UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

LEONARDO CRAUSS DARONCO

Avaliação Subjetiva de Qualidade Aplicada à Codificação de Vídeo Escalável

Dissertação apresentada como requisito parcial
para a obtenção do grau de
Mestre em Ciência da Computação

Prof. Dr. José Valdeni de Lima
Orientador

Porto Alegre, março de 2009

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Daronco, Leonardo Crauss

Avaliação Subjetiva de Qualidade Aplicada à Codificação de Vídeo Escalável / Leonardo Crauss Daronco. – Porto Alegre: PPGC da UFRGS, 2009.

146 p.: il.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2009. Orientador: José Valdeni de Lima.

1. Codificação de vídeo escalável. 2. Avaliação subjetiva de qualidade. 3. Avaliação de qualidade de vídeo. 4. Transmissão multimídia. 5. H.264 SVC. I. Lima, José Valdeni de. II. Título.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS 5

LISTA DE FIGURAS 7

LISTA DE TABELAS 9

RESUMO 10

ABSTRACT 11

1 INTRODUÇÃO 12

2 CONCEITOS E TRABALHOS RELACIONADOS 16
2.1 Codificação de vídeo 16
2.2 Codificação de vídeo escalável 22
2.2.1 Escalabilidade temporal 23
2.2.2 Escalabilidade espacial 25
2.2.3 Escalabilidade de qualidade (SNR) 27
2.2.4 Outras técnicas de escalabilidade 28
2.2.5 Escalabilidade no H.264 SVC 31
2.3 Transmissão multimídia 34
2.4 Avaliação de qualidade de vídeo 36
2.4.1 Metodologias subjetivas 37
2.4.2 Métodos objetivos 42
2.5 Projeto SAM 45
2.6 Trabalhos relacionados 49

3 DESENVOLVIMENTO DO TRABALHO 53
3.1 Objetivos e contextualização 53
3.2 Definição do plano de avaliação 56
3.2.1 Configurações de codificação 57
3.2.2 Padrões de instabilidade 62
3.3 Seleção e processamento dos vídeos 64
3.3.1 Pré-seleção 65
3.3.2 Pré-processamento 67
3.3.3 Seleção final 68
3.3.4 Codificação escalável 72
3.3.5 Simulação da instabilidade 78
3.4 Execução das avaliações subjetivas 82

4 APRESENTAÇÃO DOS RESULTADOS 87
4.1 Análise inicial 87
4.2 Apresentação de todos os votos atribuídos 89
4.3 Média dos votos para SRCs e HRCs 93
4.4 Resultados da instabilidade 94
4.5 Análise dos resultados em relação às taxas de codificação 96
4.6 Comparação entre os métodos de escalabilidade 98

5 CONCLUSÕES 102
5.1 Trabalhos futuros 104

REFERÊNCIAS 106

APÊNDICE A OBJETIVOS PROPOSTOS PARA AVALIAÇÕES SUBJETIVA DE VíDEO ESCALÁVEL 113
A.1 II - Avaliação dos métodos de escalabilidade 113
A.2 III - Avaliação dos métodos de escalabilidade com variação nas camadas 115
A.3 IV - Quantidade de camadas 117
A.4 V - MGS vs. Escalabilidade espacial 119

APÊNDICE B APLICATIVOS DESENVOLVIDOS 122
B.1 TI & SI 122
B.2 LYUV 123
B.3 wxSVQ 125

APÊNDICE C PROCESSAMENTO DOS VíDEOS 127
C.1 Pré-processamento 127
C.1.1 AviSynth 127
C.1.2 VirtualDub 128
C.1.3 FFmpeg 129
C.2 Codificação escalável 129

APÊNDICE D DADOS DAS AVALIAÇÕES SUBJETIVAS 139
D.1 Instruções impressas 139
D.2 Questionário 140
D.3 Faixa etária e gênero dos avaliadores 141
D.4 Resultados dos questionário 142
D.5 Relatório dos votos de todo avaliadores 142

LISTA DE ABREVIATURAS E SIGLAS

ANSI	American National Standards Institute
AVC	Advanced Video Coding
CIF	Common Intermediate Format
DPCM	Differential Pulse Code Modulation
EBU	European Broadcasting Union
FGS	Fine Grain Scalability
FPS	Frames Per Second
GOP	Group Of Pictures
HD	High Definition
HRC	Hypothetical Reference Circuit
ISO	International Organization for Standardization
IEC	ISO International Electrotechnical Commission
ITS	Institute for Telecommunication Sciences
ITU	International Telecommunication Union
ITU-R	ITU Radiocommunication Sector
ITU-T	ITU Telecommunication Standardization Sector
JPEG	Joint Photographic Experts Group
JVT	Joint Video Team
MPEG	Moving Pictures Experts Group
NTSC	National Television System Committee
PAL	Phase Alternating Line
PSNR	Peak Signal-to-Noise Ratio
PVS	Processed Video Sequence
QCIF	Quarter Common Intermediate Format
RMSE	Root Mean Square of the Error
SAM	Sistema Adaptativo Multimídia

SD	Standard Definition
SNR	Signal-to-Noise Ratio
SRC	Source Reference Signal
SVC	Scalable Video Coding
VCEG	Video Coding Experts Group
VQM	Video Quality Metric(s)

LISTA DE FIGURAS

Figura 2.1: Diagrama padrão para um *codec* baseado em DCT com compensação de movimento. 17
Figura 2.2: Diagrama temporal dos padrões de codificação de vídeo MPEG e ITU-T. 20
Figura 2.3: Exemplo de codificação escalável temporal utilizando quadros B na camada adicional. 24
Figura 2.4: Codificação escalável temporal utilizando subsequências de quadros. 24
Figura 2.5: Exemplo da criação da pirâmide laplaciana para redução dos quadros. 26
Figura 2.6: Diagrama de um codificador com escalabilidade espacial. 26
Figura 2.7: Diagrama de um decodificador com escalabilidade espacial. 27
Figura 2.8: Organização dos quadros na codificação escalável de qualidade. . . . 28
Figura 2.9: Exemplo de escalabilidade por particionamento de dados. 29
Figura 2.10: Exemplo de uma sequência de coeficientes com os *bit-planes* formados. 30
Figura 2.11: *Bit-planes* da figura 2.10 codificados em RLE. 30
Figura 2.12: Exemplo de uso do *bit-plane shifting*. 31
Figura 2.13: Exemplo de estrutura de quadros B hierárquicos. 32
Figura 2.14: Sequência de execução de uma avaliação utilizando ACR. 40
Figura 2.15: Escalas de votação para a metodologia ACR. 40
Figura 2.16: Exemplo de uma tela para aplicação da metodologia SAMVIQ. . . . 41
Figura 2.17: Sensibilidade do sistema visual humano. 43
Figura 2.18: Diagrama padrão para métodos perceptuais de avaliação objetiva. . . 43
Figura 2.19: Visão geral da arquitetura do SAM. 46
Figura 2.20: Exemplo de vídeo codificado com o Vebit em 5 camadas. 48

Figura 3.1: Simulação de protocolos de controle de congestionamento exibindo a variação das camadas (e banda) ao longo da transmissão. 55
Figura 3.2: SRCs, HRCs e PVSs. 56
Figura 3.3: Diferença entre as camadas temporais quando utilizada uma camada com 15 fps e quando não utilizada. 60
Figura 3.4: Exemplo dos resultados da codificação de um dos vídeos utilizados, chamado “rushfieldcuts”. 61
Figura 3.5: Variações das camadas ao longo do tempo nos padrões de instabilidade. 63
Figura 3.6: Área utilizada para aplicação do filtro de Sobel e cálculo da medida SI. 70
Figura 3.7: Valores TI e SI para os vídeos coletados, com destaque para aqueles que foram selecionados. 71
Figura 3.8: Um quadro de exemplo para cada um dos 11 vídeos selecionados. . . 72
Figura 3.9: Etapas do processo de codificação escalável utilzando o JSVM. . . . 75

