primeira fase, nenhum grupo obteve grande destaque diante dos demais.

O desafio de superar o modelo PSNR [12] (pico da relação sinal-ruído) não foi alcançado em sua totalidade, existindo condições de degradação em que este modelo é mais eficiente. O PSNR representa um parâmetro simples e é o único, até o momento, que a ITU

recomenda. O modelo submetido pelo CPqD alcançou excelentes resultados, principalmente para as seqüências de baixas taxas.

Diante das correlações obtidas pelas técnicas propostas, relativas às bases subjetivas e ao PSNR, fez-se necessária a criação de uma nova fase, para que os grupos possam melhorar seus algoritmos e corrigir pontos fracos.

A nova fase, marcada inicialmente para agosto de 2002, tratará também dos outros tipos de avaliação: com “Referência Reduzida” e “Sem Referência”. Os grupos poderão disputar as modalidades de avaliação em que tiver em maior interesse ou melhores resultados. A Fundação CPqD disputará, em princípio, os processos de avaliação com “Referência Total”, do qual disputou a primeira fase, e com “Referência Reduzida”.

3.8 Propostas de Avaliação (Grupos do VQEG)

Os projetos apresentados a seguir fazem parte da primeira fase do processo de competição internacional pela padronização da avaliação objetiva de TV Digital, referentes à avaliação com “Referência Total”. São descritas, resumidamente, as propostas de sete grupos de pesquisa [12], sendo discutidas as formas de obtenção dos parâmetros de avaliação e as formas de relacionar estes parâmetros às bases subjetivas de dados.

3.8.1 CPqD

O modelo apresentado pelo CPqD [32] para o VQEG foi denominado de CPqD-IES (Avaliação de Imagem Baseada em Segmentação). As cenas naturais são segmentadas em regiões planas, de borda e de textura, e um conjunto de parâmetros objetivos é calculado para cada um destes contextos. Os tipos de regiões são caracterizados a seguir.

Região plana: É uma partição onde a variação do nível de cinza é baixa. É também denominada de região homogênea e o desvio padrão nesta região deve ser baixo. Uma

pequena região não plana imersa dentro de uma região plana também pode ser considerada como sendo parte da região plana.

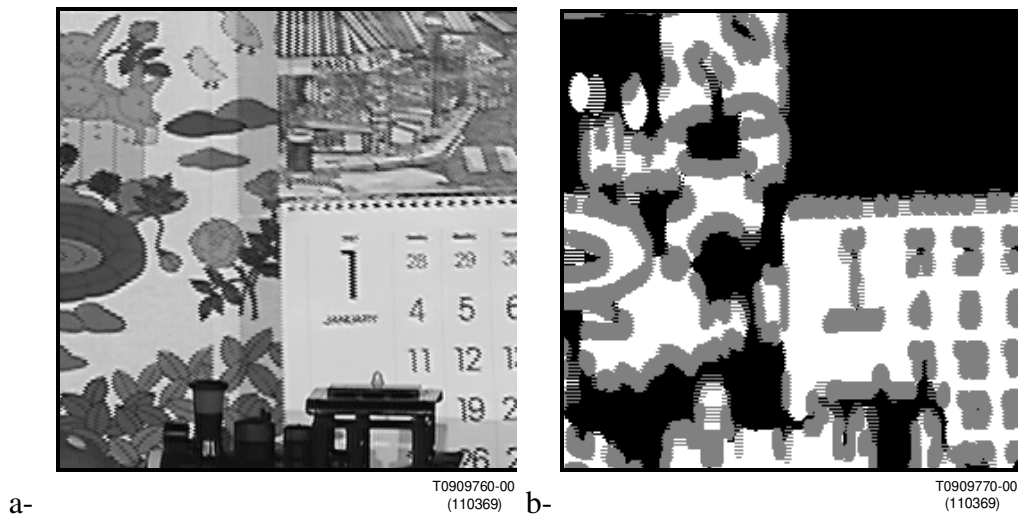
Região de textura: É uma partição onde existe grande variação do nível de cinza. É normalmente uma região complexa. O desvio padrão, em qualquer pequena vizinhança local e em qualquer posição da partição, deve ser alto.

Região de bordas: Aparece em duas ocasiões: 1) como linha divisória entre regiões planas, porém de intensidades diferentes e 2) como linha divisória entre regiões planas e regiões de textura. Apesar de existir borda entre duas regiões de textura diferentes, estas bordas não são importantes para a aplicação de avaliação objetiva contextual de *codecs*.

A separação em regiões é um dos aspectos mais importantes da metodologia. Uma distorção de efeito de bloco, por exemplo, pode ser medida através de um detector de borda aplicado nas regiões planas da cena de vídeo, onde a percepção visual desta distorção é mais notada. A complexidade computacional do método é reduzida pelo uso de medidas de baixa complexidade e pelas restrições computacionais dentro do contexto correspondente das cenas. Estes contextos são definidos através de um algoritmo de segmentação que é aplicado ao sinal de teste da entrada. O algoritmo de segmentação requer normalmente alta complexidade computacional, porém é executado apenas uma vez.

Para extração das regiões foram utilizados três algoritmos descritos sucintamente a seguir.

1º algoritmo: inicialmente, cada pixel é classificado, baseado na variância de seu brilho computado dentro de sua vizinhança, verificando se o pixel pertence ou não a uma região plana da imagem. A imagem binária resultante é então suavizada através de um filtro de média. O algoritmo também aplica à imagem original um detector de borda baseado em filtro recursivo. Os limites das regiões planas são classificados como regiões de bordas e as regiões de textura são as regiões remanescentes. A Figura 3.12 mostra a segmentação de uma imagem nas regiões definidas.



extraído de [32]

Figura 3.12- Segmentação por regiões (**a**) imagem original (**b**) imagem segmentada por regiões (planas, de textura e de borda)

2º algoritmo: a primeira parte deste algoritmo relaciona funções de pertinência, definidas em um intervalo (0,1), para cada um dos três contextos de classificação. Nas funções de pertinência das regiões planas, o valor de pertinência de um pixel é definido inversamente proporcional à variância do brilho computado dentro da vizinhança do pixel. O gradiente morfológico aplicado para esta função define a função de pertinência das regiões de borda. O complementar da união fuzzy entre essas duas funções de pertinência define a função membership da região de textura. Em um segundo passo, cada pixel é classificado como pertencente ao contexto de alto valor de membership junto com os três valores determinados no passo anterior.

3º algoritmo: este algoritmo, primeiramente, simplifica o componente de luminância reforçando as regiões homogêneas da imagem através da aplicação de um filtro suavizador com borda preservada. Posteriormente, é aplicado o algoritmo de watershed [33] com o gradiente morfológico da imagem simplificada. O watershed detecta regiões homogêneas, denotadas pelas chamadas “bacias hidrográficas”, com área maior que um limiar. As

regiões de textura são dadas pela erosão do complemento das regiões planas. As regiões de borda são as regiões remanescentes do processo.

Os parâmetros objetivos são obtidos para cada contexto (bordas, texturas e regiões planas) e de amostras de luminância e cromaticidade dos sinais de entrada e saída, após um registro temporal e espacial e correção de ganho e offset.

Conceitos importantes devem ser abordados na área de morfologia matemática, para entendimento do processo de segmentação, como rotulação, *flat-zones* e filtros conexos. A rotulação é uma ferramenta clássica de análise de imagens e sua função é atribuir a cada pixel da imagem binária um valor único que identifica cada região conexa da imagem. O conceito de *flat-zones* é uma extensão da rotulação quando aplicada em imagens em nível de cinza. Um filtro conexo é um filtro morfológico cuja partição em *flat-zones* da imagem resultante da aplicação do filtro é uma partição que contém a partição de *flat-zones* da imagem de entrada.

Um algoritmo clássico da morfologia matemática chamado *watershed* é utilizado. O *watershed* pode ser entendido através da simulação de uma inundação no modelo topográfico da imagem em nível de cinza. Define-se mínimo regional como sendo uma região conexa (um *flat-zone*) que possui *flat-zones* vizinhos com níveis de cinza estritamente mais altos. Simulando-se uma inundação de águas com colorações distintas, sendo cada uma um mínimo regional, o *watershed* permite determinar a linha de partições do modelo topográfico que divide as águas de modo que as cores não se misturem. O *watershed* para imagens em nível de cinza é normalmente aplicado em imagens de detecção de bordas, pois as linhas de partição desejadas são as linhas de bordas.

O algoritmo de avaliação com “Referência Total” proposto pelo CPqD trabalha com a simplificação e identificação dos *flat-zones*, classificando-os em regiões planas, de textura e de bordas. A partir da segmentação pode-se comparar os sinais de referência e degradado para obtenção dos parâmetros objetivos que irão qualificar o *codec*.

A relação entre cada parâmetro objetivo e o nível de degradação subjetivo é aproximada por uma curva estratégica, resultando em um nível de degradação extraído para cada parâmetro. O resultado final é conseguido através de uma combinação linear de níveis

de degradação, onde o peso de cada nível de degradação é proporcional à sua confiança estatística. Os resultados alcançados nesta contribuição mostram que o uso de medidas objetivas baseadas em regiões é mais confiável se comparado aos métodos globais de predição.

3.8.2 Tektronix/Sarnoff

A Tektronix [34,35] desenvolveu um algoritmo para medição da taxa de qualidade objetiva de uma imagem (PQR), baseada no sistema visual humano (HVS). Um pré-requisito essencial para a análise é a normalização do vídeo processado. O pré-processamento é necessário devido aos possíveis deslocamentos e cortes verticais e horizontais, além de mudanças nos níveis de luminância e cromaticidade causadas pela compressão e descompressão da imagem. A normalização não se torna necessária quando os parâmetros calculados (mostrados na Tabela 3.1) estão abaixo de um limite de tolerância.

Tabela 3.1- Parâmetros de normalização e tolerância

Parâmetro	Tolerância
Ganho de Luminância	< 0,2 dB
Ganho de cor (diferença)	< 0,2 dB
Nível DC de luminância	< 0.5% do máximo do sinal
Nível DC de cor (diferença)	< 0.5% do máximo do sinal
Atraso offset canal-para-canal	< 2 ns
Deslocamento horizontal de pixel	< 0,1 pixel
Deslocamento vertical de linha	< 0,1 linha

Como pode ser observado na Figura 3.13, o método de avaliação inicia-se com o chamado processamento frente-fim. O primeiro passo deste processamento é converter os valores de YCbCr para o sistema R', G' e B', que, logo em seguida, são convertidos para o

sistema R, G e B do monitor. Os componentes R, G e B são convertidos para o sistema X, Y e Z e estes para componentes L^* , u^* e v^* . Assim, as saídas deste primeiro bloco serão o componente Y, gerado na penúltima conversão, e u^* e v^* relativos aos componentes de cor da última conversão. Testes realizados pela Tektronix obtiveram melhores resultados quando utilizados Y, u^* e v^* , e o tempo computacional relativo a estas conversões é irrelevante.

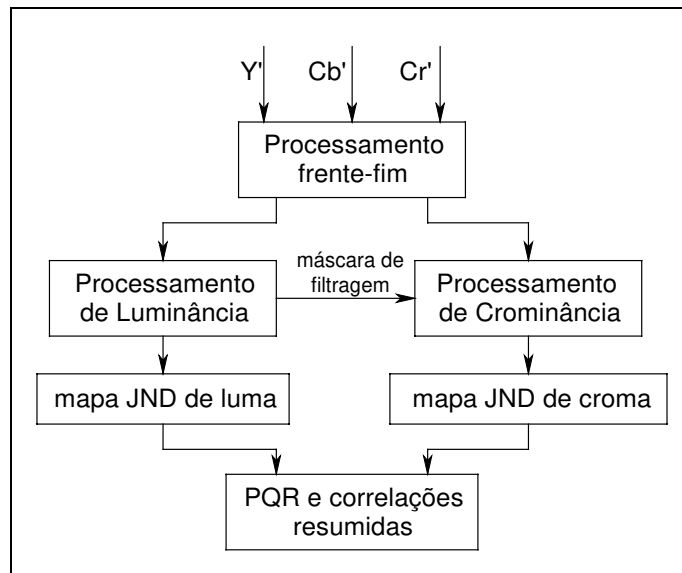


Figura 3.13- Fluxo do modelo para emular a visão humana

O componente Y de luminância gerado no bloco anterior é primeiro subjetivado através de uma curva não-linear. Após isso, cada campo de luma é filtrado e sub-amostrado em uma pirâmide Gaussiana de quatro níveis, de acordo com um modelo psicologicamente e fisiologicamente observado. Os quatro níveis da pirâmide são processados por filtros temporais e de contraste, sendo que em cada caso de contraste é realizada uma diferença local de valores de pixel por uma soma local, com escala apropriada. O valor absoluto da resposta de contraste é passado para o estágio seguinte, onde é convertido para unidades de JND (“just-noticeable difference”).

Por outro lado, os componentes de croma são processados de diversas maneiras. Diferenças de croma entre as imagens são usadas para definir a detecção de limiares em analogia ao contraste utilizado no modelo para a luminância.

Para cada campo na comparação das sequências de vídeo, os mapas JND de luminância e cromaância são combinados, gerando um mapa JND-total. Este JND-total é computado como uma combinação linear da soma e dos valores máximos dos mapas, pixel-a-pixel. E, desta forma, cada um dos três mapas (luminância, cromaância e luminância-cromaância combinadas) é reduzido para um único valor, chamado PQR (taxa de qualidade de imagem).

3.8.3 NHK

A NHK desenvolveu um sistema de avaliação de qualidade de imagem comprimida no qual a degradação de imagem é calculada em tempo real de acordo com o sistema visual humano. Neste sistema, a sensibilidade do ruído nas frequências espaço-temporais é considerada em relação ao brilho da imagem.

A Figura 3.14 mostra a estrutura do sistema de avaliação. Primeiramente, é realizada a diferença entre os sinais da imagem original e das sequências de teste (degradada). Esta imagem diferença é passada por um filtro digital 3D com a mesma resposta em frequência de dependência de visibilidade e brilho, apresentada nos gráficos da Seção 3.1. Em seguida, a diferença filtrada dos sinais é comparada com a percepção visual em cada pixel. Como resultado, uma expressão numérica de distorção baseada nos limites de percepção do olho humano é obtida.

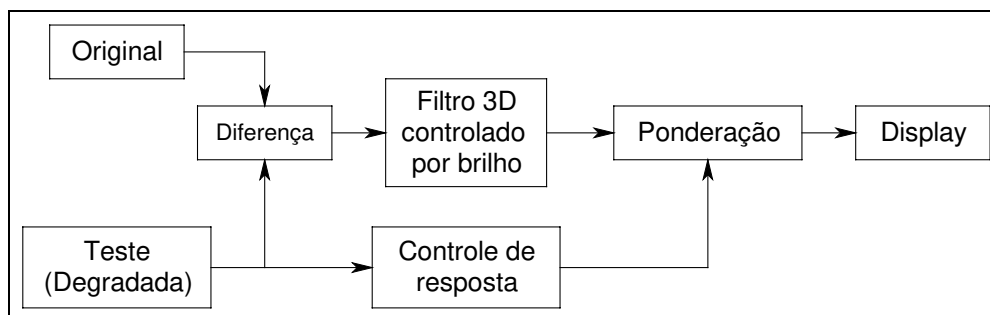
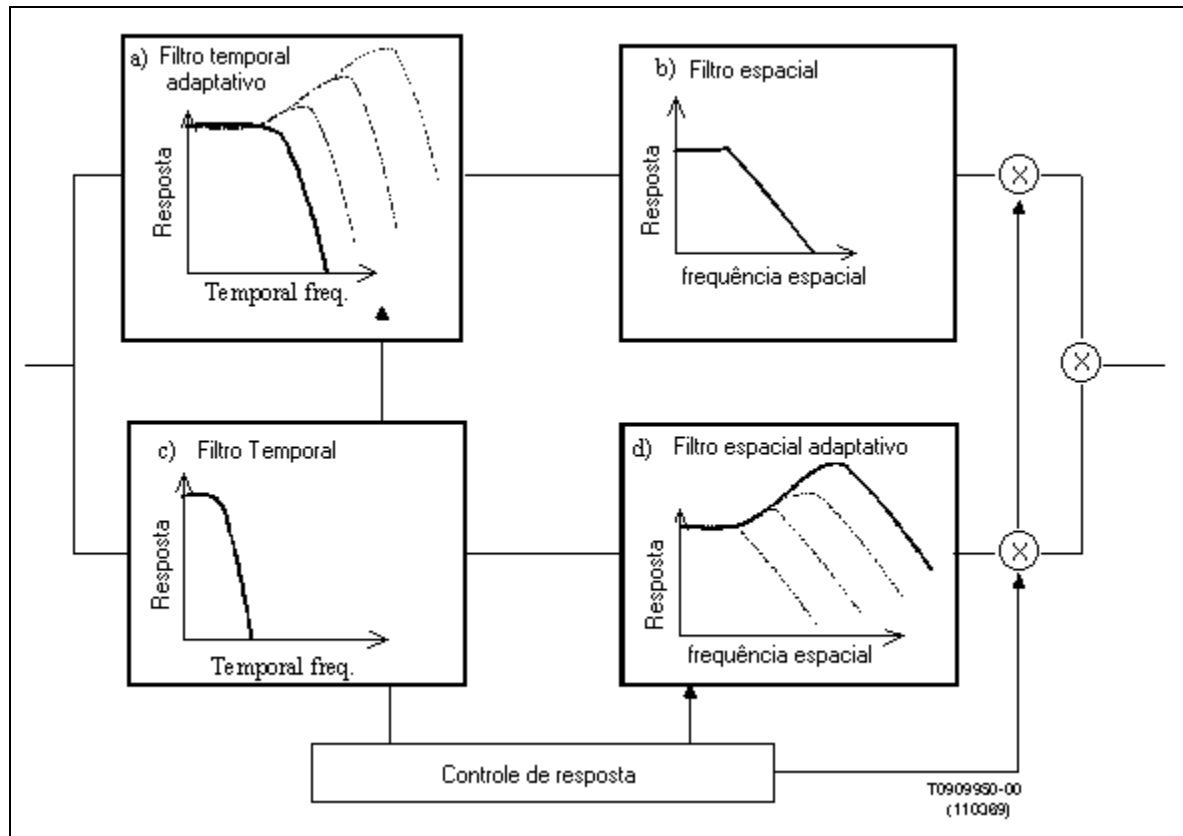


Figure 3.14- Estrutura do sistema de avaliação

A Figura 3.15 mostra a composição de filtros digitais 3D, com a resposta em frequência e mudança de sensibilidade de acordo com o brilho. Pela combinação de filtros espaciais e filtros temporais de acordo com o brilho da imagem, a resposta em frequência de visibilidade humana é emulada.



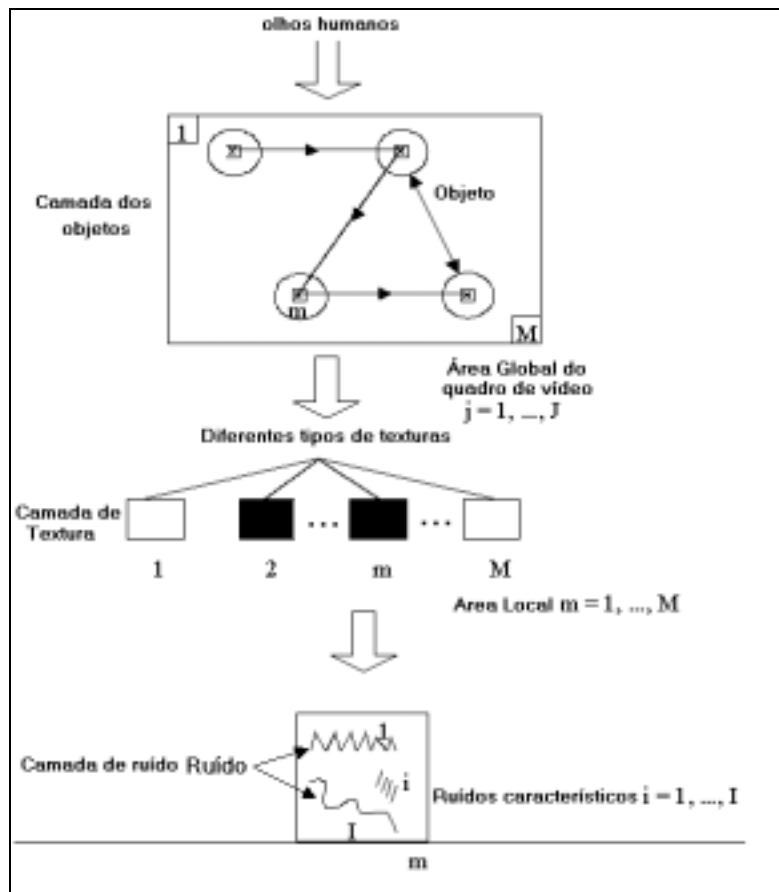
extraído de [4]

Figure 3.15- Estrutura do sistema de avaliação

Usando o sistema proposto, pode-se representar funções da habilidade visual humana, que muda substancialmente de acordo com o nível de brilho, realizando medidas com alta correlação com dados subjetivos, independente do tipo de sinal de vídeo.

3.8.4 KDD

A KDD propõe um novo esquema de avaliação de qualidade de vídeo baseado nas características de percepção visual humana. A Figura 3.16 mostra o esquema do modelo “três-camadas” para sinais de vídeo, como visto pelo olho humano.



extraído de [12]

Figura 3.16- Modelo “três-camadas” para um sinal de vídeo

Geralmente, o olho humano não consegue assistir um quadro inteiro de uma única vez, mas assiste áreas locais deste quadro, que se encontra ao redor de um ponto fixo dos olhos humanos. Os olhos humanos reconhecem a textura e também a qualidade da área dependendo do grau e características do ruído misturado com esta textura. O quadro inteiro

é entendido pelo movimento do ponto fixo dentre os objetos. Assim os componentes de imagem do quadro (camadas de objetos, textura e ruído) e a avaliação de qualidade são também conduzidas para todo o quadro ao mesmo tempo.

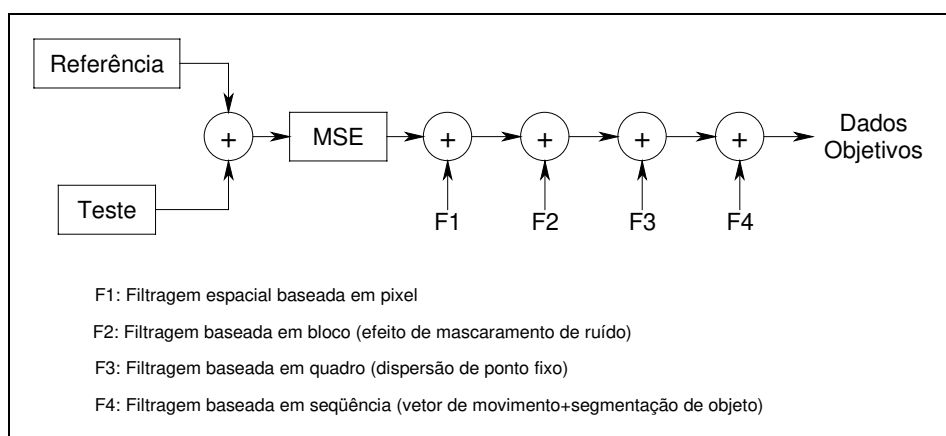


Figura 3.17- Esquema de avaliação

Como pode ser observado no esquema da Figura 3.17, é obtida a diferença entre a imagem de teste (degradada) e a de referência, e a partir deste valor são extraídos parâmetros que serão ponderados para se atingir a resposta de avaliação. Para esta ponderação, é realizada uma conversão em frequência para se classificar os ruídos. Posteriormente, a camada de textura, é classificada em vários grupos através de seus tipos de textura, e conseqüentemente, ruídos são mais ou menos ponderados de acordo com os tipos de textura. Finalmente, na camada de objeto, o grau de dispersão do ponto fixo é medido através do grau de complexidade das estruturas dos objetos nos quadros de vídeo. Desta forma, os ruídos do quadro total são ponderados de acordo com o declínio da sensibilidade do ruído causado por esta dispersão.

Os resultados obtidos foram comparados com dados de teste de avaliação subjetiva, que foram classificados seguindo a recomendação ITU-R 500-7, e foram constatados altos níveis de correlação desses valores com os obtidos pelo esquema de avaliação proposto.

3.8.5 NASA

O modelo proposto pela NASA [36,37] é chamado DVQ (Digital Video Quality) e sua métrica é uma tentativa de incorporar muitos aspectos de sensibilidade visual humana em um algoritmo de processamento de imagem simples. Simplicidade é um ponto importante, já que o ideal seria uma métrica que funcionasse em tempo real e assim requeresse recursos computacionais modestos.

Um dos mais complexos elementos consumidores de tempo presentes nas outras propostas são as operações de filtragem espacial empregadas para implementar os múltiplos de filtros passa-banda que são características do sistema visual humano. Este passo é acelerado através do uso da Transformada Discreta de Cosseno (DCT) para esta decomposição em canais espaciais. Isto gera uma vantagem poderosa já que existem eficientes hardwares e softwares que realizam esta transformação, e também porque em muitas aplicações a transformada já deve ter sido feita como parte do processo de compressão.

A entrada para a métrica é o par de seqüências das imagens coloridas de teste e de referência. O primeiro passo consiste de várias amostragens, cortes e transformações de cor que servem para restringir o processamento para uma região de interesse e para expressar as seqüências em um espaço de cor perceptual. As seqüências são então subjetivadas para a “blocagem”, processo denotado pelo BLK da Figura 3.18, e em seguida convertidas para coeficientes de DCT.