Figura 3.10: Gráficos do PSNR e taxas de codificação para os primeiros 4 SRCs utilizados na avaliação. 78
Figura 3.11: Gráficos do PSNR e taxas de codificação para os últimos 4 SRCs utilizados na avaliação. 79
Figura 3.12: Gráficos do PSNR e taxas de codificação para todos os SRCs utilizados para treinamento. 80
Figura 3.13: Exemplo da replicação de pixels para ampliação da resolução espacial. 80
Figura 3.14: Exemplo de quadros codificados do SRC “redkayak”. 82
Figura 3.15: Exemplo dos documentos utilizados para teste visão. 84
Figura 3.16: Exemplo de telas do aplicativo para execução das avaliações subjetivas. 85
Figura 3.17: Fases das avaliações de qualidade. 86

Figura 4.1: Exemplo dos votos utilizados para cálculo dos valores MOS. 89
Figura 4.2: Votos, média, intervalo de confiança e desvio padrão dos SRCs. . . . 90
Figura 4.3: Votos, média, intervalo de confiança e desvio padrão dos SRCs (continuação). 91
Figura 4.4: Maior e menor correlação entre os SRCs. 93
Figura 4.5: MOS para todos SRCs e HRCs. 94
Figura 4.6: Análise do MOS_h em relação à instabilidade. 95
Figura 4.7: Análise da qualidade em relação à taxa de codificação dos vídeos. . . 97
Figura 4.8: Análise da qualidade em relação à taxa de codificação para cada SRC individualmente. 99
Figura 4.9: Relação da qualidade entre os métodos de escalabilidade. 100

LISTA DE TABELAS

Tabela 2.1: Normas para avaliação de qualidade. 37
Tabela 2.2: Metodologias para execução das avaliações de qualidade de vídeo. . . 38

Tabela 3.1: Objetivos propostos para as avaliações de qualidade. 54
Tabela 3.2: Configurações para a codificação escalável. 58
Tabela 3.3: Tabela de comparação do das taxas de codificação das camadas de vídeo com e sem a camada utilizando 15 fps. 60
Tabela 3.4: Definição dos três padrões de instabilidade. 62
Tabela 3.5: Número de variações de camadas por minuto para alguns protocolos de controle de congestionamento. 63
Tabela 3.6: Variações de camadas por minuto em um ambiente com 10 fluxos concorrentes. 64
Tabela 3.7: Formatos dos vídeos coletados. 66
Tabela 3.8: Descrição dos vídeos selecionados para a **avaliação**. 73
Tabela 3.9: Descrição dos vídeos selecionados para o **treinamento**. 74
Tabela 3.10: Valores de PSNR e taxa de codificação para todos os SRCs. 81
Tabela 3.11: Condições do ambiente segundo a norma P.910. 83
Tabela 3.12: Condições do ambiente e especificações do monitor utilizado nas avaliações. 83

Tabela 4.1: Tabela de correlação entre os SRCs. 92
Tabela 4.2: Diferenças entre os padrões de instabilidade para cada SRC. 95
Tabela 4.3: Comparação entre a variação do MOS e da taxa de codificação dos vídeos. 97

RESUMO

Os constantes avanços nas áreas de transmissão e processamento de dados ao longo dos últimos anos permitiram a criação de diversas aplicações e serviços baseados em dados multimídia, como *streaming* de vídeo, videoconferências, aulas remotas e IPTV. Além disso, avanços nas demais áreas da computação e engenharias, possibilitaram a construção de uma enorme diversidade de dispositivos de acesso a esses serviços, desde computadores pessoais até celulares, para citar os mais utilizados atualmente. Muitas dessas aplicações e dispositivos estão amplamente difundidos hoje em dia, e, ao mesmo tempo em que a tecnologia avança, os usuários tornam-se mais exigentes, buscando sempre melhor qualidade nos serviços que utilizam.

Devido à grande variedade de redes e dispositivos atuais, uma dificuldade existente é possibilitar o acesso universal a uma transmissão. Uma alternativa criada é utilizar transmissão de vídeo escalável com IP multicast e controlada por mecanismos para adaptabilidade e controle de congestionamento. O produto final dessas transmissões mulimídia são os próprios dados multimídia (vídeo e áudio, principalmente) que o usuário está recebendo, portanto a qualidade destes dados é fundamental para um bom desempenho do sistema e satisfação dos usuários.

Este trabalho apresenta um estudo de avaliações subjetivas de qualidade aplicadas em sequências de vídeo codificadas através da extensão escalável do padrão H.264 (SVC). Foi executado um conjunto de testes para avaliar, principalmente, os efeitos da instabilidade da transmissão (variação do número de camadas de vídeo recebidas) e a influência dos três métodos de escalabilidade (espacial, temporal e de qualidade) na qualidade dos vídeos. As definições foram baseadas em um sistema de transmissão em camadas com utilização de protocolos para adaptabilidade e controle de congestionamento. Para execução das avaliações subjetivas foi feito o uso da metodologia ACR-HRR e recomendações das normas ITU-R Rec. BT.500 e ITU-T Rec. P.910.

Os resultados mostram que, diferente do esperado, a instabilidade não provoca grandes alterações na qualidade subjetiva dos vídeos e que o método de escalabilidade temporal tende a apresentar qualidade bastante inferior aos outros métodos. As principais contribuições deste trabalho estão nos resultados obtidos nas avaliações, além da metodologia utilizada durante o desenvolvimento do trabalho (definição do plano de avaliação, uso das ferramentas como o JSVM, seleção do material de teste, execução das avaliações, entre outros), das aplicações desenvolvidas, da definição de alguns trabalhos futuros e de possíveis objetivos para avaliações de qualidade.

Palavras-chave: Codificação de vídeo escalável, avaliação subjetiva de qualidade, avaliação de qualidade de vídeo, transmissão multimídia, H.264 SVC.

Subjective Video Quality Assessment Applied to Scalable Video Coding

ABSTRACT

The constant advances in multimedia processing and transmission over the past years have enabled the creation of several applications and services based on multimedia data, such as video streaming, teleconference, remote classes and IPTV. Futhermore, a big variety of devices, that goes from personal computers to mobile phones, are now capable of receiving these transmissions and displaying the multimedia data. Most of these applications are widely adopted nowadays and, at the same time the technology advances, the user are becoming more demanding about the quality of the services they use.

Given the diversity of devices and networks available today, one of the big challenges of these multimedia systems is to be able to adapt the transmission to the receivers' characteristics and conditions. A suitable solution to provide this adaptation is the integration of scalable video coding with layered transmission. As the final product in these multimedia systems are the multimedia data that is presented to the user, the quality of these data will define the performace of the system and the users' satisfaction.

This paper presents a study of subjective quality of scalable video sequences, coded using the scalable extension of the H.264 standard (SVC). A group of experiments was performed to measure, primarily, the efeects that the transmission instability (variations in the number of video layers received) has in the video quality and the relationship between the three scalability methods (spatial, temporal and quality) in terms of subjective quality. The decisions taken to model the tests were based on layered transmission systems that use protocols for adaptability and congestion control. To run the subjective assessments we used the ACR-HRR methodology and recommendations given by ITU-R Rec. BT.500 and ITU-T Rec. P.910.