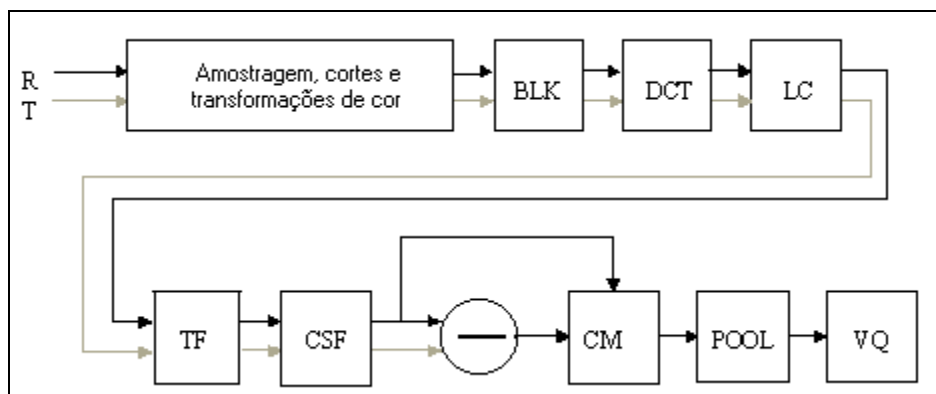


Figura 3.18- Passos para o processamento do DVQ

Os coeficientes da DCT são convertidos em unidades de contraste local (LC) da seguinte maneira. Extraem-se os coeficientes DC de todos os blocos. Estes são então filtrados, usando um filtro IIR passa-baixa de primeira ordem, com ganho 1 e constante de tempo τ_1 . Os coeficientes DCT são então divididos pelos coeficientes DC, filtrados em uma base bloco-por-bloco. Finalmente, os quocientes são ajustados pela magnitude relativa de seus coeficientes correspondendo a uma função base de contraste unitário.

O próximo passo do processo é a filtragem temporal (TF), onde ambas as seqüências são subjetivadas. O filtro temporal é um IIR de segunda ordem. O uso do filtro minimiza o número de quadros que devem ser armazenados na memória. Posteriormente, os coeficientes DCT, expressos em forma de contraste local, são convertidos em unidades de JNDs pela divisão de seus respectivos limiares espaciais. Isto implementa a parte espacial da função de contraste sensitivo (CSF). Após a conversão, os coeficientes das duas seqüências são subtraídos para produzir uma seqüência diferença. Esta diferença é então subjetivada com a operação de máscara de contraste (CM), que depende da seqüência de referência. Finalmente, as diferenças mascaradas são armazenadas de várias maneiras para ilustrar o erro perceptual sob várias dimensões (POOL), e o erro armazenado deve ser convertido para qualidade visual (VQ).

3.8.6 KPN

A KPN [38] desenvolveu um método de medida da qualidade de percepção de vídeo (PVQM) que utiliza os mesmos conceitos de medida para avaliação de qualidade de voz (PSQM). O método foi desenvolvido para confrontar distorções espaciais, temporais e espaço-temporais com aquelas encontradas em condições de erro.

Primeiramente, as seqüências de vídeo original e degradada são reamostradas para o formato 4:4:4, YCbCr. Um alinhamento espaço-temporal de luminância é incluído no algoritmo. O PVQM usa uma adaptação especial de brilho e contraste das seqüências de vídeo distorcidas, para limitar o impacto das mudanças globais no brilho e contraste. O alinhamento espaço-temporal é realizado através do chamado “casamento de bloco”. Neste processo, são encontrados os pares correspondentes das imagens originais e degradadas. Além disso, são realizadas as correções dos possíveis offsets (deslocamentos vertical e horizontal dos quadros), gerados na codificação do sinal.

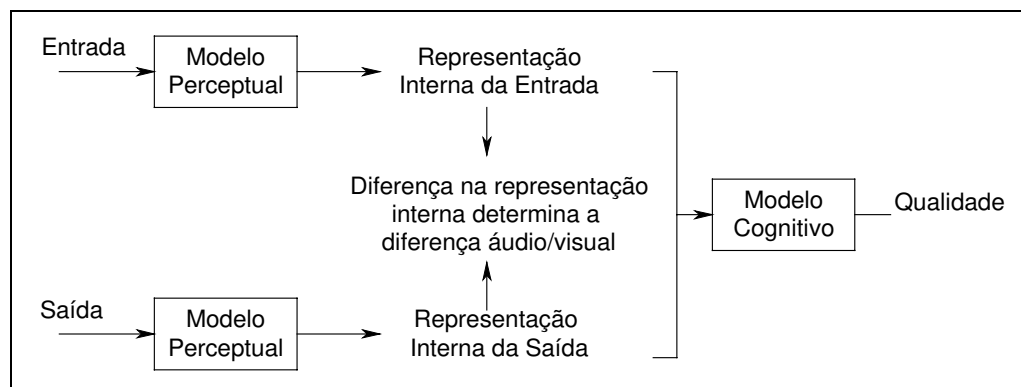


Figura 3.19- Esquema do modelo proposto

Conforme mostrado na Figura 3.19, os sinais originais e degradados são processados por modelos perceptuais e, em seguida, estes resultados e a diferença entre eles são processados por um modelo cognitivo.

A análise da luminância espacial é baseada na detecção de bordas, enquanto que a parte temporal é baseada na análise das diferenças entre os sinais Y (luminância) dos quadros. É conhecido que o Sistema Visual Humano (HVS) é muito mais sensível à

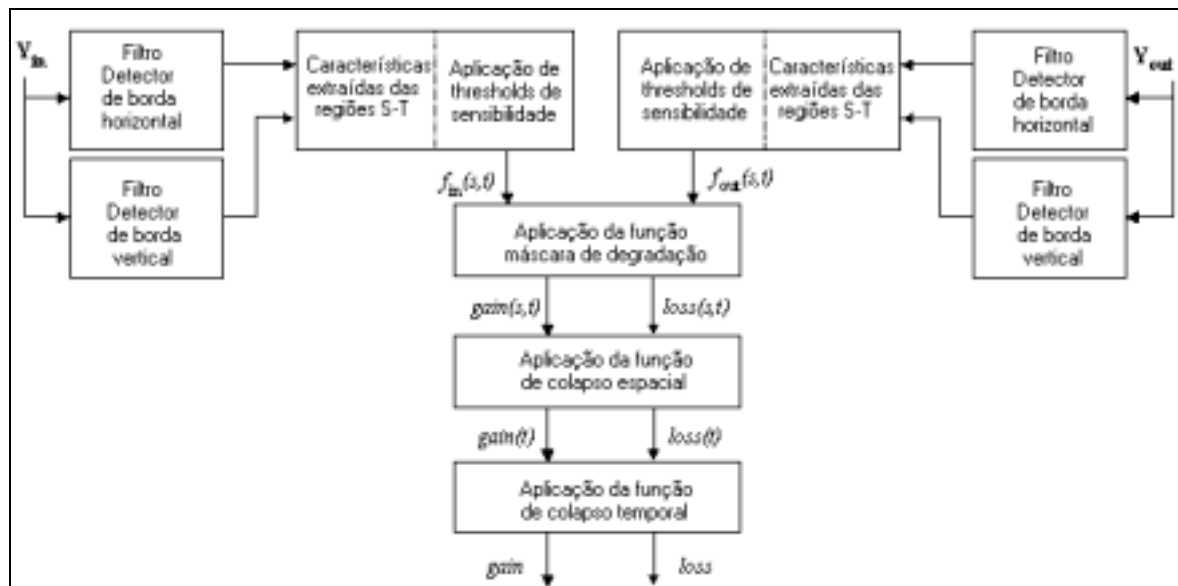
nitidez dos componentes de luminância do que a componentes de croma. Entretanto, o HVS possui uma função de contraste sensitivo que decresce para altas frequências espaciais. Estes conceitos do HVS refletem no primeiro passo do algoritmo PVQM, que provêm uma aproximação de primeira ordem das funções de contraste sensitivo dos sinais de luminância e croma.

Como segundo passo, as bordas da luminância Y são calculadas através do cálculo do gradiente local do sinal de luminância em cada quadro. Em seguida, é computado um erro de croma com ponderação média do erro de cor dos componentes Cb e Cr, normalizados no local de saturação com uma dominância do componente Cr. No último passo os três diferentes indicadores são mapeados em um indicador de qualidade simples, usando uma regressão linear múltipla, o qual correlaciona bem a qualidade de vídeo percebida subjetivamente.

3.8.7 NTIA

Um algoritmo de baixa complexidade e que possibilita aplicações de monitoramento em tempo real foi desenvolvido pelo NTIA [39,40] para métrica de qualidade de vídeo (VQM). O VQM consiste de uma combinação linear de quatro parâmetros, que têm sido otimizados para um padrão de distância de visão de seis vezes a altura da imagem. Três parâmetros são extraídos dos gradientes espaciais do componente de luminância (Y), enquanto um parâmetro é obtido a partir do vetor formado pelos componentes de croma (Cb,Cr).

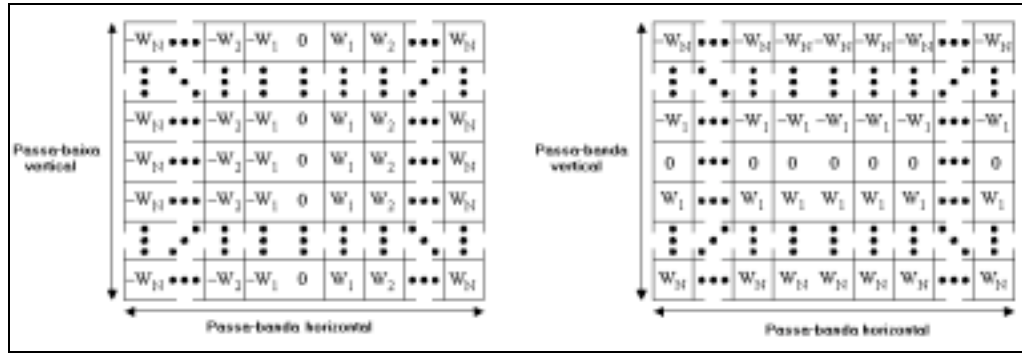
As amostras de entrada e saída de vídeo devem ser calibradas antes do processamento, para compensar a possível presença de ganho e nível de offset e para realizar os devidos registros espaço-temporais das imagens.



extraído de [12]

Figura 3.20- Algoritmo usado para extrair parâmetros do gradiente espacial

O algoritmo utilizado para extrair parâmetros do gradiente espacial é dado na Figura 3.20. Os componentes de entrada e saída de vídeo são processados usando filtros de detecção de bordas verticais e horizontais, cujo formato pode ser observado na Figura 3.21. Em seguida, os fluxos de vídeo processados são divididos em regiões espaço-temporais, de onde são extraídas características para quantificar a atividade espacial como uma função de orientação angular. Assim, essas características são separadas para emular limiares perceptíveis à visão humana. Em seguida, as distorções de vídeo, devido aos ganhos e perdas nos valores característicos, são calculadas para cada região espaço-temporal, comparando-se os valores de entrada e saída com as relações funcionais que emulam as degradações visuais. Estas distorções são então armazenadas para produzir parâmetros de qualidade para um vídeo de duração aproximada de 5 a 10 segundos.



extraído de [12]

Figura 3.21- Filtros para detecção de bordas

O primeiro parâmetro, f_1 , é computado simplesmente como o desvio padrão dado por:

$$f_1 = \{std[R(i, j, t)]\}_p : i, j, t \in \{região\ espaço - temporal\} \quad (3.2)$$

onde:

$$R(i, j, t) = \sqrt{H(i, j, t)^2 + V(i, j, t)^2}$$

H e V – respostas do filtro de realce de bordas

Um segundo parâmetro, f_2 , é obtido a partir da relação entre a imagem com gradiente vertical e horizontal e a imagem correspondente ao gradiente diagonal. O parâmetro f_2 é sensetivo a mudanças na distribuição angular de atividade espacial dentro de uma dada região espaço-temporal.

Como observado na Figura 3.20, os parâmetros f_1 e f_2 são representados por f_{in} (da imagem de referência) e f_{out} (da imagem degradada). A ponderação destes dois parâmetros, para uma nota final de avaliação, é conseguida através de um treinamento, realizado a partir de uma função de degradação subjetiva. A partir disso, são obtidas funções de ganho e perda, que passam por funções de colapso espacial e temporal responsáveis por isolar as degradações espaciais das temporais, permitindo uma nova ponderação.

Uma das grandes novidades deste método em relação aos outros é a análise de degradações no domínio do tempo, o que pode gerar parâmetros mais confiáveis e métricas mais próximas do sistema visual humano.

3.8.8 PSNR usando Wavelet

Em [41] é proposto um método para cálculo de PSNR usando a transformada de wavelet [42]. Este método não faz parte das propostas submetidas ao VQEG, mas representa uma forma inovadora de tratar o problema e que pode ser muito promissora.

Como pode ser observado na Figura 3.22, aplica-se a transformada de wavelet 2D para cada quadro das seqüências de vídeo, usando o filtro biortogonal de Daubechies como filtro de wavelet e o nível 3. Cada quadro é dividido em 10 sub-bandas pela decomposição de wavelet de três níveis.

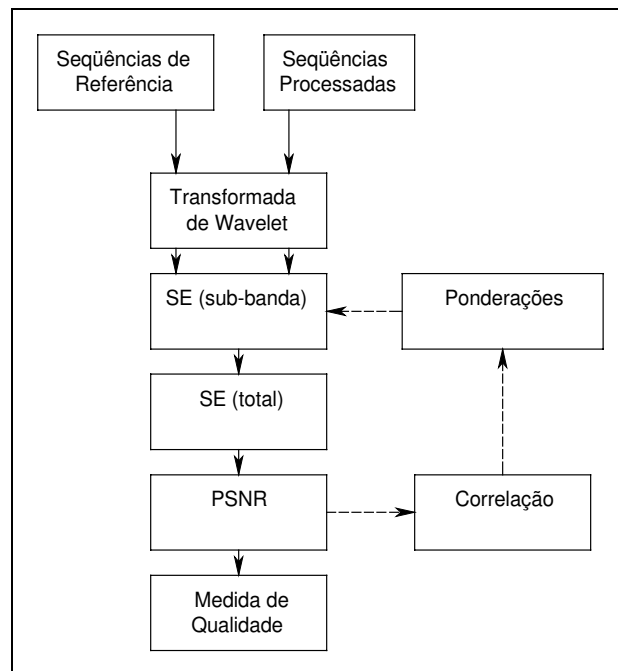


Figura 3.22 – Procedimento de avaliação objetiva usando wavelet

A partir da decomposição de wavelet, torna-se possível considerar a sensibilidade do sistema visual humano para estímulos de diferentes frequências espaciais. Por exemplo, na imagem wavelet decomposta por três camadas (Figura 3.23), LL3 representa regiões de frequência espacial baixa na imagem. Por outro lado, HH1 representa regiões de alta frequência espacial.

LL3	HL3	HL2	HL1
LH3	HH3		
LH2		HH2	
LH1			HH1

Figura 3.23 – Decomposição três-escalas de wavelet

Após a transformada, o erro quadrático da s-ésima sub-banda, SE_s , é calculado pela subtração do sinal processado do sinal de referência como segue:

$$SE_s = \sum_{i=0}^{K-1} \sum_{j=\alpha}^{N_s+\alpha-1} \sum_{k=\beta}^{M_s+\beta-1} (x_{i,j,k} - y_{i,j,k})^2 \quad (3.3)$$

onde a s-ésima sub-banda consiste dos $N_s \times M_s$ coeficientes de wavelet começando na posição (α, β) ; $x_{i,j,k}$ são os coeficientes da sequência de vídeo original ou de referência; e $y_{i,j,k}$ são os coeficientes da sequência processada.

Finalmente, o erro quadrático total SE_{total} é calculado como uma soma ponderada dos $SE_{sub-banda}$ como mostrado a seguir.

$$SE_{total} = \sum w_s . SE_s \quad (3.4)$$

onde w_s é um peso e SE_s é o erro quadrático da s-ésima sub-banda.

O PSNR ponderado é computado como:

$$PSNR(dB) = 10\log_{10} \left(\frac{255^2}{\frac{SE_{total}}{MNK}} \right) \quad (3.5)$$

onde M e N são as dimensões do quadro de vídeo, e K é o número de quadros sob análise.

Capítulo 4

Medidas Objetivas

A partir do estudo realizado das técnicas de avaliação com referência total, disponibilizadas pelos diversos grupos do VQEG, e outros métodos presentes na literatura, são propostos aqui alguns descritores de qualidade de vídeo digital. Estes descritores resultam em parâmetros de duas classes de avaliação, “Em-Laboratório”, com referência total, e principalmente “Em-Serviço”, com referência reduzida.

A idéia foi buscar resultados com correlações compatíveis às obtidas pelos métodos com referência total propostos durante a primeira fase do VQEG, além de encontrar descritores com referência reduzida eficazes e que utilizassem ferramentas baseadas em Morfologia Matemática. Além disso, trabalhou-se com todos os descritores propostos em [43].

Apresenta-se neste capítulo as técnicas de comparação dos descritores com a base subjetiva, além do processo de otimização por redes neurais utilizado.

Todas as técnicas propostas a seguir foram implementadas em linguagem Matlab e processadas para comparação com bases de dados subjetivas fornecidas pelo CPqD. Assim, foi criada uma biblioteca de funções de avaliação de qualidade de vídeo, mostradas no Apêndice I, contendo as propostas implementadas. Para criar a documentação da “Toolbox”, foi utilizado o sistema de desenvolvimento científico ADESSO [44], baseado em linguagem XML.

Apesar de existirem fundamentos teóricos para a utilização dos parâmetros como métodos de avaliação, alguns deles podem não emular muito bem o sistema visual humano. Os testes e medidas, juntamente com os resultados obtidos, estão presentes no Capítulo 5.

Como os métodos de avaliação com “Referência Reduzida” exigem uma resposta quase que simultânea da qualidade do sistema, procurou-se também implementações com baixo esforço computacional, apesar do aumento crescente da velocidade de processamento das máquinas atuais.

4.1 Descritores implementados

Os descritores para avaliação objetiva de codificadores de vídeo aqui apresentados utilizam variadas técnicas e algoritmos. As imagens original e degradada podem ser analisadas no domínio do tempo ou da frequência, podem ser processados somente seus componentes de Luminância, ou ainda, extraídas somente as regiões planas ou de borda.

Algumas das técnicas são propostas tanto para avaliação com “Referência Total” como “Referência Reduzida”, sendo que todos os descritores são apresentados neste tópico. As formas de extração das informações de referência serão diferenciadas a partir das funções Matlab implementadas e contidas no Apêndice I.

Cada descritor é primeiro definido teoricamente, sendo apresentadas suas equações e procurando sua relação com o processo de compressão. Em seguida, são apresentados exemplos numéricos ou com imagens reais com diferentes tipos de degradação.

4.1.1 PSNR - Pico da Relação Sinal Ruído

A relação sinal-ruído (SNR) pode gerar um importante descritor da qualidade de imagens digitais reconstruídas e comparadas com uma imagem original. A partir de SNR, extrai-se o pico da relação sinal ruído (PSNR), que é atualmente o mais importante descritor de qualidade entre duas imagens [45]. Esta medida tem sido usada por muito tempo e apresenta-se como um método simples e robusto, tendo superado modernos e complexos algoritmos na primeira fase de testes do VQEG. Esta métrica pode ser aplicada apenas para sistemas de avaliação com referência total, e é o único, até o momento, recomendado pela ITU.

O pico da relação sinal-ruído é a razão entre o pico do sinal e a raiz quadrada média do ruído observado entre o sinal de vídeo processado e o sinal de vídeo de referência.

O cálculo do PSNR envolve os passos descritos a seguir.

Seleção da seqüência de vídeo processada: Deve-se selecionar uma seqüência de M quadros de vídeo processados sob o qual o PSNR será calculado. Usando o sistema de coordenadas da Recomendação 601 [4], a seqüência dos componentes de luminância Y processados será denotada como $\{\mathbf{Y}_{\text{proc}}(t_n)\}_{n=0}^{M-1}$.

Normalização da seqüência de vídeo processada:

- 1) Encontrar o conjunto de quadros de vídeo de referência correspondentes às seqüências de vídeo processadas (registro temporal ou medida de tempo de espera).
- 2) Executar o registro espacial horizontal e vertical da seqüência de vídeo processada.
- 3) Executar a correção de ganho e offset da seqüência de vídeo processada.

Os procedimentos de normalização devem ser executados quando os níveis de tolerância ultrapassarem os valores estabelecidos na Tabela 4.1.

Tabela 4.1: Exigências para normalização

Parâmetro	Tolerância
Ganho de Luminância	< 0.2 dB
Ganho de Cor (diferença)	< 0.2 dB
Nível de Luminância DC	< 0.5 % do máximo do sinal
Deslocamento horizontal	< 0.1 pixel
Deslocamento vertical	< 0.1 linha

A seqüência combinada de M quadros de vídeo de referência (respectivos aos M quadros de vídeo processados), usando o sistema de coordenadas da Rec. 601 será denotada por $\{\mathbf{Y}_{\text{ref}}(t_n - d)\}_{n=0}^{M-1}$. Neste caso, d é o tempo de espera de um sistema de vídeo unidirecional determinado durante o sub-passo 1 do registro temporal. A seqüência normalizada final dos M quadros de vídeo processados será denotada por $\{\hat{\mathbf{Y}}_{\text{proc}}(t_n - d)\}_{n=0}^{M-1}$.

Seleção da região espacial de interesse (SROI):

Alguns sistemas de transmissão unidirecional de vídeo podem não produzir uma imagem válida durante todo o intervalo disponível. Nestes sistemas a SROI deve ser reduzida para que os valores de medida PSNR não sejam comprometidos. O Item 3.4.1 especifica este processo.

Cálculo dos valores de PSNR quadro a quadro:

Os valores de PSNR quadro a quadro são calculados através da seguinte equação:

$$PSNR(t_n) = 20 \log_{10} \left[\frac{Y_{\text{peak}}}{\sqrt{\frac{1}{(N_h N_v)} \sum_{j=O_h}^{O_h+N_h-1} \sum_{i=O_v}^{O_v+N_v-1} [Y_{\text{ref}}(i, j, t_n - d) - \hat{Y}_{\text{proc}}(i, j, t_n)]^2}} \right], \quad (4.1)$$

onde Y_{peak} é definido como sendo 255.

Cálculo da seqüência PSNR

O valor do pico da relação sinal ruído para uma seqüência com M quadros será calculada por:

$$PSNR = 20 \log_{10} \left[\frac{Y_{\text{peak}}}{\sqrt{\frac{1}{(M N_h N_v)} \sum_{n=0}^{M-1} \sum_{j=O_h}^{O_h+N_h-1} \sum_{i=O_v}^{O_v+N_v-1} [Y_{\text{ref}}(i, j, t_n - d) - \hat{Y}_{\text{proc}}(i, j, t_n)]^2}} \right], \quad (4.2)$$

onde M é o número de quadros da seqüência

Exemplo: Na Figura 4.1 a seguir, são mostrados os componentes de luminância de duas imagens, uma original e a outra sua respectiva degradada, e logo abaixo é indicado o valor PSNR resultante entre elas.



Figura 4.1 – Imagens para cálculo de PSNR (a) imagem original (b) imagem degradada

4.1.2 Regiões planas

As distorções que ocorrem nas regiões planas de uma imagem podem ser importantes fatores no processo de avaliação de qualidade. Os codificadores de vídeo, geralmente, procuram representar estas regiões, por serem homogêneas, com uma quantidade reduzida de bits. Diante disso, as imagens reconstruídas podem ter suas regiões homogêneas alteradas, principalmente com a blocagem.

As regiões planas são definidas como as porções da imagem que possuem pequena variação espacial local na sua intensidade. Uma técnica eficiente para extração destas regiões usa o operador morfológico watershed com marcadores [33]. Primeiramente, é aplicado um filtro sequencial para suavizar levemente a imagem. Uma imagem de marcadores é obtida extraindo os mínimos regionais do gradiente morfológico suavizado. Em seguida, o algoritmo de watershed é aplicado no gradiente morfológico da imagem simplificada, usando a imagem de marcadores. O watershed detecta as regiões homogêneas, denominadas bacias de contenção, com contraste mínimo especificado. As regiões planas são as bacias de contenção com área superior a um limiar.

Finalmente, as regiões planas podem ser comparadas para obtenção de um descritor único. Usando “Referência Total”, a forma de comparação que obteve melhores resultados foi à relação entre a área do erro das regiões planas das imagens original e degradada, e o perímetro das regiões planas extraídas da imagem original. Tal comparação está especificada na equação a seguir:

$$estimador_regplanas = \frac{\sum_{j=1}^N \sum_{i=1}^M abs(seg_{ref}(i, j) - seg_{deg}(i, j))}{\sum_{j=1}^N \sum_{i=1}^M (seg_{ref} \oplus H_1 - seg_{ref})(i, j)} \quad (4.3)$$

sendo: seg_{ref} e seg_{deg} - as imagens binárias respectivas às regiões planas das imagens de referência e degradada; H_1 é um elemento estruturante tipo *cruz* com raio 1; N e M são as dimensões dos quadros da sequência; e \oplus é um operador de dilatação.

Para “Referência Reduzida”, calcula-se o valor absoluto médio da diferença entre o somatório de intensidades dos pixels pertencentes às imagens seg_{ref} e seg_{deg} , como mostrado na Equação 4.4 a seguir.

$$Descritor_regplanas = \frac{1}{N.M} .abs\left(\sum_{j=1}^N \sum_{i=1}^M seg_{ref}(i, j) - \sum_{j=1}^N \sum_{i=1}^M seg_{deg}(i, j)\right) \quad (4.4)$$

Exemplo: Aplica-se o algoritmo de extração de regiões planas em uma imagem muito usada em testes de qualidade de vídeo, a imagem *Mobile&Calendar*. Como pode ser observado na Figura 4.2, são extraídas as regiões da imagem original com variações mais suaves de intensidade.