The results show that the instability modelled does not causes significant alterations on the overall video subjective quality if compared to a stable video and that the temporal scalability usually produces videos with worse quality than the spatial and quality methods, the latter being the one with the better quality. The main contributions presented in this work are the results obtained in the subjective assessments. Moreover, are also considered as contributions the methodology used throughout the entire work (including the test plan definition, the use of tools as JSVM, the test material selection and the steps taken during the assessment), some applications that were developed, the definition of future works and the specification of some problems that can also be solved with subjective quality evaluations.

Keywords: Scalable video coding, Subjective video quality, Quality assessment, Layered transmission, Multimedia transmission, H.264 SVC.

1 INTRODUÇÃO

A utilização de dados multimídia em computadores já é uma realidade há diversos anos, envolvendo tarefas bastante conhecidas e estudadas na área da computação, como o armazenamento, a codificação e a transmissão desses dados. Aplicações e serviços que utilizam vídeo eram, até alguns anos atrás, restritos apenas a sistemas analógicos, mas rapidamente começaram a ser convertidas para o mundo digital (JACK, 2005).

Os constantes avanços nas áreas de transmissão e processamento de dados multimídia ao longo dos últimos anos permitiram essa migração para o sistema digital e a criação de aplicações e serviços baseados nesses dados multimídia, como *streaming* de vídeo, videoconferências, aulas remotas e IPTV. Além disso, avanços nas demais áreas da computação e engenharias, possibilitaram a construção de uma enorme diversidade de dispositivos de acesso a esses serviços, desde computadores pessoais até celulares, para citar os mais utilizados atualmente. Muitas dessas aplicações e dispositivos estão amplamente difundidos hoje em dia, e, ao mesmo tempo em que a tecnologia avança, os usuários tornam-se mais exigentes, buscando sempre melhor qualidade nos serviços que utilizam.

Dada a diversidade de dispositivos e redes disponíveis atualmente, um desafio é possibilitar que a transmissão multimídia se ajuste às condições de cada usuário, e uma das alternativas para resolver este problema é a integração entre codificação de vídeo escalável (OHM, 2005; SCHWARZ et al., 2007) e transmissão em múltiplas camadas (MACCANNE et al., 1996; VICISANO et al., 1998; LI et al., 2007). Através da codificação escalável, o vídeo é dividido em diversas camadas, que podem ser dispostas em diferentes fluxos para sua transmissão. Cada camada de vídeo é complementar às anteriores, ou seja, as camadas superiores adicionam qualidade às inferiores. Portanto, quanto maior o número de camadas que o usuário receber, maior será a qualidade do vídeo que ele estará visualizando.

A codificação de vídeo escalável possui métodos já bastante estudados, que podem ser divididos em três conceitos principais: escalabilidade temporal (variação no número de quadros por segundo), espacial (variação na dimensão espacial das imagens) e de qualidade (variação na medida SNR — *Signal-to-Noise Ratio* — dos quadros). As escalabilidades temporal, espacial e de qualidade já eram suportadas em padrões de codificação mais antigos, como o MPEG-2 e o H.263, mas evoluíram especialmente no atual estado da arte em codificação, o padrão H.264 e sua extensão escalável chamada SVC (*Scalable Video Coding*). Mais conceitos sobre codificação de vídeo tradicinoal e escalável, assim como os métodos de escalabilidade existentes no padrão H.264 SVC, serão apresentados nas seções 2.1 e 2.2.

Apesar da existência desses métodos de escalabilidade e sua adoção em importantes padrões de codificação, a análise da qualidade obtida por eles quando utilizando suas diferentes configurações ainda necessita maiores investigações. Por exemplo, a defini-

ção de qual método de escalabilidade causa menores degradações de qualidade dada uma taxa de codificação fixa, ou quando ocorrem variações no número de camadas utilizadas para decodificação. Esta variação no número de camadas é causada tipicamente quando há instabilidade na transmissão, que geralmente ocorre devido à existência de tráfegos concorrentes na rede e também devido ao comportamento similar ao TCP que é implementado por diversos protocolos desenvolvidos para controle de transmissões multimídia em múltiplas camadas.

Esses protocolos são chamados de protocolos de controle de congestionamento, e têm como objetivo geral verificar as condições atuais da rede e adaptar a transmissão a essas condições. As pesquisas dividem estes protocolos em duas categorias principais: taxa única e multi-taxa. Em protocolos de taxa única, o transmissor envia os dados para todos os receptores a uma taxa ajustada dinamicamente, baseada no receptor mais lento. Os protocolos multi-taxa permitem que a transmissão seja feita em mais de uma taxa de transmissão simultaneamente, onde a idéia central consiste em codificar os dados multimídia em diversas camadas e transmitir cada uma delas em um diferente grupo multicast. Recentemente, uma nova linha de pesquisa nesta área passou a unir conceitos de protocolos de taxa única e conceitos multi-taxa para desenvolver protocolos chamados híbridos. A forma como esses protocolos trabalham tem influência direta sobre a qualidade dos vídeos, pois são eles que controlam o número de camadas recebidas por cada receptor em uma transmissão em camadas. Na seção 2.3 esses modelos de transmissão serão comentados e serão apresentados alguns protocolos já desenvolvidos.

O produto final em ambientes de transmissão multimídia é o próprio dado multimídia (vídeo, áudio) que o usuário está recebendo. A qualidade destes dados é fundamental para um bom desempenho do sistema e satisfação dos usuários. Os métodos utilizados para verificar a qualidade de dados multimídia, especialmente a qualidade do vídeo, são divididos em métodos subjetivos ou objetivos. Métodos objetivos são aplicados por ferramentas automatizadas, que analisam o vídeo de entrada e o vídeo de referência (é opcional e, em alguns casos, é formado por um conjunto reduzido de dados derivados do vídeo de referência) e resultam em determinado valor (ou valores), que correspondem à qualidade estimada para o vídeo de entrada. Já as métricas subjetivas são obtidas através de avaliações envolvendo seres humanos, que usualmente são instruídos a visualizar uma série de vídeos e atribuir uma nota à cada um de acordo com sua percepção de qualidade. Apesar das dificuldades de criação de uma técnica objetiva que apresente resultados precisos para avaliação de qualidade de vídeo, os resultados da aplicação delas são obtidos de maneira muito mais simples, enquanto a aplicação de metodologias subjetivas normalmente requer mais tempo, esforço e investimento. Porém, se bem aplicadas, as avaliações subjetivas geralmente apresentam resultados confiáveis e precisos (KOZAMERNIK et al., 2005). Os conceitos fundamentais sobre avaliação de qualidade de vídeo são descritos na seção 2.4.

Esta dissertação foi desenvolvida no contexto do projeto SAM (Sistema Adaptativo Multimídia) (ROESLER, 2003), que é um sistema de transmissão multimídia cujo objetivo principal é permitir que a transmissão seja acessível para o maior número possível de receptores (universalidade da transmissão), mesmo que estes estejam localizados em ambientes heterogêneos e apresentem diferentes características (como variações na capacidade de memória, processamento, entre outros). Resumidamente, o sistema se baseia na transmissão de vídeo escalável utilizando IP multicast e usa mecanismos para adaptabilidade e controle de congestionamento. As taxas de transmissão utilizadas nas camadas de vídeo são fixas e pré-definidas, e elas são acessadas sequencialmente, ou seja, as camadas inferiores são pré-requisitos para as camadas superiores.