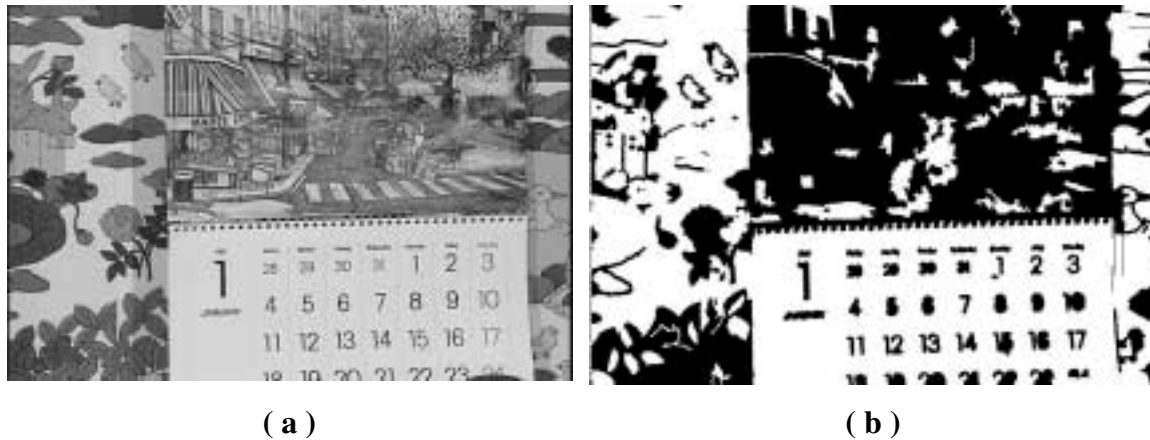


Figura 4.2- Exemplo do descritor de regiões planas (a) luminância da imagem original
(b) regiões planas extraídas da imagem 4.2a

4.1.3 Dimensão fractal

A robustez de uma medida realizada em uma imagem compreende sua invariância a algumas transformações, tais como rotação e translação. Uma medida obtida para uma dada escala deve também ser relacionada com aquelas obtidas por outras escalas. Além disso, medidas robustas devem realizar comparação entre imagens levemente diferentes.

Os parâmetros de regiões de textura são de particular interesse para a caracterização de uma imagem. Muitas texturas naturais apresentam estruturas em todas as escalas ou mostram um alto grau de granularidade. Uma métrica executada sob tais objetos deve ter, portanto, dependência de escala e tender a infinito quando o tamanho do objeto de teste tender a zero. O termo *fractal* foi proposto por Mandelbrot [48] para qualificar estes objetos. Um parâmetro chave de um objeto fractal é sua *dimensão fractal*, que é um número não-inteiro e que pode ser usado como um descritor adicional de textura. Existem muitas diferentes definições para a dimensão fractal, e isto tem introduzido uma grande quantidade de versões na literatura. Este problema foi tratado por Carr e Benzer [49].

Neste trabalho, a dimensão fractal é calculada utilizando o modelo Minkowski Sausage [50]. Este método baseia-se na resposta de uma imagem a dilatações morfológicas sucessivas, com elementos estruturantes cada vez maiores.

O modelo Minkowski revela uma medida robusta da dimensão fractal que é obtida considerando um disco plano He de diâmetro e . Devido à natureza multi-escala do algoritmo, dilatações podem ser realizadas para todos os tamanhos de elemento estruturante. Com isso, a dimensão fractal pode ser calculada através do volume de superfícies externas, como pode ser visto nas equações a seguir.

$$D_{fractal} = - \left(-1 + \frac{\Delta(\log(S_e))}{\Delta(\log(e))} \right) \quad (4.5)$$

$$S_e = \frac{V[(f \oplus He) - f]}{e} \quad (4.6)$$

sendo: V – o volume de uma superfície; f – uma imagem de entrada binária ou em nível de cinza; He – um elemento estruturante de raio e .

Desta forma, é calculado o vetor de medidas do decaimento logaritmo natural da subtração entre as áreas da imagem original dilatada e a da imagem original, dividido pelo diâmetro do disco Euclidiano usado na dilatação. O valor de $D_{fractal}$ é extraído a partir da inclinação predominante da curva formada pelo $\log(S_e)$ versus e .

A comparação final entre os valores de dimensão fractal dos quadros de vídeo é realizada através da relação entre as dimensões fractais das imagens original e degradada, que pode ser vista na equação a seguir.

$$Descriptor_{fractal} = \frac{D_{fractal_orig}}{D_{fractal_deg}} \quad (4.7)$$

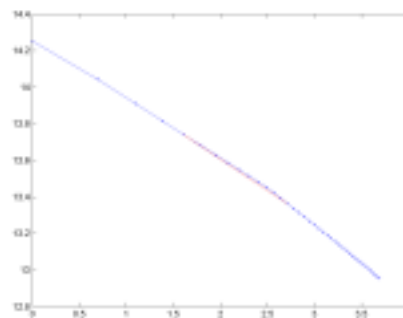
sendo: $D_{fractal_orig}$ e $D_{fractal_deg}$ - as medidas de dimensão fractal das imagens original e degradada respectivamente.

Na função *vqfractal.m* implementada, o diâmetro do elemento estruturante varia de 1 a 40, e é possível fornecer a região do gráfico *Vxe* onde será calculada a inclinação. Um intervalo típico do raio do elemento estruturante para medir esta inclinação está entre 5 e 15. Esta inclinação ou coeficiente angular gera a dimensão fractal da imagem em nível de cinza de entrada (Equação 4.5).

Exemplo: Utilizando a função *vqfractal.m* implementada, extrai-se o gráfico *Vxe* e indica-se o valor da inclinação predominante.



(a)



dimensão fractal = 1.3415

(b)

Figura 4.3- Análise de imagens pela dimensão fractal (a) imagem original (b) gráfico *Vxe* com inclinação predominante em destaque

4.1.4 Descritor de Blocagem

A blocagem ou efeito “tiling” é percebida mais facilmente em relação a outras formas de degradação, e pode ser considerada de forma isolada. A blocagem é normalmente detectada pela procura de transições verticais e horizontais abruptas no vídeo codificado.

Como uma das partes mais importantes dos processos de compressão do formato MPEG é a transformada de cossenos (DCT), que utiliza blocos 8x8, torna-se conveniente realizar uma análise de blocos com estas mesmas dimensões. Quando os coeficientes menos significativos da DCT são desprezados, a imagem reconstruída pode conter padrões retangulares predominantes da blocagem.

Em [27] é proposta uma análise dos componentes harmônicos do gradiente da imagem original sem referência. Quando o gradiente da luminância da imagem é submetido à transformada de cossenos, as pequenas discontinuidades dos blocos da DCT são reveladas como padrões pseudo-periódicos no domínio espacial. Quando este padrão é transformado (DFT) para o domínio da frequência, harmônicas de frequências fixas são obtidas. O comprimento dessas harmônicas é proporcional à quantidade de blocagem na imagem codificada.

Na proposta deste trabalho, utilizou-se a mesma técnica de análise de blocos 8x8, extraíndo-se primeiramente o gradiente do componente de luminância de imagem. Mas para detecção dos componentes de blocagem, foi utilizada a transformada de watershed com marcadores. Os marcadores foram definidos como pontos localizados nos centros de cada bloco 8x8. Tal operação está representada na equação a seguir.

$$ws = watershed(f_{grad}, f_m) \quad (4.8)$$

onde: f_{grad} – é o gradiente da imagem ; e f_m a imagem de marcadores

Após a transformada de watershed, os efeitos de blocagem aparecem com alta intensidade, principalmente nas regiões homogêneas da imagem. Este efeito das linhas de *watershed* pode ser observado na imagem (*ws*) da Figura 4.4d, onde foi aplicado o algoritmo proposto na imagem degradada mostrada na Figura 4.4c. Por outro lado, a imagem original, mostrada na Figura 4.4a, não apresentou os efeitos quadriculados na Figura 4.4b, e, portanto, não possui blocagem.

Para extração das informações de referência, realiza-se uma análise por blocos, cujas dimensões podem variar de acordo com a quantidade de dados de referência permitida. Esta análise é regida pela seguinte equação:

$$S(i, j) = \sum_{x=i.range1}^{(i+1).range1-1} \sum_{y=j.range2}^{(j+1).range2-1} ws(x, y) \quad (4.9)$$

com $i \in [0, N_i]$ e $j \in [0, N_j]$, onde N_i e N_j são o número de blocos na horizontal e vertical, respectivamente; e $range1$ e $range2$ são as dimensões do bloco de análise.

Para a avaliação, portanto, dispõe-se de duas matrizes contendo o somatório dos pixels pertencentes aos blocos. Testes preliminares revelaram que, utilizando blocos 32x32 para extração dos dados de referência, são alcançados os melhores resultados. Além disso, o uso destas dimensões obedece aos limites de referência estabelecidos pelo VQEG.

Finalmente, o descritor final é calculado através da seguinte equação:

$$Descritor_block = \frac{1}{N_i \cdot N_j} \sum_{j=1}^{N_j} \sum_{i=1}^{N_i} abs(S_{orig}(i, j) - S_{deg}(i, j)) \quad (4.10)$$

onde: S_{orig} e S_{deg} – são as matrizes de referência das imagens original e degradada; e N_i e N_j são o número de blocos na horizontal e vertical.

Para imagens com dimensões 480x720 pixels, o tamanho do vetor de referência é de 345 posições para cada quadro de análise. Cada posição exige 4 bytes ou 32 *bits*, e assim cada quadro precisa de 5.5 *Kbits*. A partir das taxas de referência VQEG especificadas no Item 3.4.2, constata-se que este método é aplicável para as taxas de 56 *Kbits/s* e 256 *Kbits/s*, para análise quadro a quadro. A referência de 10 *Kbits/s* pode também ser utilizada, mas para uma análise de 15 *quadros/s*.

Exemplo: São apresentados a seguir dois resultados do algoritmo para detecção de blocagem proposto neste trabalho (função *vqblock.m*). O primeiro resultado gerado pela imagem original e o segundo pela mesma imagem degradada.

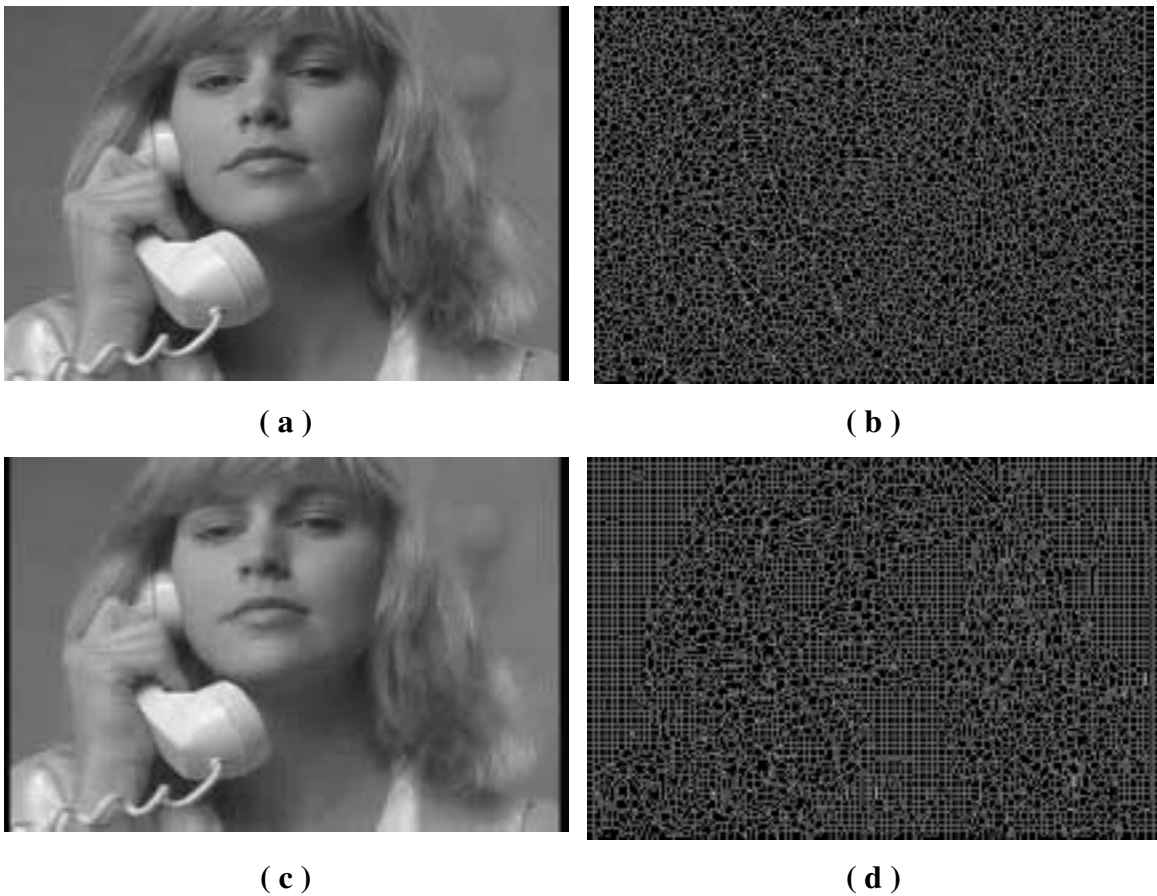


Figura 4.4- Detecção de blocagem em imagens reais (a) imagem original (b) resultado do processo aplicado à imagem 4.10a (c) imagem degradada (d) resultado do processo aplicado à imagem 4.10c

4.1.5 Descritores de Fourier

Os codificadores de vídeo têm dificuldade em comprimir pequenos detalhes ou as bordas de um quadro de vídeo. Estes pequenos detalhes repercutem em componentes de média e alta frequência, que sofrem consideráveis variações diante de processos de degradação. Uma alternativa para utilizar esta característica no problema de avaliação de qualidade, é extrair um valor obtido de uma determinada região da imagem no domínio da frequência. Este parâmetro é definido como descritor de Fourier.

Estes descritores expressam as distribuições de orientação e frequência espacial na imagem, sendo tipicamente usados para caracterizar textura. Enquanto baixas frequências significam áreas mais uniformes (em relação à distribuição de nível de cinza), altas frequências tipicamente caracterizam bordas ou algum tipo de ruído.

A dificuldade maior para a qualidade deste parâmetro é escolher a melhor ou mais importante região da transformada de Fourier da imagem. Para os devidos testes, foram criadas regiões nos formatos radial e setorial, sendo o número de regiões especificado pelo usuário. Estas regiões podem ser obtidas, respectivamente, através das equações a seguir.

$$S_{rad}[k] = \sum_{\sqrt{u^2+v^2}=r_k}^{\sqrt{u^2+v^2}=r_{k+1}} F_I(u, v) \quad \text{para } \left| \sqrt{u^2 + v^2} \right| < 0,5 \quad (4.11)$$

onde: F_I é a magnitude da Transformada Discreta de Fourier; u e v são as coordenadas da imagem normalizadas entre -0.5 e 0.5 ; e r é o raio definido para uma região k .

$$S_{set}[k] = \sum_{\arctan(v/u)=t_k}^{\arctan(v/u)=t_{k+1}} |F_I(u, v)|^2 \quad \text{para } \sqrt{u^2 + v^2} < 0,5 \quad (4.12)$$

onde t_k e t_{k+1} indicam as inclinações-limite para as regiões.

Exemplo: A Figura 4.5 a seguir mostra um exemplo dessas regiões, 4.5b mostra 4 regiões radiais rotuladas e 4.5c mostra 4 regiões setoriais rotuladas.

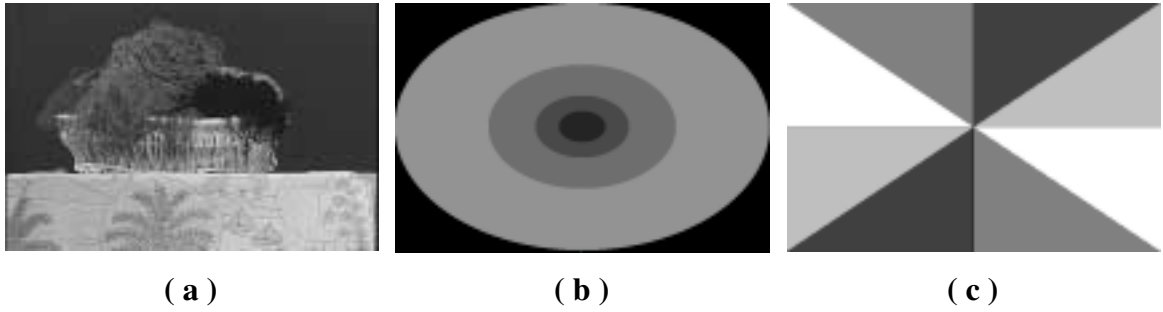


Figura 4.5 – Regiões para análise por descritores de Fourier (a) imagem original
(b) regiões radiais (c) regiões setoriais

Desta forma, podem ser extraídas características como os valores médios dos pixels pertencentes ao espectro de Fourier da imagem original limitado pelas regiões.

A dificuldade maior para a qualidade deste parâmetro é escolher a melhor ou mais importante região da transformada de Fourier da imagem. A partir de testes iniciais, constatou-se que a região radial compreendida entre os períodos 4 e 8 produz os melhores resultados. Tal região (S_{rad}) está definida na equação a seguir.

$$S_{rad} = \begin{cases} F_I(u, v) & \text{se } 0.125 \leq \left| \sqrt{u^2 + v^2} \right| \leq 0.25 \\ 0 & \text{caso contrário} \end{cases} \quad (4.13)$$

sendo F_I a magnitude da Transformada Discreta de Fourier; u e v são as coordenadas da imagem normalizadas entre -0.5 e 0.5 .

A Figura 4.6 ilustra uma aplicação do descritor de Fourier na região radial de interesse.

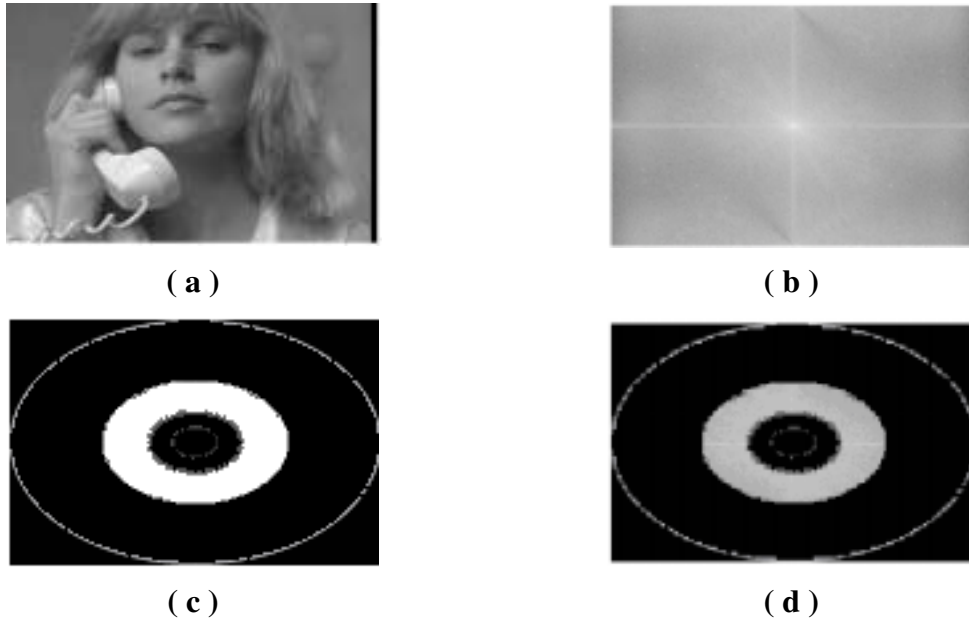


Figura 4.6- (a) Imagem Original (b) Magnitude da Transformada de Fourier
(c) Região radial de interesse (d) Porção do espectro de Fourier pertencente a região

Foi implementada a versão deste descritor para sistemas com “Referência Reduzida”, denominada *vqdefourier.m*, que utiliza apenas um valor como referência. O descritor é calculado a partir da relação entre os somatórios dos espectros radiais da imagem original e da imagem degradada. Tal relação é mostrada na equação a seguir.

$$\text{Descritor}_{\text{radial}} = \frac{\sum_{j=1}^N \sum_{i=1}^M S_{\text{rad_orig}}(i, j)}{\sum_{j=1}^N \sum_{i=1}^M S_{\text{rad_deg}}(i, j)} \quad (4.14)$$

onde: $S_{\text{rad_orig}}$ e $S_{\text{rad_deg}}$ – são as regiões radiais das imagens original e degradada, respectivamente; e M e N são as dimensões da imagem.

4.1.6 Descritor de Sobel

Durante o processo de compressão, o codificador MPEG encontra dificuldade nas componentes de alta frequência de uma imagem. As bordas de uma imagem representam um ponto muito susceptível a degradações, seja o efeito serrilhado [47], tipo ruído ou “blurring”. Um dos métodos mais consagrados para detecção de bordas de uma imagem utiliza o filtro de Sobel [46], representado pelas duas matrizes a seguir.

$$S_x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad S_y = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

O método baseia-se na extração do valor quadrático médio das bordas horizontais e verticais detectadas. A imagem que contém as bordas horizontais (f_x) é gerada pela convolução entre a imagem original e S_x , e a imagem de bordas verticais (f_y) origina-se da convolução entre a imagem original e S_y (equações 4.16).

$$f_{sobel} = \sqrt{f_x^2 + f_y^2} \quad (4.15)$$

$$f_x = conv(f, S_x) \quad \text{e} \quad f_y = conv(f, S_y) \quad (4.16)$$

onde *conv* o operador de convolução.

Na Figura 4.7 a seguir, são mostrados os resultados parciais e final do processo.

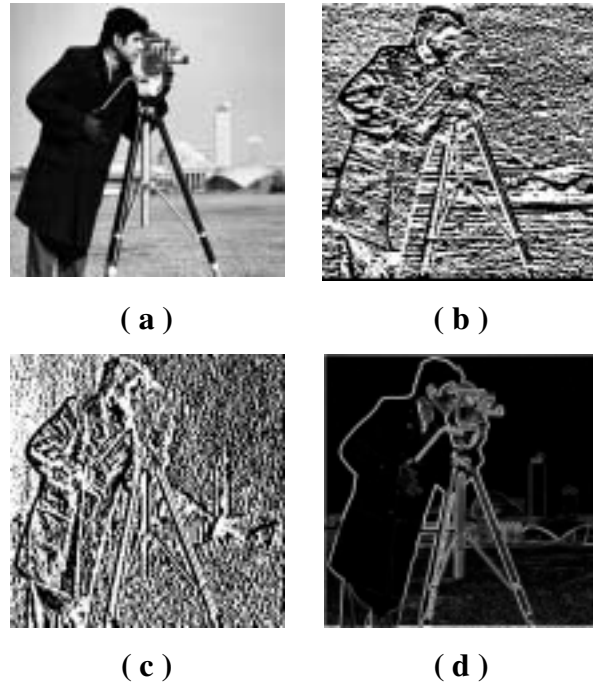


Figura 4.7- Aplicação do Descritor de Sobel (**a**) imagem original (**b**) imagem com bordas horizontais extraídas (**c**) imagem com bordas verticais extraídas (**d**) valor quadrático médio entre as imagens dos Itens b e c

A grande vantagem da convolução com o filtro de Sobel é sua capacidade de suavizar a imagem ao mesmo tempo em que são extraídas altas frequências. Isto permite que ruídos ou porções irrelevantes da imagem com frequência muito alta sejam eliminadas.

Extraídas as bordas das imagens original e degradada sob análise, pode-se realizar uma comparação com referência total através da equação mostrada a seguir.

$$estimator_Sobel = \frac{1}{M \cdot N} \sum_{j=1}^N \sum_{i=1}^M \frac{(f_{sobel_orig}(i, j) - f_{sobel_deg}(i, j))^2}{(f_{sobel_orig}(i, j))^2 + (f_{sobel_deg}(i, j))^2 + 1} \quad (4.17)$$

sendo: f_{sobel_orig} e f_{sobel_deg} – as imagens contendo as bordas extraídas das imagens original e degradada; M e N – as dimensões da imagem.

Para avaliação com referência reduzida, utiliza-se um único valor de cada imagem como referência, extraíndo-se a relação entre os valores médios de intensidade entre f_{sobel_ref} e f_{sobel_deg} , como mostrado a seguir.

$$estimator_Sobel = \frac{\sum_{j=1}^N \sum_{i=1}^M f_{sobel_orig}(i, j)}{\sum_{j=1}^N \sum_{i=1}^M f_{sobel_deg}(i, j)} \quad (4.18)$$

sendo: f_{sobel_orig} e f_{sobel_deg} – as imagens contendo as bordas extraídas das imagens original e degradada; M e N – as dimensões da imagem.

Exemplo: A seguir, são mostradas as bordas extraídas de um par de imagens reais, original e degradada, fornecidas pela Fundação CPqD.