O protocolo proposto para controle de congestionamento no SAM é o ALM (*Adaptive Layered Multicast*), assim como o ALMTF (*ALM TCP-Friendly*), uma variação do ALM designada para uso na Internet. Estes protocolos foram inicialmente desenvolvidos e validados através do simulador de redes NS-2 e, atualmente, estão sendo implementados em um ambiente real (KROB et al., 2007). O projeto SAM serviu como ambiente modelo para as avaliações de qualidade, pois utiliza os conceitos de codificação escalável e transmissão em camadas já comentados. Mais detalhes sobre o SAM são apresentados na seção 2.5, enquanto o capítulo **??** apresenta outros trabalhos relacionados que também utilizam conceitos semelhantes aos utilizados no SAM e, principalmente, trabalhos que envolvem avaliação de qualidade de vídeo.

Diversas decisões tomadas em projetos como o SAM têm influência direta na experiência de visualização dos vídeos por parte dos receptores. Nos protocolos utilizados, normalmente são feitas tentativas frustradas de adicionar qualidade ao vídeo que acabam aumentando a instabilidade do sistema, além de provocar perdas de pacotes que também alteram na qualidade do vídeo. Outro fator que afeta a transmissão é o comportamento similar ao TCP que é implementado por grande parte dos protocolos, como já citado. Além disso, também há a preocupação em relação à quantidade de camadas utilizadas, pois o uso de um número elevado de camadas facilita a adaptação e a manutenção da justiça com outros tráfegos, porém, reduz a estabilidade da transmissão (LI; LIU, 2003).

Avaliações de qualidade podem auxiliar na análise dos problemas existentes nestes ambientes de transmissão e na tomada de decisões, tais como auxiliar na definição da configuração das camadas de vídeo conforme as características do receptor e sua banda disponível. A análise é feita sobre vídeos escaláveis, portanto os três conceitos de escalabilidade são parâmetros fundamentais, ou seja, devem ser consideradas a dimensão espacial, a dimensão temporal e a qualidade (medida PSNR) que são alteradas pelo processo de codificação.

No trabalho descrito nesta dissertação, foram realizadas avaliações subjetivas de vídeos codificados de forma escalável e utilizando os três conceitos principais de escalabilidade: temporal, espacial e de qualidade. As avaliações também foram baseadas em sistemas de transmissão em camadas e, especialmente, na instabilidade existente nestes ambientes. Os vídeos avaliados simulam comportamentos identificados nestes sistemas de transmissão em camadas, ou seja, as variações de camadas que ocorrem quando a transmissão não é estável. A instabilidade aqui citada não se refere apenas à perda de pacotes durante a transmissão, mas à qualquer mudança (na rede, nos protocolos de controle de congestionamento ou em outro componente do sistema) que provoque alteração nas camadas de vídeo recebidas. Por este motivo, o termo instabilidade é tratado na seqüência deste trabalho como um sinônimo de "variação de camadas existente durante a transmissão dos vídeos" ou "a instabilidade da transmissão que provoca variações no número de camadas recebidas para decodificação e, conseqüentemente, variação na qualidade dos vídeos".

O objetivo principal dessas avaliações de qualidade foi analisar os efeitos que a instabilidade tem sobre a qualidade dos vídeos, já que a instabilidade é considerada um dos piores problemas existentes nesses sistemas e diversos esforços são feitos para minimizá-la. As configurações de codificação utilizadas permitiram a análise da qualidade subjetiva comparando vídeos com diferentes características, como (i) estáveis e instáveis, (ii) com pouca ou bastante instabilidade, (iii) codificados com os três conceitos de escalabilidade individualmente e (iv) com variações na complexidade de codificação e no conteúdo.

Os vídeos foram codificados utilizando a extensão escalável do padrão H.264, cha-

mada SVC (*Scalable Video Coding*) (SCHWARZ et al., 2007), e apresentados para um grupo de 22 avaliadores utilizando a metodologia de avaliação subjetiva ACR (*Adjectival Categorical Rating*) (ITU-T, 1999). A instabilidade foi criada com base nos resultados vistos em simulações de protocolos de controle de congestionamento, como os que serão comentados na seção 2.3. O capítulo 3 desta dissertação contém a descrição detalhada dos objetivos das avaliações de qualidade, do plano de avaliação e de todas as etapas de seleção e processamento dos vídeos utilizados, além da descrição do processo de execução das avaliações.

As principais contribuições deste trabalho estão nos resultados das avaliações, exibidos no capítulo 4 e que mostram, principalmente, os efeitos da instabilidade na qualidade dos vídeos e a relação entre os métodos de escalabilidade. Além disso, também são contribuições a metodologia utilizada durante o desenvolvimento do trabalho (definição do plano de avaliação, uso das ferramentas como o JSVM, seleção do material de teste, execução das avaliações, entre outros), as aplicações desenvolvidas (descritas no apêndice B), alguns possíveis objetivos que foram definidos para avaliações de qualidade (exibidos no apêndice A) e a definição de alguns trabalhos futuros. As conclusões do trabalho, que comentam sobre as principais contribuições e sobre os trabalhos futuros, finalizam esta dissertação no capítulo 5.

Os apêndices incluídos no final da dissertação são comentados ao longo do trabalho conforme forem sendo utilizados. Eles incluem a descrição de alguns objetivos propostos para avaliações de qualidade de vídeo escalável (apêndice A), os aplicativos desenvolvidos (apêndice B), as etapas técnicas do processamento dos vídeos (apêndice C) e alguns dados adicionais utilizados ou obtidos durante as avaliações de qualidade (apêndice D).

2 CONCEITOS E TRABALHOS RELACIONADOS

O trabalho descrito nesta dissertação tem como base três grandes áreas de pesquisa: codificação de vídeo (mais especificamente, codificação de vídeo escalável), transmissão de dados multimídia e avaliação de qualidade de vídeo. Para entendimento do trabalho realizado e dos resultados que serão apresentados é necessário ter conhecimento sobre esses três assuntos.

Esta seção disserta sobre essas três áreas, com ênfase nas técnicas e conceitos utilizados neste trabalho. A seção 2.1 trata dos conceitos da codificação de vídeo tradicional (não escalável), enquanto a seção 2.2 fala sobre a codificação escalável. As seções 2.3 e 2.4 abordam os conceitos de transmissão multimídia e avaliação de qualidade de vídeo, respectivamente. Ao final, a seção 2.5 apresenta o projeto SAM, que utiliza os conceitos de codificação e transmissão que aqui serão comentados e que serviu como motivação e base para realização deste trabalho, e a seção 2.6 apresenta alguns trabalhos relacionados às áreas de atuação deste trabalho.

2.1 Codificação de vídeo

Codificação de vídeo é o processo de representação digital de um vídeo, que tem como principal objetivo a compressão dos dados para viabilizar seu armazenamento e transmissão. O maior problema da codificação pode ser visto como uma troca entre a compressão alcançada e o nível de fidelidade obtido após essa compressão. Ou seja, normalmente procura-se obter a maior fidelidade possível para determinada taxa de bits máxima estipulada, ou manter a menor taxa de bits possível para determinada fidelidade (SULLIVAN; WIEGAND, 2005).

Diversos modelos de codificação foram desenvolvidos ao longo dos anos com o uso de diferentes técnicas. O modelo que se tornou mais conhecido e utilizado tem como blocos principais (i) uma transformada (a transformada discreta do cosseno — DCT (K.R. RAO, 1990) —, por exemplo), para reduzir a redundância espacial; (ii) quantização; (iii) codificação entrópica; e, paralelamente, (iv) algum método para estimativa de movimento que busca redução da redundância temporal. A figura 2.1 mostra um diagrama representativo deste modelo de codificação. Neste modelo, os quadros são divididos em diversos blocos que serão codificados separadamente. O tamanho dos blocos varia conforme o padrão de codificação, mas um valor normalmente encontrado é 8x8, ou seja, blocos de 8 pixels de largura e 8 pixels de altura. Uma implementação que realiza a codificação e decodificação de vídeo normalmente é chamada de *codec* (***co****der-****dec****oder*).

Na figura 2.1, a parte superior mostra a codificação do vídeo e a inferior mostra a decodificação. O bloco *DCT* corresponde à etapa de aplicação da transformada DCT, que, resumidamente, modifica a forma de representação dos quadros para um modelo baseado

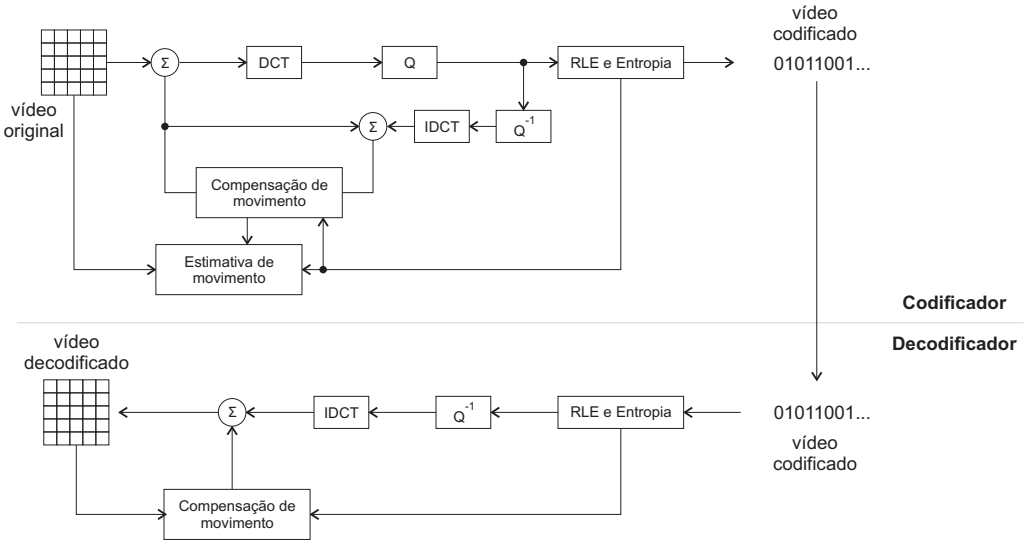


Figura 2.1: Diagrama padrão para um *codec* baseado em DCT com compensação de movimento.

em frequências, normalmente facilitando a análise e compressão desses dados (agora chamados coeficientes). Os blocos sinalizados com *Q* correspondem à quantização, processo que modifica a precisão de representação dos coeficientes para que seja possível codificá-los com um número menor de bits. A quantização é um processo que naturalmente gera perdas, mas é responsável por grande parte da compressão. Associado à etapa de transformada, a quantização pode ser aplicada de diferentes maneiras para as várias frequências, ou seja, há um controle sobre quais frequências terão maior precisão e quais terão menor precisão. Esse processo possibilita comprimir mais as áreas não visíveis ao olho humano e manter maior fidelidade nas outras.

Os blocos *RLE e Entropia* correspondem à fase de codificação (ou decodificação) entrópica, onde os coeficientes quantizados passam a ser representados de uma maneira em que é aproveitado o conhecimento sobre a probabilidade de ocorrência dos símbolos. Resumidamente, os símbolos que mais ocorrem são representados por um número reduzido de bits, enquanto os símbolos que raramente ocorrem são representados com mais bits, resultando na compressão dos dados. Ao contrário da quantização, a codificação entrópica não resulta em perdas, ou seja, os valores exatos dos dados podem ser restaurados durante a decodificação.

Os outros blocos identificam as etapas de aplicação das técnicas de estimativa de movimento para redução da redundância temporal. A estimativa de movimento utiliza informações de outros blocos do quadro que está sendo codificado ou de outros blocos de outros quadros para estimar os valores do bloco atual. O bloco atual passa então a ser representado pela diferença entre seus valores reais e os valores do outro bloco utilizado, chamado de referência. Esta diferença geralmente possui valores menores que os valores reais do bloco, portanto pode ser representada com um número menor de bits. Quanto mais próximos os valores do bloco de referência forem dos valores do bloco que está sendo codificado, menor será a diferença entre eles e maior será a compressão atingida. A busca pelos blocos de referência também é uma das etapas que mais necessitam processamento durante a codificação. Por estes motivos, o processo de estimativa de movimento é bastante estudado, e diversos métodos já foram propostos, geralmente procurando obter a melhor referência (maior compressão) com o menor custo computacional e tempo

possível.

Apesar da diversidade de nomes atribuídos a diferentes variações deste modelo, ele é normalmente conhecido por modelo híbrido entre DPCM (*Differential Pulse Code Modulation*) e compensação de movimento (será chamado apenas de modelo híbrido no restante desta dissertação).

O decodificador é, simplificadamente, uma réplica do codificador, porém executando o processo inverso, onde os blocos passam a efetuar a operação contrária da realizada durante a codificação. Inicialmente os vídeos codificado passa pelo processo de RLE e entropia, onde os códigos anteriormente atribuídos são modificados pelos valores dos coeficientes. É então realizado o processo de quantização inversa (bloco Q^{-1}) desses dados e transformada inversa (bloco IDCT), restaurando os pixels para seus valores originais. O processo de compensação de movimento também é executado quando necessário para restaurar valores de blocos codificados com base em outros blocos.

O decodificador normalmente apresenta menor complexidade que o codificador. No modelo tradicional exibido na figura 2.1 isso acontece, principalmente, devido ao processo de estimativa de movimento ser realizado apenas no codificador e não no decodificador. A etapa de estimativa de movimento é a responsável por grande parte da compactação, mas é também a que consome mais recursos entre todas do processo, portanto, o codificador acaba tornando-se mais complexo (SULLIVAN; WIEGAND, 2005).

Entre os outros modelos existentes, é válido mencionar um modelo que utiliza a transformada wavelet (GRAPS, 1995) e explora as redundâncias existentes com o uso de uma organização hierárquica dos coeficientes, através de algoritmos como EZW (MARTUCCI et al., 1997) e SPIHT (KIM et al., 2000). Mais recentemente, um novo modelo, que utiliza o teorema de Wyner-Ziv para alterar a maneira com que é feita a codificação entre diferentes quadros (redundância temporal), busca diminuir a complexidade do codificador para futuras aplicações em cenários em que ele deve ser executado com recursos limitados (um celular, por exemplo) (AARON et al., 2002; XU; XIONG, 2006). Apesar da existência desses modelos "alternativos", o modelo híbrido tornou-se o mais conhecido e difundido em aplicações reais através dos padrões MPEG/ITU que serão comentados na sequência desta seção.