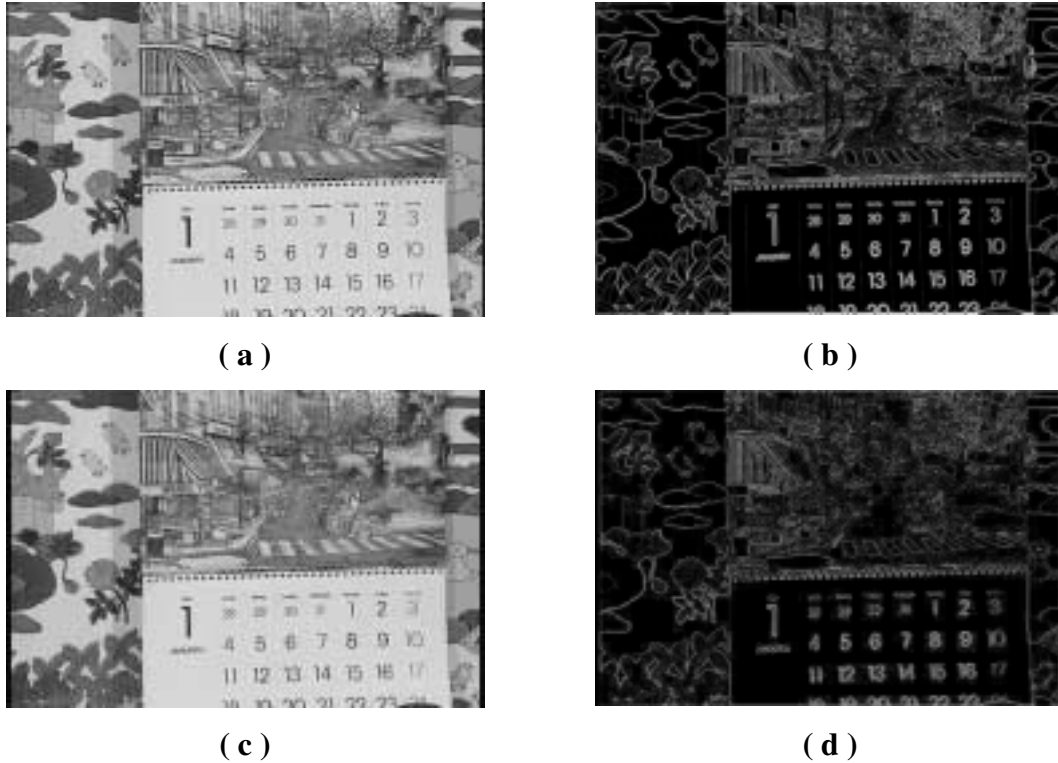


Figura 4.8- Exemplo prático do Descritor de Sobel com imagem real (**a**) imagem original (**b**) bordas extraídas da imagem original (**c**) imagem degradada (**d**) bordas extraídas da imagem degradada

4.1.7 Contraste de nível de cinza e Contraste Colorido

O contraste de nível de cinza é calculado pelas diferenças entre as vizinhanças de um pixel. Assim, dado o componente de luminância de uma imagem fonte $f(x,y)$, uma nova imagem $D_{gray}(x,y)$ é formada como:

$$D_{gray} = \frac{(4 \cdot f(x, y) - (f(x+1, y) + f(x, y+1) + f(x-1, y) + f(x, y-1)))}{4} \quad (4.19)$$

Implementou-se a função *vqcontrastegray.m* para análise com “Referência Total”, onde se aplica a Equação 4.19 nas imagens original e degradada. O descritor é gerado através da equação mostrada a seguir.

$$estimator_gray = \frac{1}{M \cdot N} \sum_{j=1}^N \sum_{i=1}^M \frac{(Dgray_{ref}(i, j) - Dgray_{deg}(i, j))^2}{Dgray_{ref}(i, j)^2 + Dgray_{deg}(i, j)^2} \quad (4.20)$$

sendo: $Dgray_{ref}$ e $Dgray_{deg}$ – os contrastes de nível de cinza das imagens de referência e degradada; M e N – são as dimensões da imagem.

Exemplo: Para uma visualização prática do contraste de nível de cinza, compara-se na Figura 4.9, a seguir, o contraste cinza entre imagens original e sua respectiva degradada.

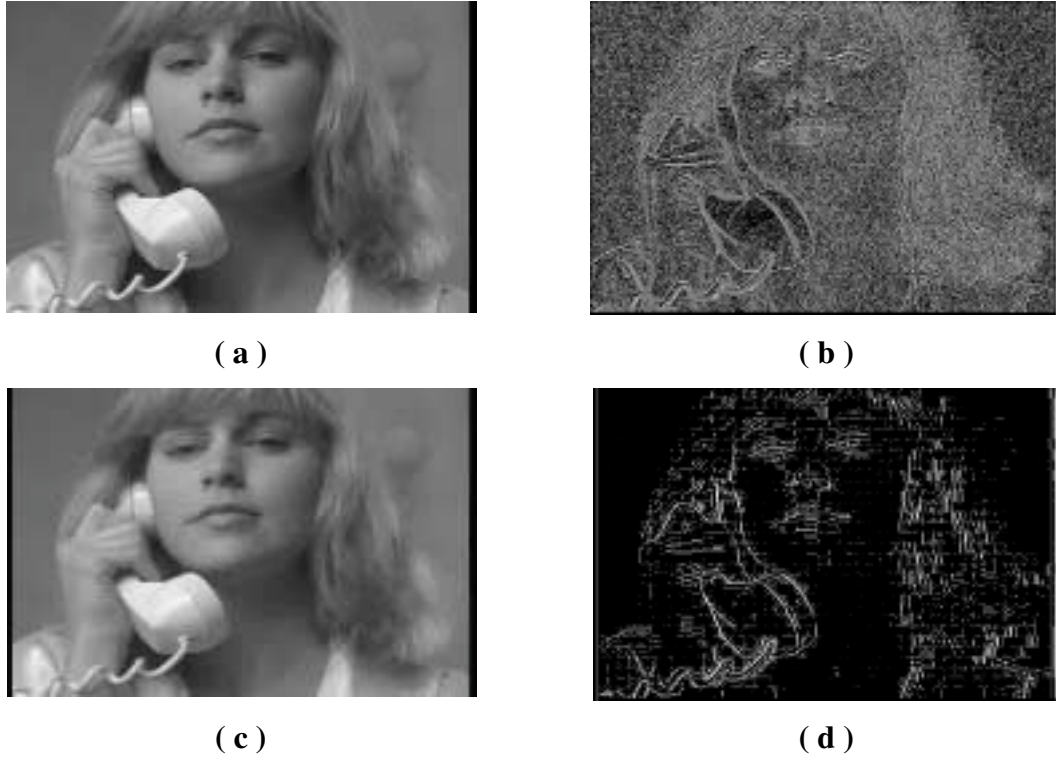


Figura 4.9- Exemplo de contraste de nível de cinza de imagens reais (**a**) imagem original (**b**) contraste de nível de cinza da imagem 2a (**c**) imagem degradada (**d**) contraste de nível de cinza da imagem 2c

Um vetor característico de cor pode ser obtido de forma similar, através do produto escalar dos vetores do espaço RGB. A imagem de entrada, portanto, deve conter os componentes: $f_R(x,y)$ para vermelho, $f_G(x,y)$ para verde e $f_B(x,y)$ para o canal azul, isto é:

$$\vec{v}(x, y) = (f_R(x, y) \quad f_G(x, y) \quad f_B(x, y)) \quad (4.21)$$

O produto escalar médio entre os valores de vizinhança dos pixels é calculado por:

$$D_{cor} = \frac{(\vec{v}(x, y) \cdot \vec{v}(x+1, y) + \vec{v}(x, y) \cdot \vec{v}(x, y+1) + \vec{v}(x, y) \cdot \vec{v}(x-1, y) + \vec{v}(x, y) \cdot \vec{v}(x, y-1))}{4} \quad (4.22)$$

Valores altos para cada produto escalar indicam regiões de cores uniformes, enquanto que valores baixos correspondem a regiões com alto grau de complexidade. Este descritor foi implementado somente para “Referência Total”, através da função denominada *vqcontrastecor.m*. Tal função utiliza o somatório das diferenças quadráticas entre os contrastes coloridos das imagens original e degradada, conforme mostrado a seguir.

$$estimador_cor = \frac{1}{M.N} \sum_{j=1}^N \sum_{i=1}^M \frac{(D_{cor_ref}(i, j) - D_{cor_deg}(i, j))^2}{(D_{cor_ref}(i, j))^2 + (D_{cor_deg}(i, j))^2} \quad (4.23)$$

sendo: D_{cor_ref} e D_{cor_deg} – os contrastes de nível de cinza das imagens de referência e degradada; M e N – são as dimensões da imagem.

Exemplo: Na figura a seguir é apresentado um exemplo prático, utilizando a função *vqcontrastecor.m* implementada.



(a)



(b)



(c)



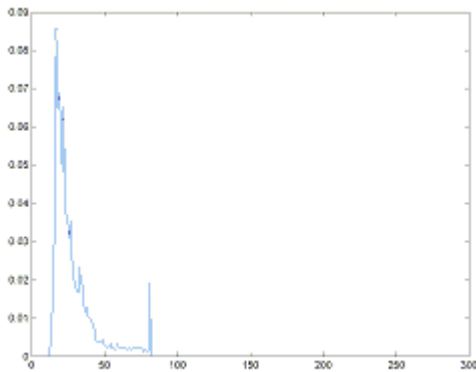
(d)

Figura 4.10- Exemplo de contraste colorido (**a**) imagem original (referência)
(**b**) contraste colorido da imagem 4.10a (**c**) imagem degradada (**d**) contraste colorido
da imagem 4.10c

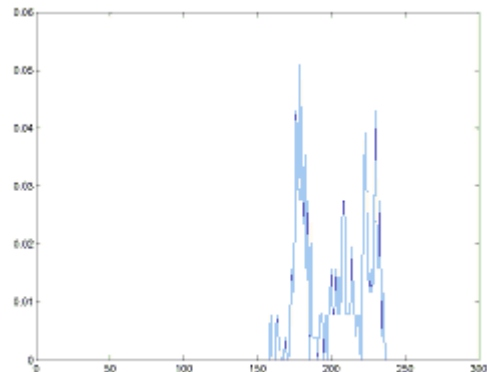
4.1.8 Histograma

O histograma de uma imagem digital com níveis de cinza no intervalo $[0, L-1]$ é uma função discreta $p(r_k) = n_k/n$ [46], em que r_k é o k -ésimo nível de cinza, n_k é o número de pixels na imagem com esse nível de cinza, n é o número total de pixels na imagem e $k=0, 1, 2, \dots, L-1$.

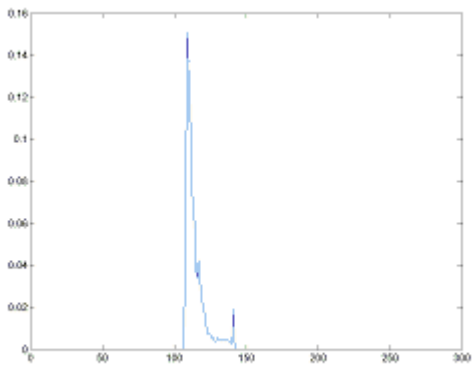
Pode-se definir $p(r_k)$ como uma estimativa da probabilidade de ocorrência do nível de cinza r_k . Um gráfico dessa função fornece, para todos os valores de k , uma descrição global da aparência de uma imagem. A Figura 4.11 mostra os histogramas de quatro tipos básicos de imagens. O histograma mostrado na Figura 4.11a mostra que os níveis de cinza estão concentrados em direção à extremidade escura do intervalo de níveis de cinza. Assim, esse histograma corresponde a uma imagem com características predominantemente escuras. Exatamente o oposto se verifica para a Figura 4.11b.



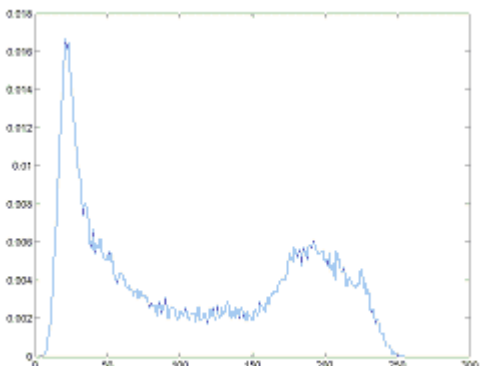
(a)



(b)



(c)



(d)

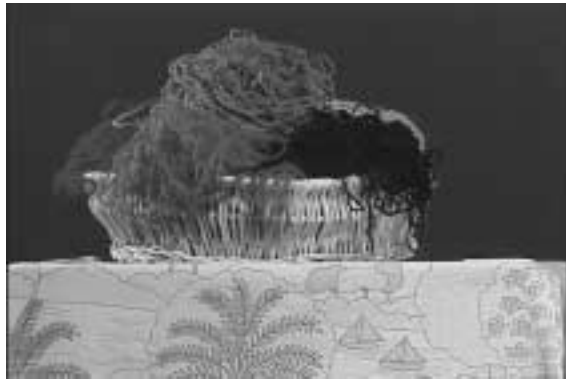
Figura 4.11- Histogramas correspondentes aos quatro tipos básicos de imagens

O histograma mostrado na Figura 4.11c tem uma forma estreita que indica uma escala dinâmica pequena e, portanto, corresponde a uma imagem de baixo contraste. Como todos os níveis de cinza estão no meio da escala, a imagem apareceria com um cinza denso. Finalmente, a Figura 4.11d mostra um histograma com espalhamento significativo, correspondente a uma imagem com alto contraste.

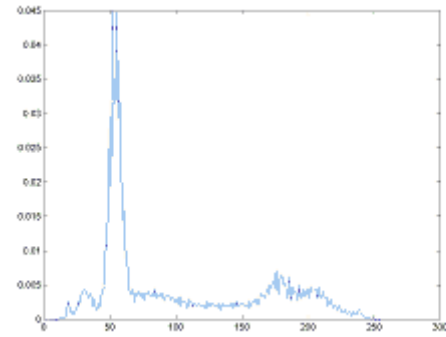
Embora as propriedades anteriormente discutidas sejam descrições globais que não dizem nada específico sobre o conteúdo da imagem, a forma do histograma de uma imagem nos dá informação útil sobre a possibilidade para realce do contraste. Uma das importantes características durante o processo de compressão é procurar diminuir este contraste, tornando a imagem mais simples e permitindo uma maior compressão. Durante o processo de decodificação, a imagem reconstruída pode não manter o nível de contraste de sua original, gerando consideráveis degradações.

O caso extremo dos valores de referência é um vetor com 256 posições, dentro dos limites estabelecidos pelo VQEG. Implementou-se neste trabalho versões de descritores por análise de histograma para avaliação “Em-Laboratório”, função *vqlabhist.m*, e “Em-Serviço”, função *vqservhist.m*. A primeira versão tratou da soma acumulada das diferenças de histograma entre a imagem original e a degradada. Para o segundo caso, utilizou-se a diferença das variâncias dos histogramas das imagens original e degradada, sendo utilizado como referência um único valor de variância.

Exemplo: Utilizando os componentes de luminância de um par de imagens, original e sua respectiva degradada, extrai-se os histogramas de cada uma. Pode-se perceber que o histograma da imagem original possui um nível maior de detalhes, característica marcante da codificação.



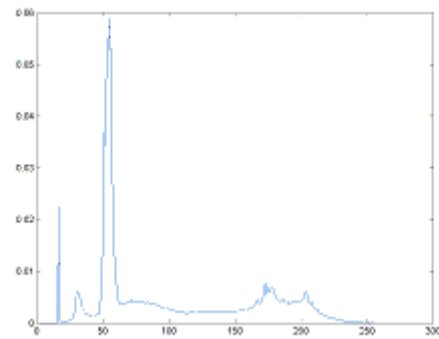
(a)



(b)



(c)



(d)

Figura 4.12- Histograma usado como descritor (a) imagem original (b) histograma da imagem original (c) imagem degradada (d) histograma da imagem degradada

4.1.9 Variância e Vida Média

A variância pode ser considerada o mais simples descritor deste trabalho. Apesar de exigir um tempo de processamento muito pequeno, pode gerar níveis de correlação grandes. Uma das características marcantes durante o processo de codificação e decodificação é a mudança de valor deste parâmetro estatístico. A equação da variância é mostrada a seguir.

$$Var = \sigma^2[f(x, y)] = E[f^2(x, y)] - E^2[f(x, y)] \quad (4.24)$$

onde: E representa a média de um conjunto; e x e y são as coordenadas horizontal e vertical de uma imagem.

Propõe-se neste trabalho várias maneiras de comparar duas imagens através de seus valores de variância. Pode-se extrair a variância das colunas da imagem (função *vqlabvar.m*), e extrair um descritor a partir da equação a seguir.

$$estimator_var = \sum_{i=1}^N \frac{(Var_{ref} - Var_{deg})^2}{Var_{ref}^2 + Var_{deg}^2} \quad (4.25)$$

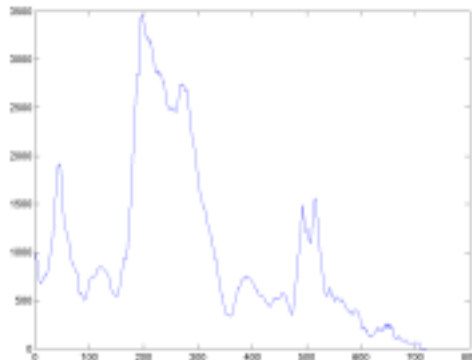
onde: Var_{ref} e Var_{deg} – são as variâncias das colunas das imagens de referência e degradada; N – é o número de colunas da imagem.

Pode-se também extrair o valor único de variância de toda a imagem sob análise (função *vqservvar.m*), e, em seguida, subtrair estes valores para avaliação “Referência Reduzida”. Mas a técnica que se mostra muito robusta, é o cálculo da variância da imagem dividida por blocos de tamanhos iguais (função *vqparbloco.m*). O descritor é gerado também pela Equação 4.25, sendo Var_{ref} e Var_{deg} o vetor com as variâncias de cada bloco. A função implementada *vqparbloco.m* permite também a comparação de desvio padrão, soma e média da intensidade dos pixels pertencentes aos blocos.

Exemplo: A seguir, é apresentada uma análise da variância por colunas de uma imagem real.



(a)



(b)

Figura 4.13- Análise de imagens pela variância das colunas (a) imagem original
(b) gráfico da variância das colunas

Pode-se notar através das Figuras 4.13a e 4.13b, a existência de colunas com pouca variância nas extremidades direita e esquerda da imagem, onde o nível de cinza é praticamente constante.

Outra importante medida que pode ser extraída a partir da variância é a vida média de uma imagem. A vida média é calculada a partir da variância do quadro de vídeo, suavizado sucessivamente através de um filtro Gaussiano com parâmetros σ cada vez maiores. Quando o valor da variância da imagem suavizada for metade da variância da imagem sem suavização, o processo pára. O valor de vida média será o parâmetro *sigma* (σ) que gerou esta condição. A função Matlab *vqhlife.m* (Apêndice I) gera este descritor.

4.1.10 Entropia

A entropia é um parâmetro único vindo da teoria de informação que revela o nível de complexidade da imagem sob análise. Para isso, é considerado que a imagem é formada por um processo estocástico. Este parâmetro foi definido em [51] como mostrado a seguir.

$$H = S[f(a_k)] = -\sum_{k=0}^{K-1} P(a_k) \cdot \log(P(a_k)) \quad (4.26)$$

onde: a_k – representa um nível de cinza; f – é a imagem de entrada; $P(a_k)$ – é a probabilidade de encontrar um nível de cinza a_k na imagem f

As imagens reconstruídas após processos de compressão podem não manter a complexidade de suas regiões, já que imagens menos complexas exigem uma menor quantidade de bits para representação. A comparação entre os valores de entropia das imagens sob análise é realizada a partir da relação quadrática mostrada a seguir.

$$estimator_var = \frac{(H_{ref} - H_{deg})^2}{H_{ref}^2 + H_{deg}^2} \quad (4.27)$$

onde: H_{ref} e H_{deg} – são os valores de entropia das imagens original (referência) e degradada

Exemplo: A partir da função *vqentropia.m* implementada, são apresentados a seguir os valores de entropia aplicados a imagens originais e suas respectivas imagens degradadas.

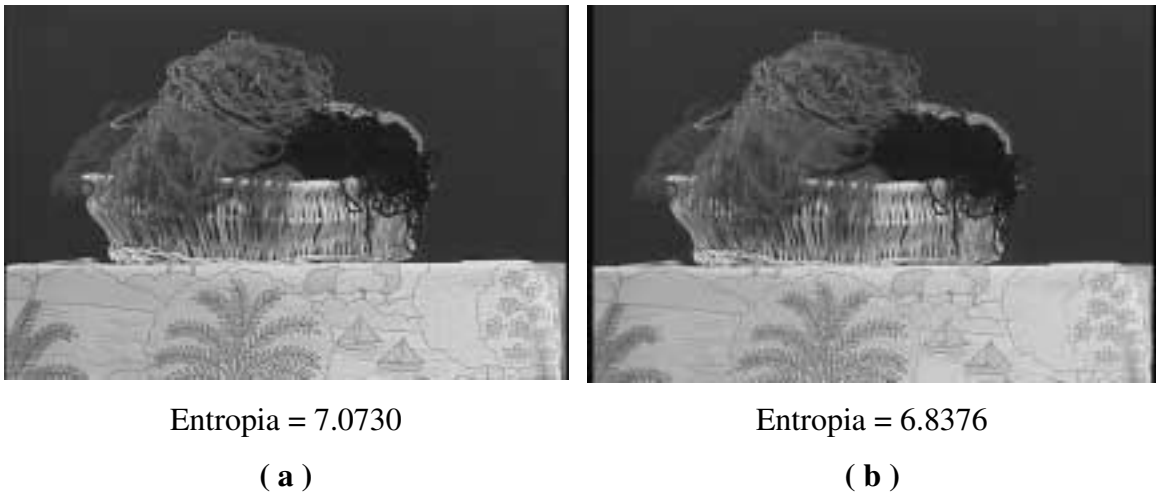


Figura 4.14- Imagens e valores de entropia (imagem *Ball*) (**a**) imagem original
(**b**) imagem degradada



Entropia = 7.0985

(a)

Entropia = 7.0834

(b)

Figura 4.15- Imagens e valores de entropia (imagem *Susy*) (a) imagem original
(b) imagem degradada

4.2 Métodos de comparação

Uma das fases do processo de avaliação de qualidade é obter uma medida que revela a eficácia das funções propostas. Para isso, deve-se comparar os descritores calculados com os valores de uma base subjetiva (considerados ideais). Existem diversas técnicas de comparação entre valores objetivos e subjetivos de qualidade, que analisam de diferentes maneiras e levam em considerações diferentes fatores.

Os dois métodos de comparação mais utilizados pelos grupos participantes do VQEG foram o MSE (Erro Quadrático Médio) e a Correlação de Pearson (r). Desta forma, foram considerados estes dois comparadores neste trabalho.

O MSE e a Correlação de Pearson podem ser calculados a partir das equações seguintes.

$$MSE = \frac{1}{m} \sum_{i=1}^m \sqrt{(x_i - S_i)^2} \quad (4.28)$$

$$r = \frac{\sum_{i=1}^m x_i \cdot S_i - \frac{\sum_{i=1}^m x_i \sum_{i=1}^m S_i}{m}}{\sqrt{\left[\sum_{i=1}^m x_i^2 - \frac{\left(\sum_{i=1}^m x_i \right)^2}{m} \right] \cdot \left[\sum_{i=1}^m S_i^2 - \frac{\left(\sum_{i=1}^m S_i \right)^2}{m} \right]}} \quad (4.29)$$

onde: x_i – os valores objetivos extraídos; S_i – valores da base subjetiva; m – dimensão de x_i .

O valor de MSE pode variar no intervalo $[0, \infty)$ e melhora à medida que se aproxima de zero. O valor de correlação r varia no intervalo $[-1, 1]$, sendo que 1 indica que as bases estão totalmente correlacionadas, zero indica bases totalmente desconrelacionadas e -1 indica que as bases estão total e inversamente correlacionadas.

4.3 Modelo de regressão por busca de projeção

Uma forma conveniente de manipulação de todos os valores dos descritores de qualidade ou vetores de características obtidos é através de uma função interpoladora, buscando os valores da base subjetiva disponível. Nós temos um problema de regressão não-linear multidimensional: para estimar os valores subjetivos através dos descritores de imagem. Devido à alta dimensionalidade do problema, está disponível um número muito pequeno de amostras para desenvolver a função de interpolação. Uma técnica muito apropriada para isso é relacionada ao modelo de Regressão por Busca de Projeção [52,53,54,55,56].

As saídas dos m descritores são os elementos do vetor $x \in \Re^m$. O descritor final é uma soma ponderada dos n mapeamentos não-lineares das n projeções distintas de x , como descrito a seguir.

$$\hat{s} = \sum_{j=1}^n w_j f_j(v_j^T x) \quad (4.30)$$

sendo \hat{s} a superfície final de interpolação, onde é extraído o descritor final de qualidade; x contém os valores dos vetores de características extraídos (variáveis de entrada); w_j são as ponderações da combinação linear das funções não-lineares f_j ($j=1, \dots, n$); v_j são as direções de projeção; e $v_j^T x$ é a projeção de x na direção v_j .

Como pode ser observado na Figura 4.16, os valores objetivos extraídos são as entradas do processo. Cada um dos m valores será conduzido para os n neurônios utilizados, que irão buscar as melhores projeções dos vetores de entradas. A partir de um método de interpolação não-linear, os vetores com as novas projeções irão formar superfícies que, uma vez combinadas, formarão uma superfície próxima à formada com os valores da base subjetiva.

A utilização de modelos na forma da equação (4.30) conduz a processos de aproximação por expansão ortogonal aditiva [57]. Neste modelo, os termos da composição aditiva correspondem a funções escalares de expansão ortogonal a projeções unidirecionais.

A projeção consiste de operações lineares em que um mapeamento de uma determinada dimensão tem suprimidas algumas de suas estruturas de modo a tornar possível sua representação em espaços de menor dimensão. A estrutura projetada pode ser considerada como uma “sombra” da estrutura original, fazendo com que as projeções mais interessantes sejam aquelas que preservam parcelas representativas da estrutura original. A busca destas direções de projeção envolve uma série de manipulações do conjunto de dados de aproximação disponível.

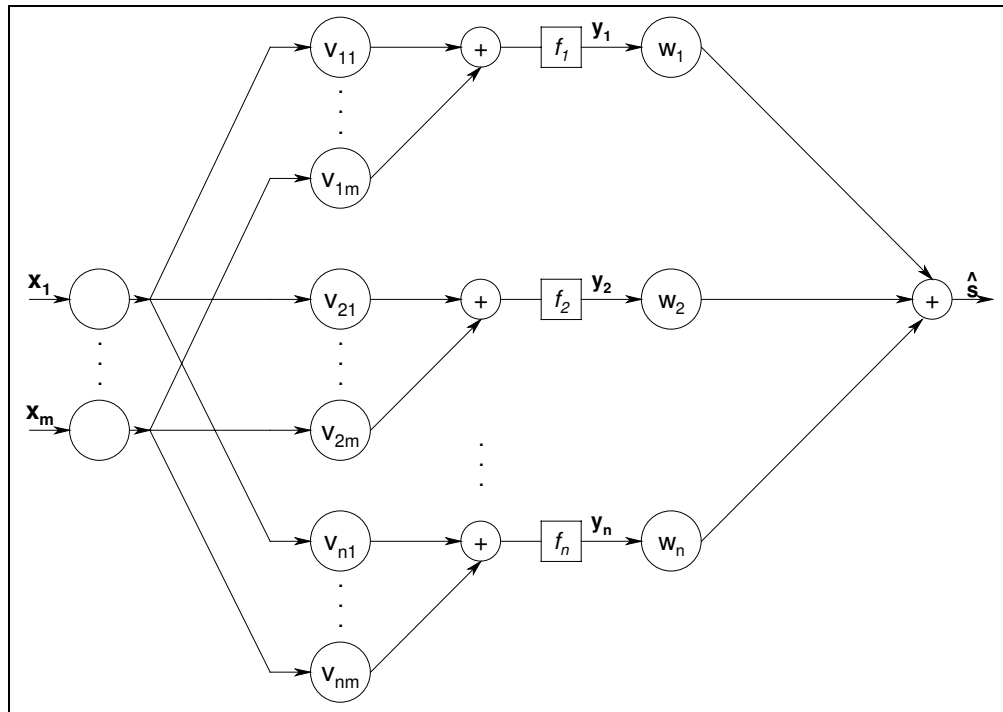


Figura 4.16 – Modelo de regressão por busca de projeção

Baseadas nos dados de entrada-saída, as direções de projeção devem enfatizar as relações, possivelmente não-lineares, existentes entre as variáveis do problema de aproximação. A questão que surge é como obter de forma automática estas direções de projeção. Uma alternativa foi apresentada em [58], em que a direção de projeção corresponde à solução que maximiza um determinado índice numérico de projeção. A utilização de um índice de desempenho na determinação da direção de projeção, ao invés de determiná-la arbitrariamente, permite assegurar apenas um aumento da probabilidade de se encontrar direções de projeção interessantes. Em boa parte dos problemas de aproximação, este aumento da probabilidade é bastante significativo [59].

Uma vez definida a direção de projeção, funções monovariáveis são então determinadas de tal forma que sua expansão em direções ortogonais à direção de projeção forneça a melhor aproximação possível com base nos dados disponíveis. A interdependência entre as funções de expansão ortogonal e a correspondente direção de projeção acaba conduzindo a processos iterativos de aproximação.

Dentre as vantagens de se utilizar projeções unidirecionais têm-se a manutenção de uma maior simplicidade do processo de aproximação e a possibilidade de visualizar graficamente o comportamento da função de aproximação na direção de projeção. Além disso, é possível explicitar o tipo de associação não-linear existente entre as variáveis e uma série de outras informações que não estão diretamente disponíveis considerando-se a dimensão completa do espaço de aproximação.

No entanto, a interpretação dos dados projetados geralmente não se apresenta como uma tarefa simples. Estruturas interessantes presentes na projeção dos dados não necessariamente correspondem à projeção de estruturas interessantes, da mesma forma que estruturas interessantes presentes nos dados podem conduzir a nenhuma projeção interessante [60].

Após a definição de uma direção de projeção e da correspondente função de expansão ortogonal, uma transformação deve ser aplicada ao conjunto de dados para que a informação já representada seja removida, permitindo o reinício do processo a partir de uma nova direção de projeção. Logo, a busca seqüencial de direções de projeção pode ser implementada através dos passos descritos a seguir.

- 1- Encontra-se uma direção de projeção *ótima*;
- 2- Remove-se do conjunto de dados a estrutura resultante da projeção dos dados nesta direção;
- 3- Reinicia-se o processo até que nenhuma outra projeção revele qualquer estrutura, ou seja, até que o modelo de aproximação concorde com os dados amostrados em todas as projeções.

Este procedimento iterativo e construtivo, em que cada novo sub-problema de aproximação deve apresentar apenas informações não representadas pelos sub-problemas de aproximação anteriores, produz funções de aproximação multivariáveis utilizando composição aditiva de funções monovariáveis, na forma da equação (4.30).

Como as funções $f_j(.)$ ($j=1,...,n$) são constantes para valores de x em hiperplanos do \mathcal{R}^m , elas são denominadas funções de expansão ortogonal a uma determinada direção. O problema de aproximação por expansão ortogonal é descrito em [57,59].

Baseado na Figura 4.16 é apresentado a seguir o algoritmo de aprendizado construtivo para uma única saída utilizado no trabalho:

- 1- Dados X (vetores de características extraídos) $\in \Re^{m \times N}$ e S (dados subjetivos) $\in \Re^N$, tome $j=0$, $D=S$;
- 2 - Utilizando X e D , faça $j=j+1$ e atribua:
 - um valor inicial para $v_j \in \Re^m$;
 - uma forma inicial para f_j (sendo que f_j pode ser uma função paramétrica [61] ou não-paramétrica [52]);
 - um valor inicial para $w_j \in \Re$, obtido pelo método dos quadrados mínimos;
- 3 – Utilizando X e D , resolva os seguintes problemas em seqüência até convergência (medida por algum critério de parada):
 - 3.1 - fixe f_j e obtenha um valor ótimo para v_j ;
 - 3.2 - fixe v_j , obtenha um f_j ótimo via técnicas de regularização e retorne ao passo 3.1.
- 4 - Obtenha um valor ótimo para w_j pelo método dos quadrados mínimos;
- 5 – Para cada b tal que $1 \leq b \leq j$, calcule

$$D = S - \sum_{\substack{k=1 \\ k \neq b}}^j w_k f_k(v_k^T X) \quad (4.31)$$

e repita os passos 3 e 4, com $j=b$;

- 6- Por avaliação da participação de cada neurônio $w_k f_k(v_k^T X)$, $k=1, \dots, j$, na representação da matriz S , aplique um procedimento de poda de neurônios que não apresentem um nível de participação mínima;
- 7- Enquanto um determinado nível de aproximação não for atingido (medido por algum critério de parada, por exemplo, utilizando validação cruzada), retorne ao passo 2.

Capítulo 5

Experimentos

Este capítulo descreve o material de vídeo utilizado, os passos seguidos para a realização dos experimentos e os resultados alcançados. A escolha das cenas envolvidas nos experimentos é de fundamental importância, já que é preciso selecionar uma quantidade limitada de tipos de cenas que englobam a maior parte das características presentes em imagens naturais e artificiais.

No desenvolvimento dos experimentos, procurou-se eliminar as medidas dos descritores mais afastados dos valores subjetivos, e, além disso, combinou-se as melhores medidas visando resultados mais consistentes.

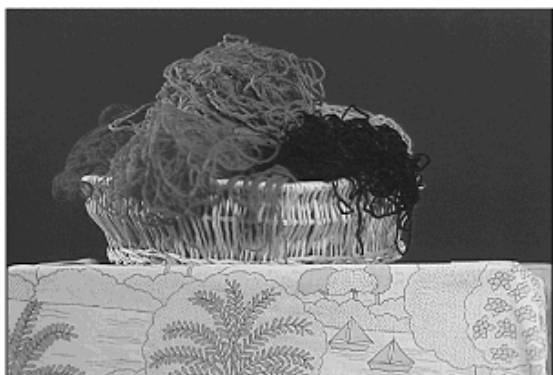
5.1 Base de Dados

Uma das importantes fases, em qualquer experimento envolvendo avaliação de qualidade de vídeo objetiva ou subjetiva, é a escolha adequada da base de dados ou seqüências de vídeo a serem analisadas. Esta é uma fase que pode gerar muitas discussões, a partir do momento em que é preciso escolher um limitado número de seqüências, que possuam peculiaridades ou detalhes bem diferenciados.

Neste trabalho, um conjunto contendo 12 seqüências de vídeo, degradadas por 13 processos diferentes, foi disponibilizado pela Fundação CPqD para realização dos experimentos. Nestes 13 processos de degradação, foram realizadas cascatas de *codecs*, processos de suavização, inserção de ruído e de sinal interferente analógico, dentre outros, de forma a emular os fenômenos mais comuns de perda de qualidade. A Tabela 5.1 mostra os nomes das seqüências.

Tabela 5.1 – Cenas de vídeo selecionadas

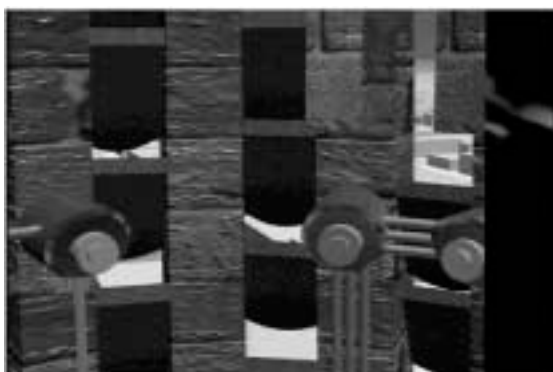
Nº da cena	Nome da Cena
<u>1</u>	Flower Garden
<u>2</u>	Mobile & Calendar
<u>3</u>	Kielharbor
<u>4</u>	Tree
<u>5</u>	Diva
<u>6</u>	Ball
<u>7</u>	Table Tennis
<u>8</u>	Cacau
<u>9</u>	Susie
<u>10</u>	Moto
<u>11</u>	Girassol
<u>12</u>	Catedral



(a)



(b)



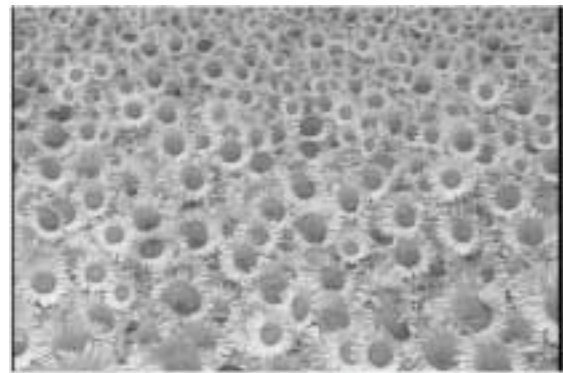
(c)



(d)



(e)



(f)

Figura 5.1 – Quadros de vídeo das seqüências trabalhadas (a) Ball (b) Cacau
(c) Catedral (d) Diva (e) Flower (f) Girassol



(g)



(h)



(i)



(j)



(k)



(l)

Figura 5.1 – Quadros de vídeo das seqüências trabalhadas (g) Kiel (h) Móbile&Calendar
(i) Moto (j) Susie (k) Table Tennis (l) Tree

Estas seqüências envolvem variados tipos de cenas naturais e artificiais, como pode ser visto nos quadros de vídeo mostrados na Figura 5.2. Apesar disso, representam um número reduzido de amostras, considerando a enorme dimensionalidade do problema.

Nos testes realizados neste trabalho, como já comentado, foram utilizados apenas os primeiros quadros de cada seqüência, por já conterem as informações espaciais necessárias aos algoritmos propostos. Desta forma, as informações temporais não são consideradas, impedindo a detecção de degradações deste tipo.

Uma base de dados subjetiva, também disponibilizada pela Fundação CPqD, fornece as notas dos testes com seres humanos e é utilizada como os valores ideais a serem alcançados pelos descritores. Esta base subjetiva foi gerada através de testes com especialistas e usuários comuns, obedecendo a rígidas normas como as especificadas na ITU-R BT.500-7 [21].

5.2 Descrição do Experimento

O experimento realizado envolveu duas fases, buscando a seleção dos melhores descritores de qualidade para avaliação em ambos os sistemas de referência: total e reduzida, e a combinação destes melhores descritores para obtenção de um descritor final.

Primeiramente, foram processados todos os quadros de vídeo, calculando-se todos os descritores propostos no Capítulo 4. Após a obtenção dos vetores de características, sejam eles de referência total ou referência reduzida, realizou-se uma comparação inicial com a base de dados subjetiva. Desta forma, puderam ser descartadas as medidas que não correlacionavam ou que se distanciavam demasiadamente das notas subjetivas. Estes descritores, portanto, não fizeram parte do conjunto de medidas ponderadas pelo processo de otimização.

Os seguintes descritores para referência reduzida foram descartados: descritor de entropia, descritor de vida média, variância de histograma, variância de colunas, somatório e média por blocos, variância de contraste colorido e de nível de cinza. Apesar da

reconhecida aplicabilidade da medida de entropia na determinação do nível de complexidade de uma imagem, este descritor não gerou bons níveis de correlação. Como a imagem de teste original teoricamente possui um nível de complexidade maior que sua respectiva degradada, esperava-se um melhor resultado deste descritor.

Desta forma, foram selecionadas sete medidas de descritores para sistemas de avaliação com referência total e sete medidas para sistemas com referência reduzida, formando duas bases de dados contendo os respectivos vetores de atributos. Tais descritores estão especificados na Tabela 5.2 a seguir, juntamente com cada um dos níveis individuais de correlação.

Tabela 5.2- Descritores selecionados em ordem decrescente de correlação

Base 1 (Referência Total)	Correlação de Pearson (FR)	Base 2 (Referência Reduzida)	Correlação de Pearson (RR)
1- PSNR	0.74	1- Dimensão Fractal	0.76
2- Regiões Planas	0.69	2- Descritor de Blocagem	0.74
3- Contraste Colorido	0.67	3- Desvio Padrão por Bloco	0.69
4- Descritor Sobel	0.64	4- Descritor de Fourier	0.67
5- Variância por Colunas	0.55	5- Variância por Bloco	0.59
6- Descritor de Histograma	0.51	6- Descritor Sobel Em-Serviço	0.57
7- Contraste Nível de Cinza	0.49	7- Regiões Planas Em-Serviço	0.52

Na segunda fase, os vetores de atributos de ambos os sistemas de referência, juntamente com a base subjetiva, foram submetidos ao processo de treinamento do modelo especificado no Item 4.3. Para isso, foram usados dois neurônios e funções de expansão ortogonal Hermitianas. Aplicando-se assim o método de Regressão por Busca de Projeção [54], foram procuradas as melhores projeções dos vetores de atributos combinados, para se chegar a valores próximos da base de dados subjetiva.

Num primeiro momento, todas as imagens disponíveis foram utilizadas no treinamento, para se verificar o erro e a correlação globais, utilizando 2, 3, 5 e 7 descritores. A escolha dos descritores foi feita utilizando sempre aqueles com maior correlação inicial (sem se combinar com outros descritores). Assim, no caso de dois descritores para “Referência Total”, foram utilizados o PSNR e o Descritor de Regiões Planas. Para três descritores, foram utilizados o PSNR, o Descritor de Regiões Planas e o Contraste Colorido, e assim, sucessivamente. Um ponto importante a ser observado nos Experimentos é se a combinação dos descritores melhora ou não os níveis de correlação dos descritores isolados.

Em seguida, realizou-se o treinamento a partir de 11 das 12 imagens, e foram testadas as projeções e ponderações de saída do processo de otimização na imagem restante. Este procedimento pode ser realizado já que o método de Regressão por Busca de Projeção não exige um número elevado de amostras. Além disso, a imagem restante representaria uma imagem qualquer de teste em um sistema de avaliação objetiva.

O processo, mostrado na Figura 5.2, foi assim realizado para cada uma das imagens utilizando novamente 2, 3, 5 e 7 descritores.

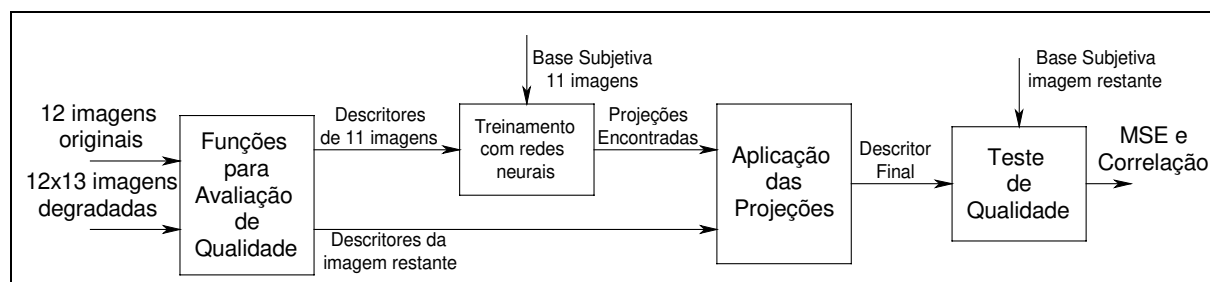


Figura 5.2 – Diagrama de blocos do Experimento

Finalmente, foi verificado se as ponderações obtidas juntamente com as superfícies de interpolação geradas no processo de treinamento foram eficientes. Para isso, foram utilizados novamente os métodos de comparação especificados no Item 4.2. Valores de correlação acima de 0.8 são considerados bons, baseando-se nos resultados encontrados pelos grupos VQEG na primeira fase [12].

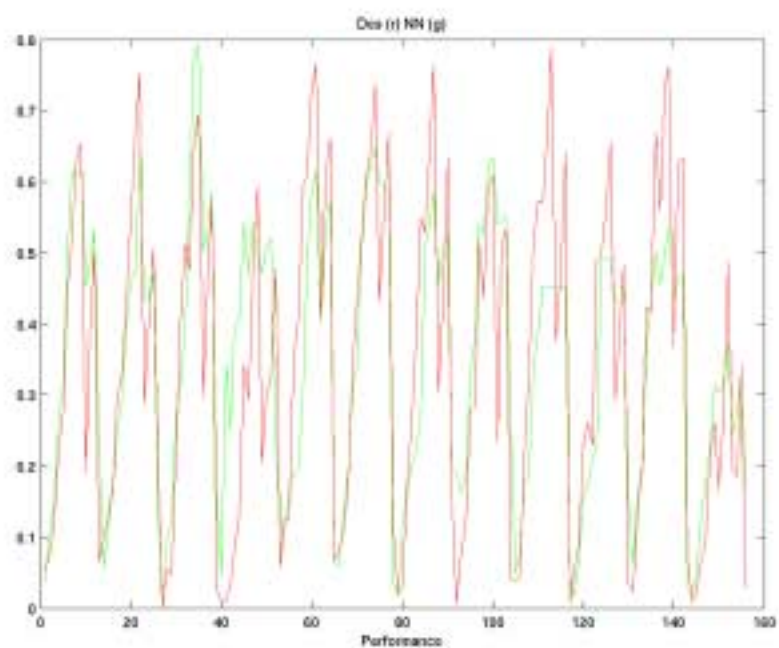
5.3 Resultados

A Tabela 5.3 mostra os valores de Erro Quadrático Médio (MSE) e de Correlação de Pearson (r) para a primeira fase de testes, onde todos as seqüências de vídeo foram utilizadas no treinamento.

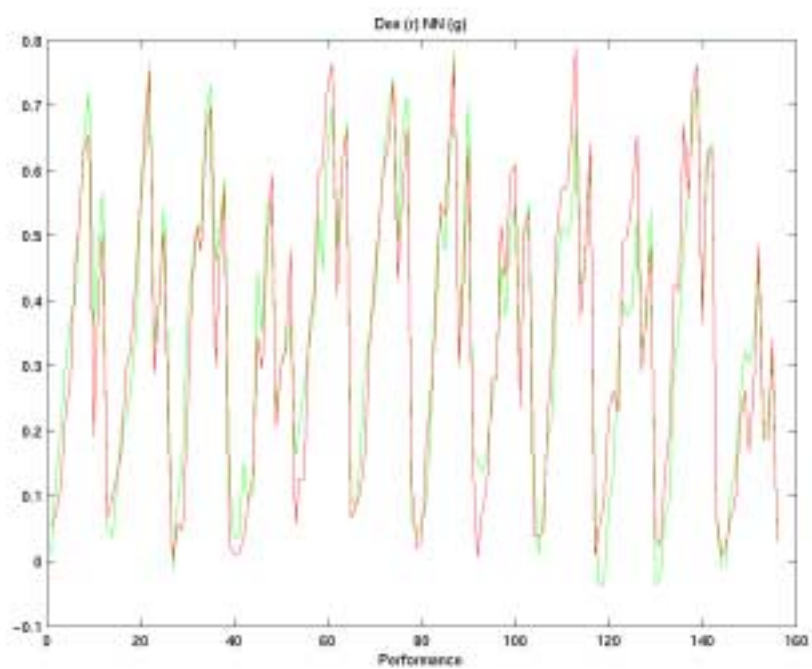
Tabela 5.3 – Resultados utilizando as 12 imagens

Referência	MSE / Correlação de Pearson			
	2 desc.	3 desc.	5 desc.	7 desc.
Total	0.12 / 0.83	0.12 / 0.84	0.09 / 0.91	0.07 / 0.95
Reduzida	0.10 / 0.88	0.09 / 0.89	0.09 / 0.91	0.08 / 0.92

Pode-se notar, a partir dos dados contidos na Tabela 5.3, que tanto o *erro* quanto a *correlação* melhoram com o aumento do número de descritores. Para todos os casos, o erro foi pequeno e os níveis de correlação próximos de 1.



(a)



(b)

Figura 5.3 – Curvas com descritores finais calculados (verde) e dados subjetivos (vermelho) para referência total usando (a) 2 (b) 7 descritores

A Figura 5.3 mostra as curvas que comparam os descritores finais e os valores da base de dados subjetiva, para referência total. São mostrados os casos de 2 e 7 descritores. Pode-se perceber que as curvas com as medidas objetivas acompanham muito bem os valores subjetivos. Os 12 “picos” que aparecem nas curvas são respectivos a cada uma das 12 imagens trabalhadas, e cada uma dessas regiões possui 13 pontos, respectivos às condições de degradação.

Um resultado importante que pode ser observado é o fato que os níveis de *erro* e *correlação*, utilizando as 12 imagens no treinamento, foram melhores para o caso de sistema com referência reduzida. Contrariando o que se esperava teoricamente, os descritores para referência reduzida, a priori, tiveram desempenho melhor.

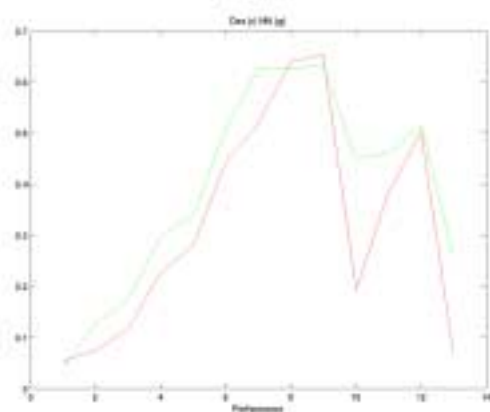
Os valores altos de correlação, observados neste caso, não demonstram a qualidade dos descritores, visto que foram utilizadas todas as imagens no treinamento e já eram conhecidas as medidas subjetivas. Porém esta fase é importante para comprovar a eficácia das técnicas de treinamento e para observar o comportamento dos diferentes tipos de imagens no processo. Para comprovar a eficácia dos descritores de qualidade extraídos, realizou-se o treinamento com 11 das 12 imagens disponíveis, aplicando-se as projeções ou ponderações calculadas na imagem restante. A Tabela 5.4 mostra os parâmetros de comparação para o conjunto de medidas de qualidade com referência total.

Pode-se notar, pela Tabela 5.4, que o melhor caso refere-se ao uso de 5 descritores para a avaliação, onde o nível médio de correlação global atingiu 0.86. Os valores de MSE para este caso são bons, exceto para a imagem 7 (*Table Tennis*), onde o processo divergiu. Assim, quando todos os parâmetros são combinados, o treinamento envolvendo o PSNR, o descritor de Regiões Planas, Contraste Colorido, Descritor de Sobel e Variância de Colunas gerou os melhores resultados. Além disso, percebe-se a necessidade de se utilizar determinados tipos de imagens, definidas como “imagens chave”, fundamentais no processo de treinamento. Isto pode ser notado, por exemplo, nos baixos níveis de correlação encontrados na imagem 4 (*Tree*). Isto se deve por tal imagem conter peculiaridades não presentes nas outras cenas, importantes para o treinamento de descritores capazes de avaliar todo tipo de imagem digital.