As redundâncias espaciais de um vídeo podem ser reduzidas com a compressão de apenas um quadro isoladamente (compressão *intra-frame*) e as redundâncias temporais são reduzidas utilizando diversos quadros em uma sequência temporal (compressão *inter-frame*). Os quadros codificados normalmente são identificados conforme a técnica utilizada para sua compressão. Eles são normalmente categorizados como quadros I, P ou B, que representam o modo com o qual eles foram codificados. Abaixo é descrito o funcionamento dos quadros I, P e B dos padrões MPEG:

I (*Intra-coded frames*): Quadros codificados isoladamente, ou seja, aproveitando-se apenas da sua redundância espacial, sem utilizar nenhum outro quadro para redução da redundância temporal. São os quadros que necessitam maior número de bits para serem codificados e também são os quadros mais importantes, pois carregam um número maior de informações. Eles são codificados de forma semelhante à codificação de imagens estáticas, como no padrão no JPEG;

P (*Predictive-coded frames*): São quadros que, além da opção de serem codificados como os quadros I, podem ser formados a partir da codificação preditiva realizada pelos algoritmos de estimativa de movimento. Estes quadros podem ser criados a partir de quadros I ou P *anteriores* aproveitando a redundância temporal existente no vídeo

para aumentar a taxa de compressão. Quadros P podem ser utilizados em sequência, porém sequências muito grandes são evitadas para não permitir a propagação dos erros de codificação que podem existir devido à estimativa de movimento. Isto é, se um bloco possui valores com determinado erro (diferença em relação aos valores reais do bloco) e ele é utilizado como referência para codificação de outro bloco, os erros do bloco utilizado como referência serão propagados para o outro bloco. Se este novo bloco posteriormente também for utilizado como referência, o erro será propagado novamente. Assim, quanto maior for a sequência de quadros que utilizam estimativa de movimento, maior será a chance de acontecer esse tipo de erro e maior será a propagação dos erros. Para resolver o problema, são utilizados quadros I periódicamente (um exemplo seria utilizar um quadro I a cada segundo de vídeo);

B (*Bidirectionally-predictive-coded frames*): Assim como os quadros P, os quadros B são criados de forma preditiva a partir de outros quadros. A diferença entre eles é que os quadros B também permitem a utilização de quadros ainda não codificados como referência. Em um GOP (*Group of Pictures*) no formato "IBP", por exemplo, o quadro B poderia usar tanto o quadro I quanto o P como referência, mesmo neste caso onde o quadro P é codificado após o quadro B. O padrão H.264 também permite a utilização de quadros B como referência e também a utilização de blocos do mesmo quadro para predição de outros blocos.

Existem diversos outros fatores, além dos citados, que devem ser considerados durante a codificação de vídeo. Por exemplo, um vídeo pode ser progressivo ou entrelaçado, e esta característica irá influenciar em diversas etapas do processo. Para fornecer uma maneira única de tratamento dos dados e permitir a integração entre diferentes implementações de *codecs*, alguns padrões de codificação de vídeo foram criados. Estes padrões normalmente especificam como deve ser implementado o decodificador ou como deve ser organizada a *bitstream* (saída do codificador), deixando em aberto algumas questões que permitem flexibilidade na implementação, principalmente no codificador.

Os padrões utilizam técnicas avançadas de codificação e normalmente consideram fatores como eficiência de compressão, perda de qualidade devido à compressão e consumo de recursos. Diferentes padrões foram propostos ao longo dos anos, cada um com sua área de atuação específica, e são eles que possibilitam as diversas aplicações de dados multimídia existentes e, principalmente, a interoperabilidade entre elas.

Duas organizações internacionais são notáveis pelos seus padrões desenvolvidos: a ITU-T (*International Telecommunication Union - Telecommunication sector*) e a ISO/IEC (*International Organization for Standardization - International Electrotechnical Commission*). Cada uma possui um grupo destinado especificamente à codificação de vídeo. Eles são o VCEG (*Video Coding Experts Group*) da ITU-T e o MPEG (*Moving Pictures Experts Group*) da ISO/IEC. A figura 2.2 exibe um diagrama temporal de alguns dos padrões mais importantes para codificação de vídeo estabelecidos pelos grupos VCEG, MPEG e pela união deles através do JVT (*Joint Video Team*).

Os padrões MPEG normalmente são divididos em diversas partes, cada uma especificando determinadas etapas ou processos da codificação (codificação de imagens, de áudio, processo de testes, entre outros). Devido à grande quantidade de métodos e funcionalidades que eles costumam ter, para a maioria das aplicações não é necessário (e é até inviável) a implementação do padrão completo. Por isso, além de diversas partes, os padrões são divididos em diversos perfis e níveis. Perfis definem o conjunto de funcio-

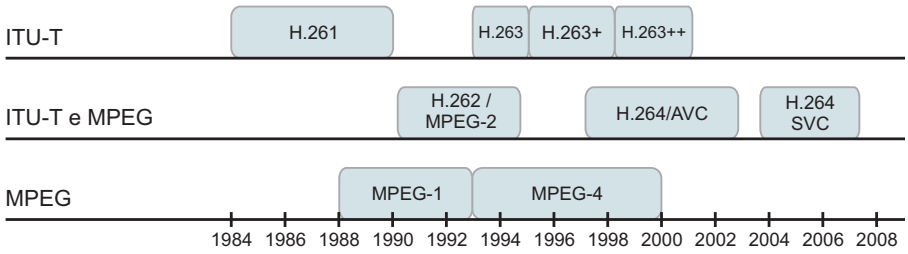


Figura 2.2: Diagrama temporal dos padrões de codificação de vídeo MPEG e ITU-T.

nalidades que serão utilizadas, como métodos para estimativa de movimento e modo de representação das cores, por exemplo. Níveis definem as capacidades quantitativas, como a taxa de bits mínima/máxima e as resoluções suportadas. O MPEG-2, por exemplo, possui um perfil chamado *Main Profile*, que especifica a possibilidade de uso de quadros I, P ou B, o formato de representação de cores 4:2:0, entre outros. Dois dos níveis para o *Main Profile* são: (i) o nível *Low Level* e (ii) o *Main Level*. O *Low Level* limita a resolução espacial a 352x288 e atinge taxa de codificação máxima de 4 Mbit/s, enquanto o *Main Level* limita a resolução em 720x576 para atingir taxas máximas de 15 Mbit/s. A combinação de perfis e níveis é utilizada para indicar o modo de codificação que é utilizado na aplicação ou que é suportado por determinado codificador/decodificador. Os perfis e níveis são especificados de forma a permitir um grande número de aplicações e facilitar seu desenvolvimento, mas também procura-se manter um número reduzido de combinações.

O MPEG-1 foi o primeiro padrão para codificação de vídeo especificado pelo grupo. Ele é baseado em blocos, utiliza DCT, quantização escalar, DPCM, compensação de movimento e é otimizado para taxas em torno de 1.2 Mbit/s. Apesar de bastante utilizado, os novos padrões acabaram superando o MPEG-1 e ele passou a ser utilizado em muito menor escala. Entre as especificações contidas no MPEG-1, uma que tornou-se bastante conhecida foi a camada 3 da especificação de áudio, conhecida por *MPEG-1 Audio Layer III*, ou apenas MP3, formato de áudio que ainda é amplamente utilizado para codificação.

Como sucessor do MPEG-1 está o MPEG-2, que foi especificado em 1994 com objetivo de suportar *broadcast* de televisão digital. Sua base é o MPEG-1, incluindo diversas mudanças e anexos para suportar as aplicações alvo. O MPEG-2 tornou-se um grande sucesso em todo o mundo e ainda é utilizado em uma grande quantidade de aplicações, como gravações de DVDs, transmissão de TV digital, televisão a cabo, entre outros.