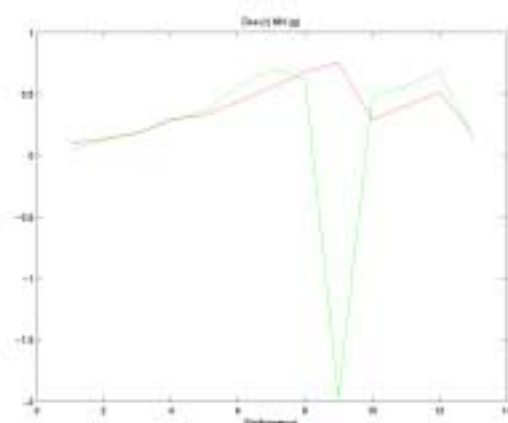
Tabela 5.4 – Resultados dos testes para referência total

No da imagem de teste	MSE / Correlação de Pearson			
	2 desc.	3 desc.	5 desc.	7 desc.
<u>1</u>	0.10 / 0.92	0.10 / 0.93	0.11 / 0.87	0.40 / 0.66
<u>2</u>	0.09 / 0.91	0.76 / -0.27	0.08 / 0.97	0.17 / 0.64
<u>3</u>	0.30 / 0.88	0.96 / 0.81	0.11 / 0.94	0.13 / 0.92
<u>4</u>	0.34 / -0.26	0.34 / -0.28	0.31 / 0.67	0.13 / 0.67
<u>5</u>	0.19 / 0.91	0.13 / 0.88	0.14 / 0.88	0.13 / 0.90
<u>6</u>	0.15 / 0.85	0.12 / 0.84	0.19 / 0.97	0.11 / 0.95
<u>7</u>	0.15 / 0.83	0.18 / 0.59	6.92 / 0.58	0.18 / 0.81
<u>8</u>	0.13 / 0.91	0.21 / 0.90	0.09 / 0.89	0.08 / 0.96
<u>9</u>	0.22 / 0.92	0.17 / 0.72	0.15 / 0.94	0.18 / 0.87
<u>10</u>	0.15 / 0.78	0.19 / 0.87	0.12 / 0.87	0.15 / 0.90
<u>11</u>	0.16 / 0.91	0.17 / 0.72	0.25 / 0.98	0.27 / 0.94
<u>12</u>	0.10 / 0.84	0.14 / 0.86	0.12 / 0.85	0.40 / 0.36
Pior Caso	0.34 / -0.26	0.34 / -0.28	6.92 / 0.58	0.40 / 0.36
Melhor Caso	0.10 / 0.92	0.10 / 0.93	0.25 / 0.98	0.08 / 0.96
Valor Médio	0.17 / 0.78	0.29 / 0.64	0.71 / 0.86	0.19 / 0.79
Valor Médio (excluindo pior caso)	0.15 / 0.87	0.23 / 0.73	0.15 / 0.89	0.17 / 0.83
Valor Médio (excluindo 2 piores casos)	0.14 / 0.88	0.17 / 0.83	0.13 / 0.91	0.15 / 0.85

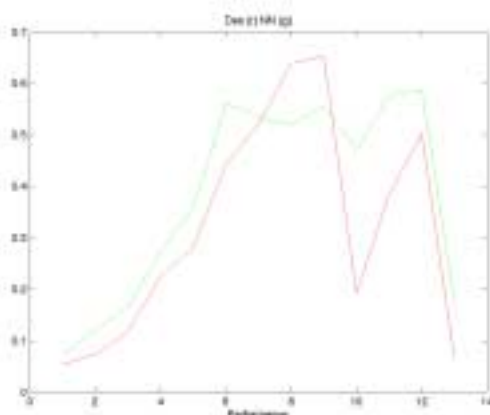
São mostradas a seguir, algumas das importantes curvas comparativas geradas nesta fase (referência total).



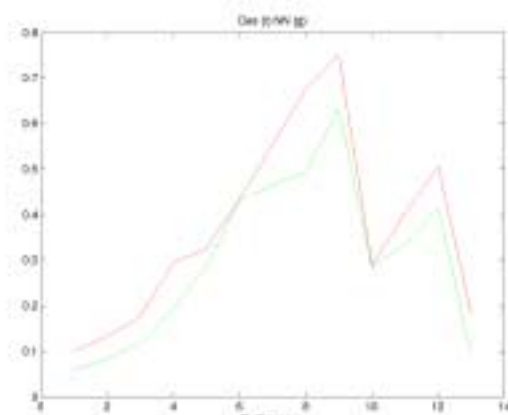
(a)



(b)



(c)



(d)

Figura 5.4 - Curvas com descritores calculados (verde) e dados subjetivos (vermelho) para referência total usando (**a**) 3 descritores e imagem 1 (*Garden*) (**b**) 3 descritores e imagem 2 (*Mobile&Calendar*) (**c**) 5 descritores e imagem 1 (*Garden*) (**d**) 5 descritores e imagem 2 (*Mobile&Calendar*)

As duas curvas escolhidas representam bem os resultados alcançados para referência total, referindo às imagens *Garden* e *Mobile&Calendar*, esta última considerada problemática. No caso de 3 descritores, Figuras 5.4a e 5.4b, a curva em verde, referente aos descritores extraídos, acompanhou bem a curva vermelha, dos dados subjetivos, para a imagem 1 (*Garden*). No caso da imagem 2, a curva em verde divergiu para uma das condições de degradação, diminuindo assim o nível de correlação global e, conseqüentemente, diminuindo a qualidade dos três descritores combinados. No caso de 5 descritores, Figuras 5.3c e 5.3d, isto não ocorre, sendo que a curva em verde está sempre próxima da curva com as medidas subjetivas, independente da imagem e condição de degradação analisada. Neste caso, constata-se que a influência dos descritores Sobel e Variância por Colunas melhoram a qualidade do descritor final de qualidade encontrado. Por outro lado, quando utilizados o Descritor de Histograma e de Contraste Nível de Cinza, o nível médio de correlação cai, que pode ser resultante de duas condições presentes. Uma primeira, onde estes descritores não se combinam bem com os outros, extraindo características já consideradas nos algoritmos dos outros descritores. E, uma segunda, onde a complexidade do processo de treinamento aumenta, usando agora 7 descritores, dificultando encontrar uma solução ótima.

Pode-se afirmar, portanto, que o conhecido descritor que relaciona o Pico da Relação Sinal-Ruído (PSNR), pode ter seu nível de correlação com a base subjetiva melhorado se combinado com outros descritores. Um nível de correlação médio de 0.86 representa um bom valor, considerando que os grupos participantes da primeira fase do VQEG [1,12] obtiveram valores próximos a este, nos testes para baixas taxas. Porém, os testes do VQEG envolveram um número maior de seqüências, com várias bases de dados subjetivas, inviabilizando assim maiores comparações com os resultados encontrados neste trabalho.

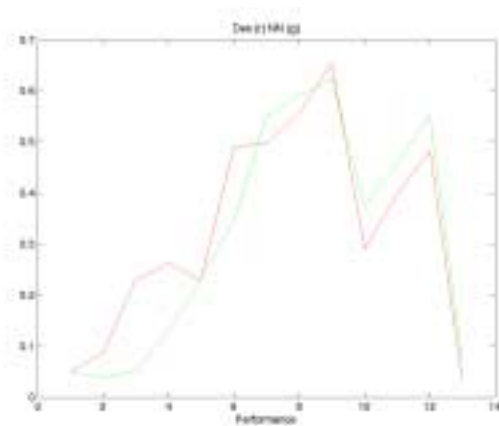
A Tabela 5.5 a seguir mostra os parâmetros de comparação para a *base2*, referente a sistemas de avaliação com referência reduzida.

Tabela 5.5 – Resultados dos testes para referência reduzida

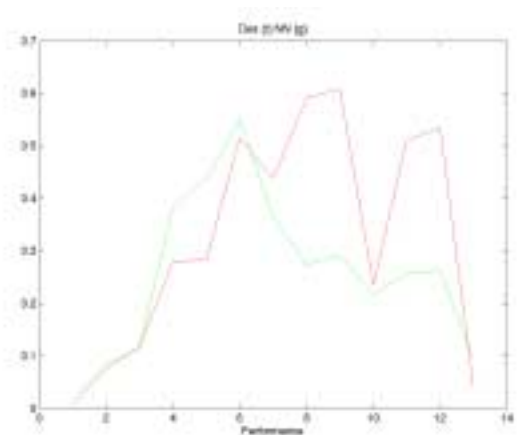
No da imagem de teste	MSE / Correlação de Pearson			
	2 desc.	3 desc.	5 desc.	7 desc.
<u>1</u>	0.06 / 0.94	0.10 / 0.94	0.06 / 0.94	0.06 / 0.95
<u>2</u>	0.16 / 0.85	0.13 / 0.87	0.14 / 0.84	0.20 / 0.80
<u>3</u>	0.08 / 0.93	0.08 / 0.94	0.09 / 0.93	0.08 / 0.94
<u>4</u>	0.15 / 0.78	0.17 / 0.83	0.21 / 0.79	0.33 / 0.77
<u>5</u>	0.14 / 0.90	0.19 / 0.90	0.18 / 0.87	0.27 / 0.92
<u>6</u>	0.05 / 0.97	0.06 / 0.96	0.09 / 0.96	0.09 / 0.96
<u>7</u>	0.23 / 0.65	0.16 / 0.82	0.38 / 0.52	0.22 / 0.77
<u>8</u>	0.04 / 0.97	0.04 / 0.98	0.08 / 0.96	0.17 / 0.66
<u>9</u>	0.21 / 0.64	0.57 / 0.52	0.28 / 0.41	0.13 / 0.93
<u>10</u>	0.08 / 0.92	0.07 / 0.94	0.06 / 0.95	0.08 / 0.92
<u>11</u>	0.14 / 0.97	0.10 / 0.98	0.20 / 0.88	1.03 / 0.58
<u>12</u>	0.13 / 0.92	0.13 / 0.89	0.27 / 0.85	0.29 / 0.78
Pior Caso	0.21 / 0.64	0.57 / 0.52	0.28 / 0.41	1.03 / 0.58
Melhor Caso	0.04 / 0.97	0.10 / 0.98	0.08 / 0.96	0.09 / 0.96
Valor Médio	0.12 / 0.87	0.15 / 0.79	0.17 / 0.82	0.24 / 0.83
Valor Médio (excluindo pior caso)	0.11 / 0.89	0.11 / 0.91	0.15 / 0.86	0.17 / 0.85
Valor Médio (excluindo 2 piores casos)	0.10 / 0.91	0.10 / 0.92	0.13 / 0.89	0.15 / 0.87

Pode-se observar, a partir da Tabela 5.5, que os níveis de correlação obtidos para as medidas com referência reduzida foram altos. Utilizando a Dimensão Fractal e o Descritor de Blocagem no processo de treinamento, conseguiu-se as medidas mais robustas deste trabalho. Os níveis de correlação foram altos até mesmo para imagens consideradas críticas, como a *Tree* e a *Mobile&Calendar*.

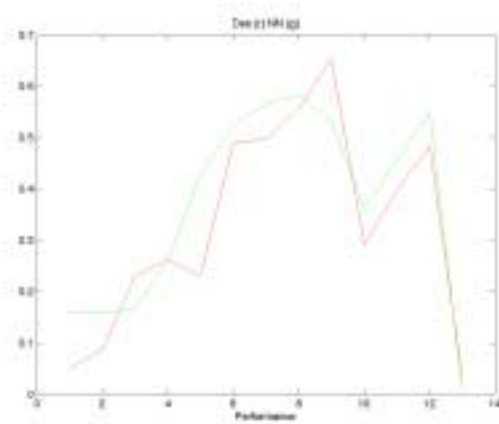
São mostradas, a seguir, algumas das curvas comparativas geradas nesta fase.



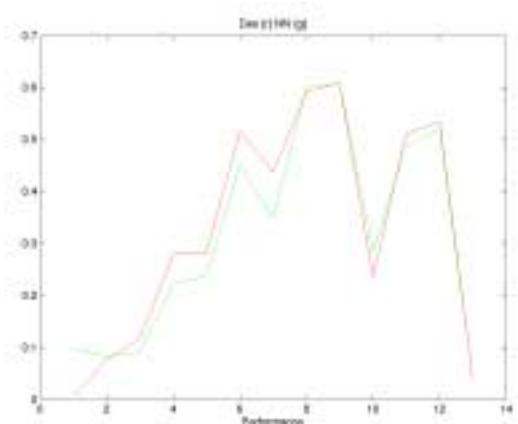
(a)



(b)



(c)



(d)

Figura 5.5 - Curvas com descritores extraídos (verde) e dados subjetivos (vermelho) para referência reduzida usando (a) 7 descritores e imagem 10 (*Moto*) (b) 7 descritores e imagem 8 (*Cacau*) (c) 2 descritores e imagem 10 (d) 2 descritores e imagem 8

Nota-se, através das curvas da Figura 5.5, que uma quantidade elevada de descritores não implica em melhores resultados. Utilizando as imagens 8 (Cacau) e 10 (*Moto*) nos testes, encontra-se níveis de correlação maiores para o caso de dois descritores combinados. Isto ocorre porque a combinação de um número grande de vetores de características no processo de treinamento dificulta a convergência para uma solução ótima, comprometendo os resultados finais.

Os resultados comprovam a eficácia de duas técnicas inéditas, que podem ser consideradas complementares diante da alta qualidade encontrada para um sistema com referência reduzida. Por um lado, calcula-se um parâmetro de variação de volume de imagens dilatadas com elementos estruturantes cada vez maiores, e por outro, analisa-se a blocagem separadamente, detectando as estruturas quadriculadas presentes, e, finalmente, combina-se estes resultados para encontrar um descritor final de alta qualidade.

O fato dos descritores para referência reduzida gerarem melhores resultados é proveniente da maior ênfase dada durante o trabalho para este tipo de análise. Os descritores para referência reduzida poderiam ter sido combinados com os de referência total, para encontrar, possivelmente, descritores de maior qualidade. Esta aplicação pode ser promissora em trabalhos futuros.

Capítulo 6

Conclusões

Este trabalho apresentou uma metodologia para a avaliação de qualidade subjetiva de vídeo digital usando parâmetros objetivos. Embora se acredite que a melhor avaliação da qualidade de vídeo é feita por observadores humanos, é necessário o desenvolvimento de medidas objetivas de qualidade de vídeo comprimido. Medidas que são necessárias para organizações de padronização, usuários finais e provedores de serviços. Medidas objetivas imparciais, confiáveis, repetíveis e de baixo custo, aumentam a competição entre os provedores de serviços e equipamentos e, também, permitem melhor especificação e avaliação de novos sistemas.

Foram propostos descritores objetivos calculados para sistemas com referência total e referência reduzida, obedecendo aos limites estabelecidos pelo Grupo de Experts de Qualidade de Vídeo (VQEG). Deseja-se que estas medidas forneçam valores absolutos dos erros de codificação, e que se desenvolvam regras para sua concatenação, possibilitando assegurar o nível de qualidade desejado ao longo da cadeia de produção.

Os principais elementos de um sistema de transmissão de vídeo digital foram descritos, enfatizando-se o funcionamento dos equipamentos para compressão e descompressão, denominados *codecs* de vídeo. Este foi um passo importante para obter a maior quantidade de informação possível sobre o equipamento para que se pudesse propor métodos para sua avaliação. Foram apresentadas também as ferramentas de codificação do consagrado formato MPEG-2, “Motion Picture Group”, utilizado atualmente para transmissão de sinais de TV Digital.

Podem ser introduzidos vários tipos de degradações no processo de compressão. Estas degradações são, em alguns casos, perceptíveis e perturbam o usuário, embora um dos objetivos dos sistemas de compressão seja escondê-los. As deteriorações podem ser introduzidas através de filtragem, conversão, transformação, quantização e/ou transmissão, sendo assim influenciadas pelo algoritmo de compressão utilizado, conteúdo da imagem, origem do material-fonte e condições do canal de transmissão. Perda de resolução, efeito de bloco, ruído de quantização e erros de bloco são alguns dos indesejados efeitos do processo de compressão.

Um ponto importante para o estudo de sistemas de avaliação de qualidade de vídeo digital é entender, mesmo que superficialmente, as curvas de resposta do Sistema Visual Humano. Assim, uma vez conhecidas estas limitações, pode-se determinar o quanto da imagem está sendo captada pelos fotorreceptores do observador e qual a influência do ruído aleatório no processo de inteligibilidade.

A avaliação de qualidade subjetiva, que utiliza a opinião de um grupo de observadores humanos, exige altos custos, é difícil de ser realizada e por isso se torna inviável. O crescente número de novos algoritmos de codificação de imagens faz com que a realização destes testes subjetivos não seja factível. Por outro lado, na avaliação objetiva, obtida automaticamente, são extraídas características das imagens originais, antes da codificação, e imagens degradadas, após a codificação. A avaliação objetiva com “Referência Total” permite uma quantidade ilimitada de informação de referência para o processamento, o que auxilia no processo. Contudo, a exigência cada vez maior de medidas de qualidade em tempo real, incentiva o estudo de técnicas de avaliação objetiva “Em-Serviço”, que utiliza uma quantidade reduzida de informação da imagem original.

O grande número de propostas apresentadas pelos grupos pertencentes ao VQEG, para avaliação com “Referência Total”, mostra a grande diversidade de métodos que podem ser propostos para avaliação de qualidade de vídeo. A complexidade dos algoritmos são também características marcantes, se forem comparados o número grande de passos do algoritmo da Tektronix com a proposta de segmentação por regiões da Fundação CPqD. Apesar disso, houve grande similaridade nos resultados alcançados por estes grupos na primeira fase de testes [1]. O uso da unidade de medida subjetiva JND, proposto inicialmente pela NASA, representa uma inovação nos critérios de obtenção de descritores, e os grupos que utilizarem esta medida na próxima fase do VQEG devem ter atenção especial.

A metodologia utilizada nos experimentos realizados buscou um número grande de parâmetros, na tentativa de alcançar valores próximos da base subjetiva. A combinação dos melhores parâmetros pelo sistema de otimização por redes neurais apresentou-se eficaz, sendo que cada um dos parâmetros pôde sanar deficiências dos outros. Por outro lado, a combinação de um número grande de descritores no processo de treinamento dificulta a convergência para uma solução ótima, podendo comprometer os resultados finais.

As medidas objetivas para sistemas com referência total não apresentaram grandes inovações em relação às propostas presentes na literatura atual. O Pico da Relação Sinal-Ruído (PSNR), única medida recomendada pela União Internacional de Telecomunicações

(ITU) até o momento, juntamente com o Descritor de Regiões Planas, geraram as maiores correlações com a base subjetiva sem se combinarem com outros parâmetros. Quando todos os parâmetros foram combinados, o treinamento envolvendo o PSNR, o descritor de Regiões Planas, Contraste Colorido, Sobel e Variância de Colunas gerou os melhores resultados. Os níveis de correlação encontrados para este sistema de referência foram altos, sendo compatíveis com os níveis encontrados pelos grupos da primeira fase de testes do VQEG [1] em 2000.

As propostas para avaliação com “Referência Reduzida” geraram resultados excelentes. A combinação entre a medida de Dimensão Fractal e o Descritor de Blocação garantiu boas correlações até mesmo para imagens consideradas críticas. O treinamento realizado com os demais descritores não implicou em melhores resultados, provando a eficácia da combinação destes dois descritores. A medida de Dimensão Fractal necessita de atenção especial, pois utiliza um único valor como referência e apresenta alta correlação com a base subjetiva. Contudo, as dilatações morfológicas sucessivas, presentes na implementação deste descritor, tornam o processamento lento. O Descritor de Blocação proposto possui a capacidade de detectar o tipo de degradação mais comum e mais perceptível ao sistema visual humano, possibilitando a comparação entre duas imagens binárias com blocação em destaque. Tal descritor pode, ainda, ser usado para sistemas de avaliação sem referência, observando-se o número de linhas verticais e horizontais obtidas no processo. Este pode ser um caminho muito promissor e que deve ser considerado em pesquisas futuras.

Além do uso do Descritor de Blocação para sistemas sem referência, pode ser sugerido o cálculo da Dimensão Fractal para regiões específicas de uma imagem, ponderando-as de acordo com sua influência no Sistema Visual Humano. Apesar dos resultados alcançados neste trabalho e pela Fundação CPqD comprovarem a pequena relevância das degradações temporais no processo de avaliação, este tipo de deterioração pode ser considerado, analisando-se simultaneamente uma sequência de quadros de vídeo.

Diante dos resultados obtidos neste trabalho, pode-se constatar que as medidas propostas para sistemas de avaliação de qualidade de *codecs* de vídeo são eficazes,

dispensando a presença de observadores humanos. Além disso, algumas delas podem ser utilizadas em avaliações em tempo real, utilizando pequena quantidade de informação de referência e mantendo altos níveis de correlação. A partir disso, pode ser considerada a execução de um número maior de testes com diferentes tipos de cenas, e a submissão de algumas das propostas no processo de disputa pela padronização mundial de avaliação de qualidade de vídeo digital.

7 Apêndice I

Apresenta-se aqui a documentação das funções implementadas e suas funções auxiliares em linguagem Matlab, referentes a cada um dos métodos propostos no Capítulo 4. Para a criação das funções foi utilizada a *Toolbox de Morfologia Matemática – Morph*¹. Tal documentação foi gerada pelo ambiente de desenvolvimento de Software Científico (ADESSO), baseado em linguagem XML. Algumas das funções possuem versões para os dois sistemas de referência abordados: “Em-Laboratório”, ou com “Referência Total” (sigla *lab*), e “Em-Serviço”, ou com “Referência Reduzida” (sigla *serv*).

I- PSNR

vqlabPSNR – PSNR

Synopsis

`y = vqlabPSNR(fref, fdeg)`

Input

`fref`: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem original

`fdeg`: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem degradada

Output

`y`: Gray-scale (uint8 or uint16) or binary image (logical uint8).

Description

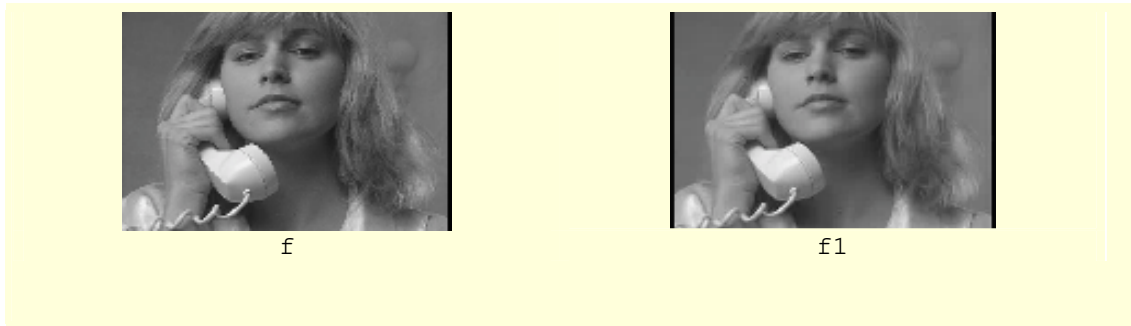
Calcula o pico da relação sinal ruído (PSNR) entre duas imagens.

Examples

```
f=mmviff1read('F:\vqeg\sus00.100');
f1= mmviff1read('F:\vqeg\sus08.100');
f=double(f(:,: ,1));
f1=double(f1(:,: ,1));
iashow(f);
[720 x 480x1] Min= 12, Max=252, Mean=113.687, Std=37.80
iashow(f1);
[720 x 480x1] Min= 3, Max=254, Mean=113.117, Std=38.70
y=vqlabPSNR(f,f1)

y =
    3.6398
```

1- www.mmorph.com



Equation

$$PSNR = 20 \cdot \log_{10} \left[\frac{Y_{peak}}{\sqrt{\frac{1}{(N_h N_v)} \sum_{j=0}^{N_h-1} \sum_{i=0}^{N_v-1} [Y_{ref}(i, j) - \hat{Y}_{proc}(i, j)]^2}} \right]$$

onde N_h e N_v são o número linhas e colunas das imagens de entrada; e $Y_{peak}=255$

Source code

```
fref=double(fref);
fdeg=double(fdeg);
res_MSE=(sum(sum(fref-fdeg).^2));
res_MSE=res_MSE/(size(fref,1)*size(fref,2)*size(fref,3));
y=10*log10(255/res_MSE);
```

II- Regiões Planas

vqlabsegm - Regiões planas

Synopsis

```
y = vqlabsegm( f, h, A, a )
```

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada: original ou degradada
h: mmdouble Limiar de intensidade de borda. **Default:** 7.
A: mmdouble Área mínima para a região ser homogênea. **Default:** 200.
a: mmdouble Área mínima para ser região de textura. **Default:** 20.

Output

y: Gray-scale (uint8 or uint16) or binary image (logical uint8).

Description

A função extrai as regiões planas de uma imagem em nível de cinza. As regiões planas são definidas como as porções da imagem que possuem pequena variação de intensidade. O usuário pode especificar 3 dos parâmetros importantes no processo: h- que limita a profundidade dos mínimos regionais usados como marcadores, A- que limita o número de pixels para que uma região seja considerada plana, e a- área mínima para ser uma região de textura. Inicialmente, a imagem original é submetida a um filtro sequencial para uma discreta suavização. A imagem resultante é subtraída por ela mesma erodida, extraindo-se seu gradiente e gerando f2. A partir desta imagem, obtém-se uma imagem de marcadores através dos mínimos regionais com profundidades não superiores a h. Realiza-se então a transformada de watershed de f2 com a imagem de marcadores obtida. Para remoção de elementos com área inferior a 'A' (regiões planas) e 'a' (regiões de textura) realiza-se, respectivamente, uma abertura por área e um fechamento por área.

Examples

```
f=mmvifflread('F:\vqeg\mob00.100');
f=double(f(:,:,1));
iashow(f);
[720 x 480x1] Min= 1, Max=254, Mean=147.083, Std=61.20
y=vqlabsegm(f);
iashow(y);
[720 x 480x1] Min= 0, Max= 1, Mean=0.531, Std= 0.50
```



f



y

Source code

```
if(nargin)==1
h=7; A=200; a=20;
elseif (nargin)==2
A=200; a=20;
elseif (nargin)==3
a=20;
end

disco=mmstdisk(8,'2D','EUCLIDEAN');
f1=mmasfrec(f,'OC');
f2=mmgradm(f1,mmsecross(0),mmsecross);
f3=mmregmin(mmhbasin(f2,h));
f4=mmareaopen(mmneg(mmcwatershed(f2,f3)),A);
y=mmareaclose(f4,a);
```

vqservsegm - Valor médio de regiões planas

Synopsis

```
[y,y_flat] = vqservsegm( f, H, A, a )
```

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada: original ou degradada

H: mmdouble Limiar de intensidade de borda. **Default:** 7.

A: mmdouble Área mínima para a região ser homogênea. **Default:** 200.

a: mmdouble Área mínima para a região ser de textura. **Default:** 20.

Output

[y,y_flat]: mmdouble

Description

Extraí as regiões planas e também o valor médio do número de pixels pertencentes a regiões planas de uma imagem em nível de cinza. As regiões planas são definidas como as porções da imagem que possuem pequena variação de intensidade. O usuário pode especificar três parâmetros importantes no processo: h- que limita a profundidade dos mínimos regionais usados como marcadores, A- que limita o número de pixels para que uma região seja considerada plana, e a- área mínima para ser uma região de textura. Inicialmente, a imagem original é submetida a um filtro sequencial para uma discreta suavização. A imagem resultante é subtraída por ela mesma erodida, extraindo-se seu gradiente e gerando f2. A partir desta imagem, obtém-se uma imagem de marcadores através dos mínimos regionais com profundidades não superiores a h. Realiza-se então a transformada watershed de f2 com a imagem de marcadores obtida. Para remoção de elementos com área inferior a 'A' (regiões planas) e 'a' (regiões de textura) realiza-se, respectivamente, uma abertura por área e um fechamento por área. O resultado deste processo gera a imagem de saída y_flat, respectiva às regiões planas. Finalmente, é extraída a média do somatório dos pixels da imagem resultante do estágio anterior.

Examples

```
f=mmviff1read('F:\vqeg\mob00.100');
f=f(:, :, 1);
iashow(f);
[720 x 480x1] Min= 1, Max=254, Mean=147.083, Std=61.20
[y,y_flat]=vqservsegm(f);
y

y =
    254.9722
```

Source code

```
disco=mmssedisk(8,'2D','EUCLIDEAN');
f1=mmasfrec(f,'OC');
f2=mmgradm(f1,mmsecross(0),mmsecross);
f3=mmregmin(mmhbasin(f2,h));
f4=mmareaopen(mmneg(mmcwatershed(f2,f3)),A);
y_flat= mmareaclose(f4,a);
y=mean(sum(y_flat));
```

III – Dimensão Fractal

vqfractal - Dimensão fractal

Synopsis

```
y = vqfractal( f, range, option )
```

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada em nível de cinza: original ou degradada

range: Gray-scale (uint8 or uint16) or binary image (logical uint8). Vetor com raios linearmente incrementados do disco usado na dilatação. Seleciona a porção de parametrização que determina a inclinação da linha. **Default:** [5,15].

option: mmstring Mostra opcionalmente o gráfico para visualizar a inclinação. 'PLOT' ou 'NOPLOT'. **Default:** PLOT.

Output

y: Gray-scale (uint8 or uint16) or binary image (logical uint8).

Description

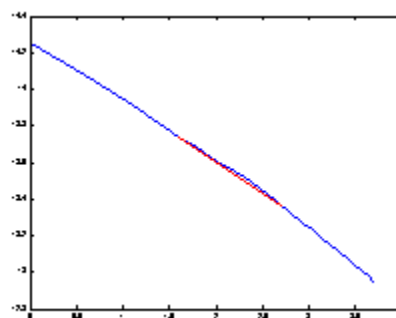
Calcula-se a dimensão fractal de uma imagem em nível de cinza, utilizando o modelo Minkowski Sausage. Uma estimativa robusta da dimensão fractal é obtida considerando um disco plano He de diâmetro e. Devido à natureza multi-escala do algoritmo, dilatações podem ser realizadas para todos os tamanhos de elemento estruturante. Assim, uma mudança de escala resulta apenas em um deslocamento da linha de regressão de $\log[V(e)]$ versus e, ao longo do eixo x. Com isso, a dimensão fractal pode ser calculada através do volume V(e) de superfícies externas. Desta forma é calculado o vetor de medidas do decaimento logaritmo natural da subtração entre as áreas da imagem original dilatada e a da imagem original, dividido pelo diâmetro do disco Euclidiano usado na dilatação. Neste processo o diâmetro do elemento estruturante varia de 1 a 40, e o usuário pode determinar a região do gráfico Vxe onde será calculada a inclinação. Esta inclinação ou coeficiente angular gera a dimensão fractal da imagem em nível de cinza de entrada.

Examples

```
f=mmviff1read('F:\vqeg\sus00.100');
f=double(f(:,:),1);
iashow(f);
y=vqfractal(f,[5,15],'PLOT')
y = 1.3415
```



(f)



(plot)

Equation

$$D_{fractal} = - \left(-1 + \frac{\Delta(\log(S_e))}{\Delta(\log(e))} \right)$$

onde

$$S_e = \frac{V[(f \oplus He) - f]}{e}$$

sendo: V o volume de uma superfície; He um disco Euclidiano de raio e

Source code

```
mmfreedom(2);
if nargin==1
    range=[5 15];
    option='PLOT';
elseif nargin==2
    option='PLOT';
end
fi=f;
for i=1:40
    fi=mmdil(fi,mmsecross(1));
    fcross=mmsubm(fi,f);
    D(i)=sum(sum(fcross))/i;
end
i=1:40;
x=log(range);
y=log(D(range));
[p s]=polyfit(x,y,1); % p(1) e a inclinação
if strcmp(upper(option),'PLOT')
    plot(log(i),log(D),'-'.x,polyval(p,x),'r');
    figure(gcf);
end
y=-(-1+p(1));
```

IV – Descritor de Blocagem

vqblock - Blocagem

Synopsis

`[S,ws] = vqblock(f, range, option)`

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada
range: Gray-scale (uint8 or uint16) or binary image (logical uint8). Dimensões do bloco para análise, dimensões mínimas 3x3. **Default**: [8 8].
option: mmstring Especifica análise completa ou das bordas do bloco. **Default**: full.

Output

[S, ws]: Gray-scale (uint8 or uint16) or binary image (logical uint8).

Description

A função retorna o vetor proporcional a blocagem de uma imagem em nível de cinza. O usuário pode especificar as dimensões do bloco padrão, para análise particionada, pela variável de entrada range, com dimensões mínimas 3x3. Pode-se também definir se a análise será de todo o bloco "full" ou somente das bordas "edge".

Examples

```
f=mmviff1read('F:\vqeg\sus00.100');
f1=mmviff1read('F:\vqeg\sus08.100');
f=f(:, :, 1);
f1=f1(:, :, 1);
[S, ws]=vqblock(f);
S1=S; ws1=ws;
[S, ws]=vqblock(f1);
iashow(f);
[720 x 480x1] Min= 12, Max=252, Mean=113.687, Std=37.80
iashow(f1);
[720 x 480x1] Min= 3, Max=254, Mean=113.117, Std=38.70
iashow(ws1);
[720 x 480x1] Min= 0, Max= 1, Mean=0.207, Std= 0.40
iashow(ws);
[720 x 480x1] Min= 0, Max= 1, Mean=0.210, Std= 0.41
disp('Parâmetro de avaliação de blocagem:');
bl=mean(abs(double(S)-double(S1)))

bl =
    4.6476
```



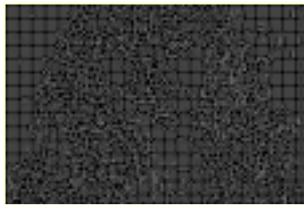
(f)



(f1)



(ws)



(ws1)

Equation

$$S = \sum_{x=i.range(1)}^{(i+1).range(1)-1} \sum_{y=j.range(2)}^{(j+1).range(2)-1} ws(x, y) \text{ com } i \in [0, N_i] \text{ e } j \in [0, N_j]$$

com N_i e N_j sendo o número de blocos na horizontal e vertical respectivamente.

$$ws = watershed(f_{grad}, f_m)$$

com f_{grad} sendo o gradiente da imagem f e f_m a imagem de marcadores descrita a seguir.

$$f_m(i, j) = \begin{cases} 1 & \text{se } i = \frac{range(1)}{2} + (N_i - 1).range(1) \text{ ou } j = \frac{range(2)}{2} + (N_j - 1).range(2) \\ 0 & \text{caso contrario} \end{cases}$$

Source code

```
mmfreedom(2);
if nargin <= 2
    option='full';
end

if nargin <= 1
    range=[8 8];
end

fgrad=mmgradm(f,mmsecross(1),mmsecross(1));

fm_aux=zeros(range(1),range(2));
fm_aux(ceil(size(fm_aux,1)/2),ceil(size(fm_aux,2)/2))=1;
fm= repmat(fm_aux,[ceil(size(f,1)/range(1)) ceil(size(f,2)/range(2))]);
fm=fm(1:size(f,1),1:size(f,2));
fm=mmbinary(fm);

ws=mmwatershed(fgrad,fm,mmsecross);
% chamada da funcao que retorna o vetor do somatorio dos pixels pertencentes aos blocos com "range"
especificados pelo usuario
S=vqanalisebloco(ws,range,'sum',option);
```

V – Descritor de Fourier

mqdefourier - Descritores de Fourier

Synopsis

```
[hr,hs] = mqdefourier( f, nr, ns )
```

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada: original ou degradada

nr: mmdouble Número de regiões radiais **Default:** 4.

ns: mmdouble Número de regiões setoriais **Default:** 4.

Output

[hr,hs]: mmimage

Description

Extrai o vetor dos descritores de Fourier de uma imagem em nível de cinza. São criadas regiões nos formatos radial e setorial, sendo o número de regiões especificado pelo usuário. Desta forma se extrai um valor médio dos pixels pertencentes ao espectro de Fourier da imagem original limitado pelas regiões. Os descritores expressam as distribuições de orientação e frequência espacial na imagem, sendo tipicamente usados para caracterizar textura. Enquanto baixas frequências significam áreas mais uniformes (em relação à distribuição de nível de cinza), altas frequências tipicamente caracterizam bordas ou algum tipo de ruído.

Examples

```
f=mmviff1read('F:\vqeg\sus00.100');
f=f(:, :, 1);
[hr,hs]=mqdefourier(f,4,4)

hr =
    1.0e+004 *
    1.4277    0.0040    0.0019    0.0014
hs =
    1.0e+004 *
    0.0240    1.3280    0.0230    0.0344
```

Equation

$$D_{\text{radial}}(i) = E(F_i)$$

sendo F_i o espectro de Fourier da imagem original pertencente a região i ; e E o valor médio da intensidade dos pixels

$$D_{\text{setorial}}(j) = E(F_j)$$

sendo F_j o espectro de Fourier da imagem original pertencente a região j

Source code

```
if nargin==1
nr=4;
ns=4;
elseif nargin==2;
ns=4;
end
f=f(:,:,1);
F=double(abs(fftshift(fft2(f))).^2);
F=F./(size(f,1)*size(f,2));

% criando regioes
[frad,fset]=vqcriareg(f,nr,ns);

% analise radial
for j=1:nr
    fs=double(zeros(size(f)));
    fs(frad==j)=1;
    hr(j)=mean(mean(F.*fs));
end

% analise setorial
for j=1:ns
    fs=double(zeros(size(f)));
    fs(fset==j)=1;
    hs(j)=mean(mean(F.*fs));
end
```

vqcriareg - Cria regiões radiais e setoriais

Synopsis

```
[frad,fset] = vqcriareg( f, nrad, nset )
```

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada: original ou degradada

nrad: mmdouble Número de regiões radiais. **Default:** 4.

nset: mmdouble Número de regiões setoriais. **Default:** 4.

Output

[frad, fset]: Gray-scale (uint8 or uint16) or binary image (logical uint8).

Description

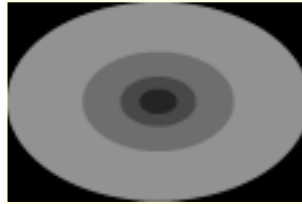
A função retorna duas imagens rotuladas contendo regiões radiais e setoriais respectivamente. Pode ser usada em conjunto com outras função para extração de parâmetros em regiões predeterminadas. O usuário pode especificar a quantidade de rótulos ou regiões das imagens de saída, ou optar pelo valor default de 4 regiões.

Examples

```
f=mmviff1read('F:\vqeg\sus00.100');
f=f(:, :, 1);
[frad, fset]=vqcriareg(f);
iashow(f);
[720 x 480x1] Min= 12, Max=252, Mean=113.687, Std=37.80
iashow(frad);
[720 x 480x1] Min= 0, Max= 7, Mean=2.883, Std= 1.61
iashow(fset);
[720 x 480x1] Min= 0, Max= 4, Mean=2.492, Std= 1.12
```



(f)



(frad)



(fset)

Equation

Geração de regiões radiais

$$S_{rad}[k] = \sum_{\sqrt{u^2+v^2}=r_k}^{\sqrt{u^2+v^2}=r_{k+1}} F_I(u, v)$$

para

$$\sqrt{u^2 + v^2} < 0.5$$

onde u e v são as coordenadas da imagem normalizadas entre -0.5 e 0.5; e r é o raio definido para uma região k

Geração de regiões setoriais

$$S_{set}[k] = \sum_{\arctan(v/u)=t_k}^{\arctan(v/u)=t_{k+1}} |F_I(u, v)|^2$$

para

$$\sqrt{u^2 + v^2} < 0.5$$

onde $t_{\{k\}}$ e $t_{\{k+1\}}$ indicam as inclinações limite para as regiões.

Source code

```
mmfreedom(2);
if nargin==1
nrad=4;
nset=4;
elseif nargin==2;
nset=4;
end
faux=double(zeros(size(f)));
i=1:size(f,1);
j=1:size(f,2);
fi= (0.5/(size(f,1)/2))*i-0.5;
fj= (-0.5/(size(f,2)/2))*j+0.5;
[f1 f2]=meshgrid(fj,fi);
frad=zeros(size(f));
fset=zeros(size(f));

% criando regioes radiais
fm=f1.*f1+f2.*f2;
passo=.5/nrad;
raio_inf=.25;
raio_sup=.5;
k=nrad;
while k>0
    frad_aux=(fm<=raio_sup^2)&(fm>=raio_inf^2);
    frad=frad+k*frad_aux;
    k=k-1;
    raio_inf=(raio_inf./2);
    if k==1
        raio_inf=0;
    end
    raio_sup=(raio_sup./2);
end

% criando regioes setoriais
passo=pi/nset;
ang_inf=0;
ang_sup=pi/nset;
f2(f2==0)=1;
k=1;
while ang_sup<=pi/2
    setor_pos=and(atan(f1./f2)>ang_inf,atan(f1./f2)<ang_sup);
    setor_neg=and(atan(f1./f2)>-ang_sup,atan(f1./f2)<-ang_inf);
    fset=fset+k*setor_pos+(k+1)*setor_neg;
    ang_inf=ang_inf+passo;
    ang_sup=ang_sup+passo;
    k=k+2;
end
```

VI – Sobel

vqlabsobel - Descritor de Sobel

Synopsis

```
y = vqlabsobel( f )
```

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada: original ou degradada

Output

y: Gray-scale (uint8 or uint16) or binary image (logical uint8).

Description

Aplica-se o filtro de Sobel em uma imagem de nível de cinza, ou na componente de Luminância em uma imagem no formato YCbCr. São aplicados filtros Sobel para detecção de bordas horizontais e verticais na imagem de entrada, e em seguida é extraída a magnitude destas imagens filtradas.

Examples

```
f=mmviff1read('F:\vqeg\mob00.100');
f=double(f(:,:,1));
y=vqlabsobel(f);
iashow(f);
[720 x 480x1] Min= 1, Max=254, Mean=147.083, Std=61.20
iashow(y);
[720 x 480x1] Min= 0, Max=1032, Mean=143.125, Std=183.90
```



(f)



(y)

Equation

$$f_{sobel} = \sqrt{f_x^2 + f_y^2}$$

onde

$$f_x = conv(f, S_x)$$

e

$$f_y = conv(f, S_y)$$

com

$$S_x = \begin{bmatrix} -1 & 0 & -1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

e

$$S_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

sendo conv o operador de convolução.

Source code

```
SOBEL_Y = [-1 -2 -1;...
0 0 0;...
1 2 1];
SOBEL_X = [-1 0 1;...
-2 0 2;...
-1 0 1];

imgSobel_x = conv2(f,SOBEL_X,'same');
imgSobel_y = conv2(f,SOBEL_Y,'same');

y = sqrt(imgSobel_x.^2 + imgSobel_y.^2);
```

vqservsobel - Filtro de Sobel (Em-Serviço)

Synopsis

```
[y,y_sobel] = vqservsobel( f )
```

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada: original ou degradada

Output

[y,y_sobel]: Gray-scale (uint8 or uint16) or binary image (logical uint8).

Description

Esta função retorna a razão entre o desvio padrão da média e a média da média de uma imagem filtrada pela máscara de Sobel. Retorna-se também a imagem de entrada filtrada por Sobel. Aplica-se filtros Sobel para detecção de bordas horizontais e verticais na imagem de entrada, e em seguida é extraída a magnitude destas imagens filtradas.

Examples

```
f=mmviff1read('F:\vqeg\mob00.100');
f=f(:,:,1);
iashow(f);
[720 x 480x1] Min= 1, Max=254, Mean=147.083, Std=61.20
[y,y_sobel]=vqservsobel(f);
y
y =
    0.2972
iashow(y_sobel);
[720 x 480x1] Min= 0, Max=1032, Mean=143.125, Std=183.90
```



(f)



(y_sobel)

Equation

$$y = \frac{\text{std}(\text{mean}(f_{\text{sobel}}))}{\text{mean}(\text{mean}(f_{\text{sobel}}))}$$

sendo: *std* e *mean* os operadores de desvio padrão e média.

$$f_{\text{sobel}} = \sqrt{f_x^2 + f_y^2}$$

onde

$$f_x = \text{conv}(f, S_x)$$

e

$$f_y = \text{conv}(f, S_y)$$

com

$$S_x = \begin{bmatrix} -1 & 0 & -1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

e

$$S_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

sendo conv o operador de convolução.

Source code

```
SOBEL_Y = [-1 -2 -1;...
0 0 0;...
1 2 1];
SOBEL_X = [-1 0 1;...
-2 0 2;...
-1 0 1];

imgSobel_x = conv2(f,SOBEL_X,'same');
imgSobel_y = conv2(f,SOBEL_Y,'same');
y_sobel=sqrt(imgSobel_x.^2 + imgSobel_y.^2);

y=std(mean(y_sobel))./mean(mean(y_sobel));
```

VII – Contraste Colorido e Nível de Cinza

vqcontrastecor - Contraste colorido.

Synopsis

```
[y,med,des] = vqcontrastecor( f )
```

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada RGB (3 bandas): original ou degradada

Output

[y,med,des]: Gray-scale (uint8 or uint16) or binary image (logical uint8).

Description

O contraste colorido é calculado para uma imagem fonte no formato RGB. Retorna-se também a média e o desvio padrão deste contraste. Realiza-se o produto escalar médio dos vetores nos espaços RGB pelos valores de suas vizinhanças. Valores altos de D{RGB} para cada produto escalar indica regiões de cores uniformes, valores baixos correspondem a regiões não-uniformes.

Examples

```
f=mmviff1read('F:\vqeg\bal00.100');
f=vqycbcr2rgb(f);
iashow(f);
[720 x 480x3] Min= 0, Max=255, Mean=81.846, Std=81.97
[y,med,des]=vqcontrastecor(f);
```

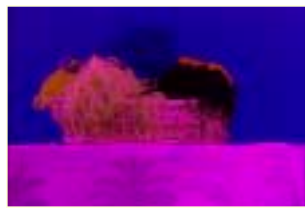
```

iashow(y);
[720 x 480x1] Min= 0, Max=130059, Mean=39558.778, Std=30314.61
med

med =
    3.9559e+004
des

des =
    3.0315e+004

```



(f)



(y)

Equation

$$D_{RGB} = \frac{v_{(x,y)} \cdot v_{(x+1,y)} + v_{(x,y)} \cdot v_{(x,y+1)} + v_{(x,y)} \cdot v_{(x-1,y)} + v_{(x,y)} \cdot v_{(x,y-1)}}{4}$$

onde v é o vetor das respectivas componentes RGB da imagem de entrada

Source code

```

f1=double(f); % convertendo a imagem para RGB
B=double([0 .25 0;.25 0 .25;0 .25 0]);

D_R=f1(:,:,1).*(conv2(f1(:,:,1),B,'same'));
D_G=f1(:,:,2).*(conv2(f1(:,:,2),B,'same'));
D_B=f1(:,:,3).*(conv2(f1(:,:,3),B,'same'));
y=D_R+D_G+D_B;

med=mean(y(:));
des=std(y(:));

```

vqcontrastegray - Contraste de nível de cinza

Synopsis

```
[y,med,des] = vqcontrastegray( f )
```

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada em nível de cinza: original ou degradada

Output

[y, med, des]: Gray-scale (uint8 or uint16) or binary image (logical uint8).

Description

Extrai-se o contraste de uma imagem em nível de cinza, retornando também a média e o desvio padrão deste contraste. Neste processo, a luminância da imagem de entrada é submetida a um operador Laplaciano sendo calculados a média e o desvio padrão deste resultado.

Examples

```
f=mmviff1read('F:\vqeg\sus00.100');
f=f(:, :, 1);
iashow(f);
[720 x 480x1] Min= 12, Max=252, Mean=113.687, Std=37.80
[y,med,des]=vqcontrastegray(f);
iashow(y);
[720 x 480x1] Min=-54, Max=104, Mean=0.176, Std= 4.80
med
med =
    0.1758
des
des =
    1.2677
```



(f)



(y)

Equation

$$D_{gray}(x, y) = \frac{(4f(x, y) - (f(x+1, y) + f(x, y+1) + f(x-1, y) + f(x, y-1))))}{4}$$

Source code

```
mmfreedom(2);
A=double([0 -.25 0;-.25 1 -.25;0 -.25 0]);
y=conv2(double(f),A,'same');
med=mean(mean(y));
des=std(std(y));
```

VIII – Variação de Histograma

vqlabhist - Histograma normalizado.

Synopsis

```
y = vqlabhist( f )
```

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada: original ou degradada

Output

y: Gray-scale (uint8 or uint16) or binary image (logical uint8).

Description

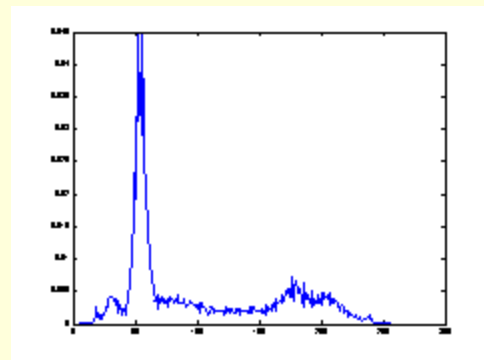
Retorna o histograma de uma imagem em nível de cinza normalizado pelo produto de suas dimensões. O histograma de uma imagem representa um parâmetro muito seletivo, já que dificilmente duas imagens reais possuem o mesmo histograma. A partir disso, pode-se comparar os histogramas entre uma imagem original e sua correspondente degradada, verificando suas variações em determinados pontos.

Examples

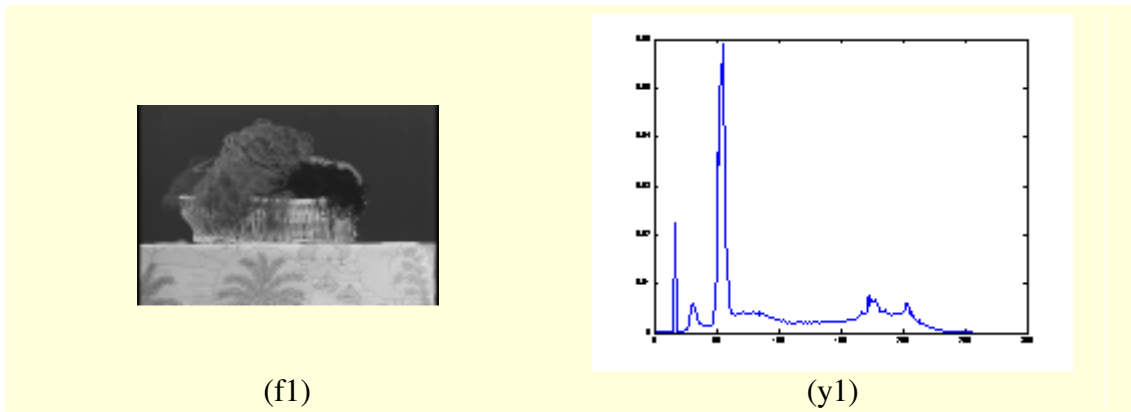
```
f=mmviffread('F:\vqeg\bal00.100');
f1=mmviffread('F:\vqeg\bal08.100');
f=double(f(:,:,1));
f1=double(f1(:,:,1));
y=vqlabhist(f);
Warning: Converting image from double to uint16.
iashow(f);
[720 x 480x1] Min= 1, Max=254, Mean=100.718, Std=61.08
plot(y);
y1=vqlabhist(f1);
Warning: Converting image from double to uint16.
iashow(f1);
[720 x 480x1] Min= 1, Max=254, Mean=100.131, Std=59.91
plot(y1);
```



(f)



(y)



Equation

$$H(i) = \text{card}(f == i)$$

onde: *card* é a cardinalidade do conjunto; e *i* é a amplitude dos pixels da imagem *f*

Source code

```
hf=zeros(1,255);
haux=double(mmhistogram(f));
haux=haux';
hf(1:size(haux,2))=haux;
y=hf./(size(f,1)*size(f,2));
```

vqservhist - Variância de histograma

Synopsis

```
[h,hf] = vqservhist( f )
```

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada em nível de cinza: original ou degradada

Output

```
[h,hf]: mmdouble
```

Description

Calcula-se a variância de histograma de uma imagem em nível de cinza. Retorna-se também o histograma normalizado da imagem para uma comparação visual do usuário.