Posteriormente foi especificado o MPEG-4, um padrão extenso que visa a representação de objetos audiovisuais. Em relação à codificação de vídeo, o MPEG-4 Part 2 é uma das especificações mais importantes dentro do MPEG-4. O mecanismo de compressão é semelhante ao MPEG-2, mas este padrão procura tratar de diversos tipos de dados, como objetos (regiões de um vídeo), redes de pontos 2D e 3D, animações e texturas. Além do MPEG-4 Part 2, o MPEG-4 Part 10 também está diretamente relacionado à codificação de imagens dentro de um vídeo. Ele é também conhecido como AVC (*Advanced Video Coding*) e H.264, nome do padrão referente à especificação idêntica feita pela ITU-T (o padrão foi especificado em conjunto pelo MPEG e VCEG).

Abaixo são relacionados os principais padrões criados pelo MPEG, com um breve comentário sobre seus objetivos e a data aproximada de sua especificação.

- **MPEG-1** (*1992*): Primeiro padrão para compressão de áudio e vídeo do MPEG. Foi utilizado para Video CD, VoD, *streaming*, entre outros.

- **MPEG-2** (*1994*): Codificação de vídeo e áudio e também define mecanismos de transporte dos dados, como no MPEG-1. O objetivo é a transmissão de televisão por *broadcast*. Utilizado em televisão digital ATSC, DVB e ISDB, TV a cabo, SVCD, DVD, entre outros.

- **MPEG-3:** Inicialmente designado para televisão de alta definição (HDTV), mas descontinuado devido à possibilidade de uso do MPEG-2 com o mesmo propósito.

- **MPEG-4** (*1998*): Codificação áudio e vídeo com suporte a objetos audiovisuais, conteúdo 3D, direitos autorais, entre outros. Expande os padrões anteriores, principalmente em relação às novas técnicas propostas para codificação de vídeo (Part 2 e Part 10/AVC/H.264).

O grupo VCEG teve como seu primeiro padrão de codificação de vídeo de maior importância o H.261, que foi lançado em meados de 1990 e foi o primeiro padrão amplamente adotado para uso em videoconferências. Para melhorar a performance do H.261, foi desenvolvido o padrão H.263 em 1995, com foco em videoconferência com baixa taxa de bits. Este padrão possui duas versões adicionais conhecidas por H.263+ e H.263++, que expandem o padrão original incluindo novas funcionalidades e melhorando a eficiência da codificação.

O VCEG também possui padrões desenvolvidos em conjunto com o MPEG. O H.262 é uma especificação idêntica ao MPEG-2 Part 2 e o H.264 é idêntico ao MPEG-4 Part 10 (AVC). O H.264 é normalmente chamado de H.264/AVC e é considerado o estado da arte atual na codificação de vídeo. Ele possui um escopo reduzido em relação ao MPEG-4, buscando eficiência na codificação e transporte apenas de quadros retangulares de vídeo (ou seja, reduz a flexibilidade do MPEG-4). O H.264 foi recentemente adotado como o padrão de codificação a ser usado pelo sistema de TV digital brasileiro (FARIAS et al., 2008).

A lista abaixo mostra os principais padrões criados pela ITU-T com um breve comentário sobre cada um.

- **H.120** (*1984*): Foi o primeiro padrão para codificação de vídeo digital. Incluía técnicas básicas como quantização escalar, VLC e, em uma segunda versão, compensação de movimento.

- **H.261** (*1990*): Primeiro padrão de codificação de vídeo que realmente foi utilizado em maior escala para videoconferências. Inicialmente projetado para operar sobre linhas ISDN com taxas de 64 kbit/s, mas para suportar taxas entre 40 kbit/s e 2 Mbit/s.

- **H.262** (*1994*): Idêntico ao MPEG-2, tendo sido desenvolvido em conjunto com o MPEG.

- **H.263** (*1995*): Uma expansão do H.261 que também baseou-se nos padrões MPEG-1 e MPEG-2. Foi originalmente projetado para compressão de vídeo visando aplicações em videoconferências.

- **H.263+** (*1998*): Também chamado H.263v2, é a segunda versão do H.263. Inclui alguns anexos à especificação do H.263 de forma a melhorar sua eficiência de codificação e incluir novas funcionalidades.

- **H.263++** (*2000*): Expande o H.263+ com a inclusão de novos anexos para melhorias na codificação.

- **H.264** (*2003*): É idêntico ao MPEG-4 Part 10 e considerado o estado da arte atualmente na codificação de vídeo. Inclui diversas novas funcionalidades no processo de codificação buscando um balanço entre eficiência da codificação, complexidade e custo.

2.2 Codificação de vídeo escalável

A codificação de vídeo escalável, apesar de ser utilizada numa escala muito menor do que a codificação tradicional (*não* escalável), também é tópico de pesquisa há pelo menos duas décadas. Um dos objetivos da codificação escalável é, simplificadamente, permitir que diferentes dispositivos localizados em ambientes diversos tenham acesso a uma mesma transmissão de vídeo. Devido à variedade de ambientes e dispositivos atuais, surgiu a necessidade de que transmissões multimídia possam ser feitas em redes com variadas configurações e capacidades, exibidas por diferentes dispositivos, desde celulares até projetores de alta resolução, e armazenadas em diversos dispositivos, desde cartões de memória reduzida até discos com alta capacidade ou mídias magnéticas como CDs e DVDs. A codificação escalável disponibiliza a mídia em sua maior resolução, mas permite a extração de camadas (*layers* ou *streams*) que possibilitam que esta mídia seja adaptada após já ter sido codificada, tanto antes quanto após a transmissão (em nós intermediários) (OHM, 2005). No padrão H.264 SVC, a escalabilidade é vista como a capacidade de codificar o vídeo em um fluxo de dados que contém um ou mais subfluxos, que podem ser decodificados independentemente e obtidos através do descarte de pacotes do fluxo principal.

A codificação de vídeo escalável utiliza-se da criação de diversas camadas de vídeo para permitir a adaptabilidade, auxiliar no controle de congestionamento e da qualidade de serviço, entre outros. Os vídeos são codificados em n camadas, mas podem ser decodificados utilizando um número menor de camadas do que n. A decodificação com todas as camadas representa o vídeo em sua maior resolução e melhor qualidade. A redução no número de camadas utilizadas possibilita a variação nos parâmetros do vídeo (como resolução espacial e temporal, por exemplo) e (geralmente) reduz a qualidade final obtida.

Apesar de o codificador escalável trabalhar com diversas camadas, normalmente sua saída é apenas um fluxo de bits. Este fluxo, porém, contém informações que permitem que as camadas sejam facilmente extraídas, no processo chamado de adaptação (ou extração) do fluxo de bits (*bitstream adaptation/extraction*). A adaptação no número de camadas utilizadas pode, portanto, ser feita antes da transmissão, onde os vídeos já são transmitidos em camadas separadas (como nos modelos de transmissão que serão comentados na seção 2.3), ou então após a transmissão inicial, onde um nodo intermediário pode receber os dados e adaptá-los para determinados receptores, por exemplo.

Entre as camadas criadas, a primeira é normalmente chamada de camada base, pois seus dados servem como base para todas as outras camadas. Esta camada costuma ser codificada de forma que possa ser decodificada mesmo por um codificador não escalável e é considerada a camada mais importante, pois a reconstrução de todas as outras camadas depende dos dados contidos nela.

As principais técnicas de escalabilidade podem ser divididas em quatro conceitos: (i) escalabilidade temporal, que consiste escalabilidade por variação no número de quadros

por segundo; (ii) espacial, que utiliza a variação na dimensão espacial dos quadros; (iii) de qualidade, a escalabilidade por variação na medida SNR (*Signal-to-Noise Ratio*) dos quadros; e (iv) particionamento de dados, onde os coeficientes dos quadros de vídeo são particionados para formar diferentes camadas.