Examples

```
f=mmviff1read('F:\vqeg\bal00.100');
f1=mmviff1read('F:\vqeg\bal08.100');
f=f(:,:,1);
f1=f1(:,:,1);
[h,hf]=vqservhist(f);
```

```

h
h =
    3.4623e+003
[720 x 480x1] Min= 1, Max=254, Mean=100.718, Std=61.08
[h,hf]=vqservhist(f1);
h
h =
    5.0766e+003

```

Equation

$$Var(H(i)) = Var(card(f == i))$$

onde: *card* é a cardinalidade do conjunto; *i* é a amplitude dos pixels da imagem *f*; e *Var* é o operador variância.

Source code

```

hf=zeros(1,255);
haux=double(mmhistogram(f));
haux=haux';
hf(1:size(haux,2))=haux;
h=var(hf)/mean(hf);

```

IX – Variância por colunas e blocos

vqlabvar - Variância das colunas.

Synopsis

```
y = vqlabvar( f )
```

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada: original ou degradada

Output

y: Gray-scale (uint8 or uint16) or binary image (logical uint8).

Description

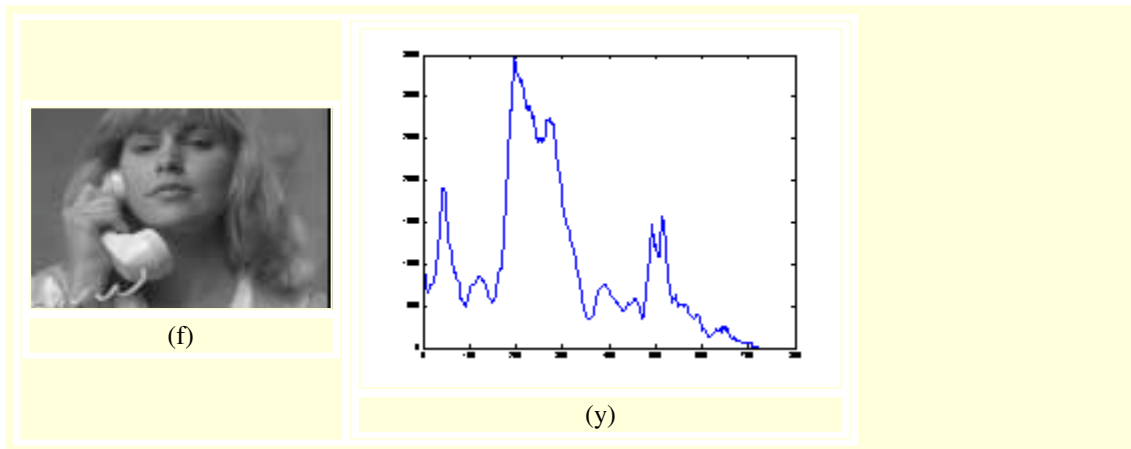
Calcula as variâncias das colunas de uma imagem em nível de cinza.

Examples

```

f=mmviff1read('F:\vqeg\sus00.100');
f=double(f(:,: ,1));
iashow(f);
[720 x 480x1] Min= 12, Max=252, Mean=113.687, Std=37.80
y=vqlabvar(f);
plot(y);

```

Equation

$$V(y) = \sigma^2 [f(x, y)] = E [f^2(x, y)] - E^2 [f(x, y)]$$

onde $E(f)$ representa a média de f .

Source code

```
mmfreedom(2);
f=double(f);
y=double(var(f));
```

vqservvar - Variâncias (Em-Serviço).

Synopsis

```
y = vqservvar( f )
```

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada: original ou degradada

Output

y: mmdouble

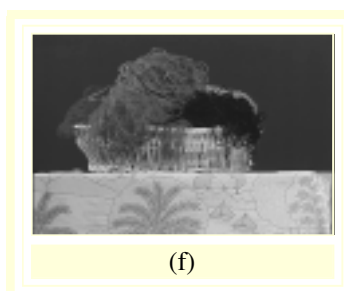
Description

Calcula-se a razão entre o desvio padrão e a média das variâncias de uma imagem em nível de cinza.

Examples

```
f=mmviff1read('F:\vqeg\bal00.100');
f=f(:, :, 1);
iashow(f);
[720 x 480x1] Min= 1, Max=254, Mean=100.718, Std=61.08
y=vqservvar(f)

y =
    0.3160
```

**Equation**

$$V_{service} = \frac{std(V(y))}{mean(V(y))}$$

$$V(y) = \sigma^2 [f(x, y)] = E [f^2(x, y)] - E^2 [f(x, y)]$$

onde $E(f)$ representa a média de f

Source code

```
f=double(f);
faux=double(var(f));
y=std(faux)/mean(faux);
```

vqparbloco - Parâmetros por bloco

Synopsis

```
y = vqparbloco( f, range, oper )
```

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada: original ou degradada

range: Gray-scale (uint8 or uint16) or binary image (logical uint8). Dimensões dos blocos de análise, dimensões mínimas 3x3. **Default**: [8 8].

oper: mmstring Especifica a operação que será realizada em cada bloco. **Default**: var.

Output

y: mmdouble

Description

Calcula parâmetros dos blocos de uma imagem em nível de cinza. O usuário pode especificar as dimensões do bloco pela variável de entrada range, com dimensões mínimas de 3x3, tendo como valor default: 8x8. A operação a ser realizada em cada bloco pode ser especificada pela variável oper, tendo as opções: "var" (default), "std", "mean" e "sum".

Examples

```
f=ones(6,6);
f(2:2:end,2:2:end)=0
```

```
f =
    1     1     1     1     1     1
    1     0     1     0     1     0
    1     1     1     1     1     1
    1     0     1     0     1     0
    1     1     1     1     1     1
    1     0     1     0     1     0
y=vqparbloco(f,[3,3],'sum')

y =
     8
     7
     7
     5
```

Equation

$$S = \sum_{x=i.range(1)}^{(i+1).range(1)-1} \sum_{y=j.range(2)}^{(j+1).range(2)-1} f(x,y) \text{ com } i \in [0, N_i] \text{ e } j \in [0, N_j]$$

N_i e N_j sendo o número de blocos na horizontal e vertical respectivamente.

Source code

```
mmfreedom(2);
f=double(f);
if nargin <= 2
    oper='var';
end

if nargin <= 1
    range=[8 8];
end
if min(range) < 3
    error('As dimensões dos blocos devem ser maiores que 2');
end

faux=ones(range(1),range(2));
faux(1,:)=0;
faux(end,:)=0;
faux(:,1)=0;
faux(:,end)=0;
fm1= repmat(faux,[ceil(size(f,1)/range(1)) ceil(size(f,2)/range(2))]);
fm1=fm1(1:size(f,1),1:size(f,2));
fr=mmlabel(fm1); % rotulando os blocos de avaliação
fr=mmdil(fr,mmsebox);

if ~strcmp(oper,'var')
    y=mmgrain(fr,f,oper,'data'); % calculando a operacao a partir dos blocos rotulados
else
    y=(mmgrain(fr,f,'std','data')).^2; % calculando a variancia a partir dos blocos rotulados
end

148
```

vqanalisebloco - Análise por blocos

Synopsis

```
y = vqanalisebloco( f, range, oper, option )
```

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada
 range: Gray-scale (uint8 or uint16) or binary image (logical uint8). Dimensões do bloco para análise, dimensões mínimas 3x3. **Default:** [8 8].
 oper: mmstring Especifica a operação que será realizada em cada bloco. **Default:** sum.
 option: mmstring Especifica análise completa ou das bordas do bloco. **Default:** full.

Output

y: Gray-scale (uint8 or uint16) or binary image (logical uint8).

Description

A função retorna o vetor de uma operação realizada com os pixels pertencentes a blocos de uma imagem. O usuário pode especificar as dimensões do bloco pela variável de entrada range, com dimensões mínimas de 3x3. A operação a ser realizada em cada bloco pode ser especificada pela variável oper, tendo as opções: "var" (default), "std", "mean" e "sum". Pode-se também especificar se a análise será de todo o bloco "full" ou somente das bordas "edge".

Examples

```
f=ones(6,6);
f(2:2:end,2:2:end)=0

f =
     1     1     1     1     1     1
     1     0     1     0     1     0
     1     1     1     1     1     1
     1     0     1     0     1     0
     1     1     1     1     1     1
     1     0     1     0     1     0

y=vqanalisebloco(f,[3 3],'sum','full')

y =
     8
     7
     7
     5

f=mmviff1read('F:\vqeg\sus00.100');
f1=mmviff1read('F:\vqeg\sus08.100');
f=f(:, :, 1);
f1=f1(:, :, 1);
y1=vqanalisebloco(f,[8 8],'var','edge');
y2=vqanalisebloco(f1,[8 8],'var','edge');
y=mean(abs(double(y1)-double(y2)))

y =
    15.1578
```

Equation

$$S = \sum_{x=i.range(1)}^{(i+1).range(1)-1} \sum_{y=j.range(2)}^{(j+1).range(2)-1} f(x, y) \text{ com } i \in [0, N_i] \text{ e } j \in [0, N_j]$$

N_i e N_j sendo o número de blocos na horizontal e vertical respectivamente.

Source code

```
mmfreedom(2);

if nargin <= 3
    option='full';
end

if nargin <= 2
    oper='sum';
end

if nargin <= 1
    range=[8 8];
end

if min(range)< 3
    error('As dimensões dos blocos devem ser maiores que 2');
end

faux=ones(range(1),range(2));
faux(1,:)=0;
faux(end,:)=0;
faux(:,1)=0;
faux(:,end)=0;
fm1=repmat(faux,[ceil(size(f,1)/range(1)) ceil(size(f,2)/range(2))]);
fm1=fm1(1:size(f,1),1:size(f,2));
fr=mmlabel(fm1); % rotulando os blocos de avaliação
fr=mmdil(fr,mmsebox);
fm1(fm1~=0)=255;
if strcmp(option,'edge')
    fr=mmintersec(mmneg(uint8(fm1)),fr);
end

if ~strcmp(oper,'var')
    y=mmgrain(fr,f,oper,'data'); % calculando a operacao a partir dos blocos rotulados
else
    y=(mmgrain(fr,f,'std','data')).^2; % calculando a variancia a partir dos blocos rotulados
end
```

X – Vida Média

vqhlife - Vida média

Synopsis

```
y = vqhlife( f, option )
```

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada em nível de cinza: original ou degradada.

option: mmstring Opção para construir o gráfico que mostra a variação entre os valores de sigma e os valores de variância. **Default:** 'PLOT'.

Output

y: Gray-scale (uint8 or uint16) or binary image (logical uint8).

Description

Extrai-se o parâmetro de vida média de uma imagem em nível de cinza. A vida média de uma imagem representa o valor de sigma, parâmetro de um filtro Gaussiano, que suaviza a imagem de entrada até que a variância da imagem filtrada atinja metade da variância da imagem original de entrada. Neste processo a imagem é filtrada no domínio da frequência, assim o efeito de suavização será maior com o declínio do valor de sigma. Um valor inicial de 20 para o sigma inicial foi adotado, por gerar uma suavização muito discreta.

Examples

```
f=mmviffileread('F:\vqeg\sus00.100');
f=f(:, :, 1);
mmshow(f);
y=vqhlife(f);
```

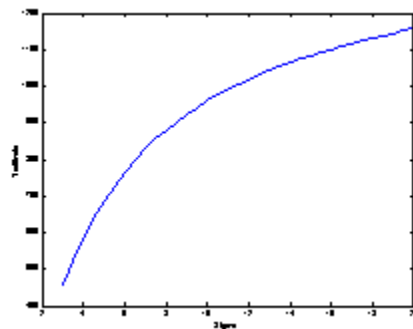
y

y =

3



(f)



(plot)

Equation

$$\sigma_{half-life} = \sigma [(Var_{\sigma=0}) / 2]$$

onde $Var_{\sigma=0}$ representa o valor da variância da imagem original sem suavização.

Source code

```
f=double(f);
var_ref=mean(var(f));
var_gauss=var_ref;
sigma(1)=20;
passo=.5;
i=1;
while var_gauss>=var_ref/2
    % criando o filtro gaussiano
    Fg=iagaussian(size(f),size(f)/2+1,[sigma(i)^2 0;0 sigma(i)^2]);
    Fg=ianormalize(Fg,[0 1]);

    F=fft2(f); % aplicando a dft na imagem original
    FH=F.*fftshift(double(Fg));
    fh=ifft2(FH);
    var_gauss(i)=var(fh(:));
    sigma(i+1)=sigma(i)-passo;
    i=i+1;
end
if strcmp(option,'PLOT')
    plot(sigma(1:end-1),var_gauss);
    xlabel('Sigma');
    ylabel('Variância');
end
```

XI – Entropia

vqentropia - Entropia

Synopsis

`y = vqentropia(f)`

Input

f: Gray-scale (uint8 or uint16) or binary image (logical uint8). Imagem de entrada: original ou degradada

Output

y: Gray-scale (uint8 or uint16) or binary image (logical uint8).

Description

Extraí o valor da entropia de uma imagem nível de cinza. A entropia de uma imagem fornece o conteúdo de informação. É um conceito vindo da teoria de informação, sendo usada como medida de

complexidade, assumindo a formação da imagem como um processo estocástico. Uma imagem com alto valor de entropia revela uma imagem rica em detalhes.

Examples

```
f=mmviffread('F:\vqeg\sus00.100');
f=f(:,:,1);
iashow(f);
y=vqentropia(f)
y = 7.0985
```



(f)

```
disp('Teste para a imagem anterior degradada');
Teste para a imagem anterior degradada
f1=mmviffread('F:\vqeg\sus08.100');
f1=f1(:,:,1);
iashow(f1);
[720 x 480x1] Min= 3, Max=254, Mean=113.117, Std=38.70
y1=vqentropia(f1)
y1 =
    7.0834
```



(f1)

Equation

$$H(f) = - \sum_{k=0}^{K-1} h(k) \log(h(k))$$

onde h é o histograma normalizado da imagem f .

Source code

```
histo=double(mmhistogram(f))./(size(f,1)*size(f,2));
i=find(histo~=0);
h=histo(i);
y=-sum( (h) .* log2 (h) );
```


8 Bibliografia

- 1- Home Page: “*Video Quality Experts Groups*”, www.crc.ca/vqeg, Dezembro (2001).
- 2- Cover, Thomas e Thomas, Joy, “*Elements of Information Theory*”, Wiley, 1991.
- 3- Richard E. Blahut, “*Principles and Practice of Information Theory*”, Addison-Wesley, 1987.
- 4- ITU-R Recommendation BT.601.5; “*Studio encoding parameters of digital television for standard 4:3 and wide screen 16:9 aspect ratios*”; 1995.
- 5- Haskell, Barry G.; Puri, Atul; Netravali, Arun N.; “*Digital video: An introduction to MPEG-2*”, International Thompson Publishing, 1996.
- 6- Le Gall, Didier J.; “*The MPEG video compression algorithm*”; Signal Processing: Image Communication, vol. 4 n° 2 pp. 129-140, 1992.
- 7- ANSI T1.801.03.1996 “*American National Standard for Telecommunications - Digital Transport of Video Teleconferencing/ Videoconferencing Signals - Performance Terms, Definitions and Examples*”, Alliance for Telecommunications Industry Solutions, 11 West 42nd Street, New York, New York DC 10036.
- 8- Home page: “*Digital Video Artefacts*” <http://www.michaeldvd.com.au/Articles/VideoArtefacts/VideoArtefacts.html> , Setembro 2001.
- 9- Wolf Stephen, “*Measuring the End-to-End Performance of Digital Video Systems*”, IEEE Trans. Broadcast., Vol.43, No. 3, pp. 320-328, September, 1997.
- 10- SBT/IEEE International Telecommunications Symposium 498-503 “*Video Quality Assessment using Objective Parameters based on Image Segmentation*”; Antonio Claudio F. Pessoa, Alexandre X. Falcão, Ana Eliza F. Silva, Ricardo M. Nishihara, Roberto A. Lotufo , vol. 2 , 1998.
- 11- J.G. Robson, “*Spatial and Temporal Contrast-Sensitivity Functions of the Visual System*”, J.Opt. Soc. Am., pp. 1141-1142, August (1966).

- 12- Video Quality Experts Group, ***“Final Report from the Video Quality Experts Group on the Validation of Objective Models of Video Quality Assessment”***, March, 2000.
- 13- Wandell B. A., ***“Foundations of Vision”***, Sinauer Associates, Inc., 1995.
- 14- Stockman A. , ***“Spectral Sensitivities of the Human Cones”***, Journal of the optical society of America A, vol. 10m n° 12, pp. 2491-2521, Dec. 1993.
- 15- Poirson A.B., Wandell B.A., ***“Pattern-color Seperable Pathways Predict Sensitivity to Simple Colored Patterns”***, Vision Res., vol. 36, n° 4, pp 515-526, 1996.
- 16- Winkler S., ***“A Perceptual Distortion Metric for Digital Color Video”***, Proc. SPIE Human Vision and Electronic Imaging Conference, vol. 3644, pp 175-184, San Jose, California, January 23-29, 1999.
- 17- Sarnoff Corp., ***“Vision Model Algorithm Description and Testing”***, VQEG, August, 1997.
- 18- Wyszecki G. ; Stiles W. S., ***“Color Science”***, Sec. Edition, Wiley, 1982.
- 19- Sankarelli M. J. , Mullen K.T., ***“Estimation of the L-, M-, e S-cone Weights of the Postreceptoral Detection Mechanisms”***, Journal Opt. Soc. Am. A, vol. 13, n° 5, pp. 906-915, 1996.
- 20- Faria e Silva, A.E., ***“Avaliação de Sistemas de Redução de Taxa de Bits de Vídeo com Enfoque na Percepção do Usuário”***, Tese de Mestrado, DECOM-FEEC-UNICAMP, fevereiro de 1998.
- 21- ITU-R Recommendation BT.500-7, ***“Methodology for the Subjective Assessment of the Quality of Televion Pictures”***, 1995.
- 22- ITU-R Recommendation BT.710-2, ***“Subjective Assessment Methods for Image Quality in High-Definition Television”***, 1994.
- 23- ITU-R Recommendation P.910, ***“Subjective Video Quality Assessment Methods for Multimedia Applications”***, 1996.
- 24- Pessoa, A.C.F., ***“Métodos para Avaliação Objetiva de Codecs de Vídeo”***, Relatório Técnico CPqD/DTS/SVI/PAPDS 02-97, Abril, 1997.

- 25- Wolf, S.; Pinson, M.H.; Voran, S.D.; Webster, A.A. , ***“Objective quality assessment of digitally transmitted video”***, Communications, Computers and Signal Processing, IEEE Pacific Rim Conference on, Page(s):477-482, vol.2, 1991.
- 26- Stephen Wolf e Margaret H. Pinson, ***“In-Service Performance Metrics for MPEG-2 Video Systems”***, Made to Measure 98 - Measurement Techniques of the Digital Age Technical Seminar, jointly sponsored by the International Academy of Broadcasting (IAB), the International Telecommunications Union (ITU), and the Technical University of Braunschweig (TUB), Montreux, Switzerland, November 12-13, 1998.
- 27- Tan, K.T.; Ghanbari, M., ***“Blockiness detection for MPEG2-coded video”*** IEEE Signal Processing Letters , Volume: 7 Issue: 8 , Page(s): 213 –215, Aug. 2000.
- 28- Caviedes, J. ; Jung, J. , ***“No-Reference Metric for a Video Quality Control Loop”***, Visual Communication on Image Processing, pp. 791-800, Perth, June 2000.
- 29- Watson, Andrew B., ***“Measurement of a JND Scale for Video Quality”***, IEEE Doc. G- 2.1.6/112, April 2000.
- 30- Recommendation ITU-R BT.500-7, ***“Subjective Quality Assessment Methods of Television Pictures”***, 1995.
- 31- Recommendation ITU-R BT.910-9, ***“Subjective Video Quality Assessment Methods for Multimedia Applications”***, 1996.
- 32- Pessoa, A.C. F. ; Falcão,A.X.; Silva, A.E., Nishihara, R.M.; Lotufo, R.A., ***“Video Quality Assessment using Objective Parameters based on Image Segmentation”***, SBT/IEEE International Telecommunications Symposium 498-503, vol. 2 , 1998.
- 33- Lotufo R.A.; Falcao A.X.,***“The ordered queue and the optimality of the watershed approaches”***, Mathematical Morphology and its Application to Image and Signal Processing, pp. 341--350., Palo Alto, USA, Kluwer Academic Publishers, June 26-28 2000.
- 34- Lubin, J. , ***“The use of psychophysical data and models in the analysis of display system performance”***, In A.B. Watson (ed.), *Digital Images and Human Vision*, MIT Press, pp. 163-178, 1993.
- 35- Lubin, J. , ***“A visual system discrimination model for imaging system design and evaluation”***, in E. Peli (ed.), *Visual Models for Target Detection and Recognition*, World Scientific Publishers, 1995.

- 36- Watson, A.B., ***“Toward a perceptual video quality metric in Human Vision”***, Visual Processing and Digital Display VIII., San Jose, CA: SPIE, Bellingham, WA., 1998.
- 37- Watson, A.B., ***“Design and performance of a digital video quality metric in Human Vision”***, Visual Processing and Digital Display IX., San Jose, CA: SPIE, Bellingham, WA, 1999.
- 38- Beerends J.G. ; Hekstra, A.P. , ***“Objective measurement of video quality”***, ITU-T Study Group 12, Document COM 12-7, February 1997.
- 39- Wolf, S. ; Pinson, M.H., ***“Spatial-temporal distortion metrics for in-service quality monitoring on any digital video system”***, SPIE International Symposium on Voice, Video, and Data Communications, Boston, MA, 11-22 September 1999.
- 40- Wolf, S. , ***“Measuring the End-to-End Performance of Digital Video Systems”***, IEEE Trans. Broadcast., Vol.43, No. 3, pp. 320-328, September, 1997.
- 41- Kwon, O. ; Lee, C. , ***“Objective Method for Assessment of Video Quality Using Wavelets”***, ISIE, Pusan-Korea, 2001.
- 42- Kaiser, G. , ***“A Friendly Guide to Wavelets”***, Birkhauser-Boston, Library of Science Book of the Month, 1999.
- 43- Costa, L.F. ; Consularo, L.A. ; Lotufo, R.A. , ***“Data Mining-Based Modeling of Human Visual Perception”***, Medical Data Mining and Knowledge Discovery, Physica-Verlag, 2001.
- 44- R.C. Machado, R.A. Lotufo , ***“Adesso: Ambiente Computacional para Desenvolvimento Rápido de Aplicações”***, Relatório de Atividades, DCA-FEEC-UNICAMP/FCTI-Campinas, maio de 2000.
- 45- Draft Technical Report A3, ***“Objective Video Quality Measurement using a Peak-Signal-to-Noise-Ratio (PSNR) Full Reference Technique”***, NTIA/ITS, Project T1A1-05, January, 2001.
- 46- Gonzalez, R. ; Wintz, P. , ***“Digital Image Processing”***, Addison Wesley, 1987.
- 47- Hearn, D. ; Baker, M.P. , ***“Computer Graphics”***, 2nd edition Prentice Hall, 1994.
- 48- Mandelbrot, B. , ***“Les objets fractals: forme, hasard et dimension”***, Flammarion, Paris, 1975.

- 49- Carr, J. ; Benzer, W. , ***“On the practice of estimating fractal dimension”***, Math. Geol. 23, pp. 945-958, 1991.
- 50- Soille, P. ; Rivest, J.F. , ***“On the Validity of Fractal Dimension Measurements in Image Analysis”***, JVCIR(7), pp. 217-229, 1996.
- 51- Castleman, K. , ***“Digital Image Processing”***, Prentice Hall, 1996.
- 52- Zuben, F. J. V., ***“Modelos paramétricos e não paramétricos de redes neurais artificiais e aplicações”***, Tese de Doutorado, Faculdade de Engenharia Elétrica e de Computação, Unicamp, 1996.
- 53- Zuben, F. J. V., ***Apostila do Curso de Redes Neurais***, UNICAMP, novembro, 2001.
- 54- Friedman, J. H. ; Stuetzle, W. , ***“Projection Pursuit Regression”***, Journal of the American Statistical Association (JASA), 76 (376): 817-823, 1981.
- 55- Von Zuben, F.J., Netto, M.L.A. ***“Unit-growing learning optimizing the solvability condition for model-free regression”***. Proceedings of the IEEE International Conference on Neural Networks, vol. 2, pp.795-800, November 1995.
- 56- Von Zuben, F.J., Netto, M.L.A. ***“Projection Pursuit and the Solvability Condition Applied to Constructive Learning”***. Proceedings of the IEEE Joint International Conference on Neural Networks, vol. 2, pp. 1062-1067, June1997.
- 57- Friedman, J. H. ; Stuetzle, W., Shroeder, A., ***“Projection Pursuit Regression”***, Journal of the American Statistical Association (JASA), vol.79, n° 387, pp. 599-608, 1984.
- 58- Friedman, J. H. & Tukey, J. A., ***“Projection Pursuit Algorithm for Exploratory Data Analysis”***, IEEE Transactions on Computers, 23(9):881-890, 1974.
- 59- Huber, P.J., ***“Projection Pursuit”***, The Annals of Statistics, 13(2):435-475, 1985.
- 60- Jones, M. C. & Sibson, R. , ***“What is Projection Pursuit”***, Journal of the Royal Statistical Society A, 150(1): 1-36,1987.
- 61- Hwang, J.N., Lay, S.R., Maechler, M. , Martin, R.D., Schimert, J., ***“Regression modeling in back-propagation an projection pursuit learning”***, IEEE Transaction on Neural Networks, 5(3): 342-353, 1994.