Padrões mais antigos, como o MPEG-2 e o H.263, já especificavam métodos para se atingir escalabilidade, mas eram raramente utilizados devido, principalmente, à perda de eficiência na codificação (compressão) e ao aumento na complexidade do decodificador (SCHWARZ et al., 2007). A especificação da extensão escalável do H.264 foi finalizada em 2007 com objetivo de fornecer escalabilidade para o H.264 e procurando eliminar os problemas dos padrões antigos. Através de melhorias nas técnicas de escalabilidade e pelo o uso do próprio H.264, o SVC tem melhor desempenho que seus antecessores e fornece vantagens em relação às técnicas de *transcoding*[1] e *simulcast*[2] (duas alternativas à codificação escalável), tornando viável a utilização de codificação escalável.

Na sequência desta seção serão apresentados os três conceitos mais utilizados atualmente para se atingir escalabilidade de vídeo (que são os métodos utilizados neste trabalho) e serão comentadas aquelas utilizadas nos padrões de codificação atuais (principalmente no H.264 SVC, que é utilizado neste trabalho).

2.2.1 Escalabilidade temporal

A escalabilidade temporal (CONKLIN; HEMAMI, 1999; SCHWARZ et al., 2006) é um método de escalabilidade que utiliza a variação do número de quadros por segundo do vídeo (medida que será referenciada por "fps" — *frames per second*). A camada base é codificada com um fps menor do que o fps do vídeo que está sendo codificado e as camadas adicionais possuem o restante dos quadros, que, unidos à camada base, atingem a taxa de quadros por segundo do vídeo original. Vídeos são comumente produzidos a taxas de 25 ou 30 fps. No caso de 30 fps, o vídeo poderia ser codificado com uma camada base contendo 15 fps e uma camada adicional com 15 fps, por exemplo.

Existem diversas técnicas para obtenção de escalabilidade temporal, que estão diretamente relacionadas com o modelo de codificação que está sendo utilizado. Duas categorias importantes são a codificação em sub-bandas e a baseada nas técnicas de estimativa de movimento (CONKLIN; HEMAMI, 1999).

No padrão MPEG-2, é utilizada a técnica baseada em estimativa de movimento (MCP — *Motion Compensated Prediction*), na qual os quadros I, P e B são ordenados e alocados em camadas de forma específica para se obter escalabilidade. A figura 2.3 mostra um exemplo simples de codificação em duas camadas utilizando a variação temporal entre os quadros. No exemplo, são utilizados quadros I e P de referência apenas na camada base, enquanto os quadros B formam a camada adicional. As flechas indicam uma relação entre os quadros: o quadro apontado utiliza o outro como referência para ser codificado. Os números ao lado dos quadros indicam a ordem de envio (e exibição) no caso da transmissão das duas camadas.

A codificação em sub-bandas consiste na separação dos quadros de vídeo em diversas sub-bandas de frequência, assim como é realizado por transformadas como a DCT. Porém, neste caso a decomposição em sub-bandas é feita em 3 dimensões, incluindo a dimensão temporal. Esta técnica é chamada *Temporal Subband Coding* (TSB) (PO-

[1] *Transcoding* é o processo decodificação do vídeo inicialmente transmitido, codificação de forma a adaptá-lo ao receptor alvo e transmissão do novo vídeo criado.

[2] *Simulcast* é a transmissão simultânea de diversos fluxos multimídia (*não* escaláveis), cada um adaptado a um receptor (ou a um grupo de receptores).

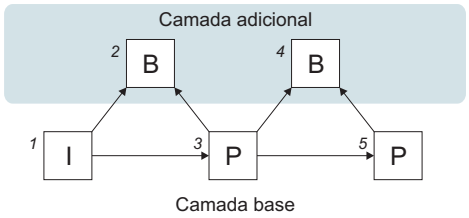


Figura 2.3: Exemplo de codificação escalável temporal utilizando quadros B na camada adicional.

DILCHUK; JAYANT; FARVARDIN, 1995), onde a escalabilidade temporal é obtida pela decodificação das sub-bandas de frequência em um número menor de quadros do que o número total que foi utilizado na geração das sub-bandas. Este processo, porém, produz borramento nos quadros, pois eles são produzidos a partir da combinação linear de um número maior de quadros. Para evitar este borramento, foi proposta uma técnica chamada MC-TSB (*Motion Compensated TSB*) (OHM, 1994), que utiliza compensação de movimento antes da decomposição em sub-bandas.

Na técnica MCP, os quadros B podem ser transmitidos em uma camada adicional, pois não são utilizados como referência por nenhum outro quadro. Padrões mais modernos como o H.264 permitem a utilização dos quadros B como referência para outros quadros. Nestes padrões, uma técnica empregada para obter escalabilidade temporal é o uso de subsequências de quadros (TIAN; GABBOUJ; HANNUKSELA, 2005).

Uma subsequência é uma sequência de quadros que não são utilizados como referência por nenhum quadro que esteja fora desta sequência (quadros da mesma camada ou de camadas inferiores, pois eles podem ser utilizados como referência por quadros que estão em camadas superiores). Dentro de uma subsequência, os quadros podem ser codificados apenas com codificação *intra-frame* ou também *inter-frame*, utilizando uns aos outros ou utilizando quadros externos como referência. Como os quadros desta subsequência não são necessários para decodificação de nenhum outro quadro de fora da subsequência, todos podem ser descartados (ou transmitidos em uma camada adicional) sem afetar a validade dos dados ou o processo de decodificação. A figura 2.4 mostra um exemplo da codificação em duas camadas utilizando o padrão “IpPpP”, onde as letras maiúsculas representam os quadros utilizados como referência e as minúsculas os quadros *não* utilizados como referência. As caixas retangulares representam as subsequências. Nota-se que a subsequência da camada base possui quadros que são referenciados por quadros de fora desta subsequência. Porém, os quadros que fazem as referências estão em uma camada superior, o que é permitido pela técnica. Já as subsequências da camada adicional não utilizam quadros de outras subsequências como referência, apenas quadros da camada inferior ou que estão dentro da subsequência. É interessante observar que, no caso de perda de dados na camada adicional, o erro se propagará somente dentro da subsequência atual.

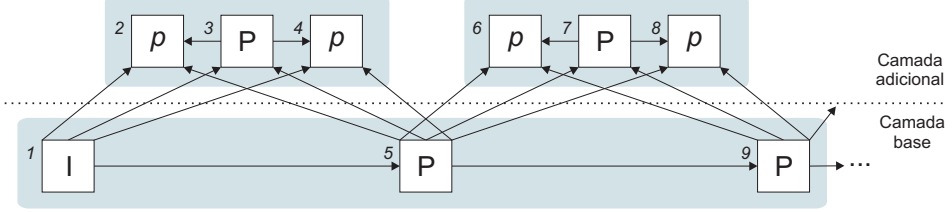


Figura 2.4: Codificação escalável temporal utilizando subsequências de quadros.

As técnicas para escalabilidade temporal possuem algumas outras características importantes que devem ser estudadas antes de sua implementação, além de serem bastante dependentes da quantidade de movimento existente nos vídeos. A redução da taxa de quadros por segundo aumenta o tempo de reprodução de cada quadro, aumentando assim o efeito de *flickering* (efeito que permite a percepção de mudança de um quadro para o outro). Além disso, também permite que o usuário tenha mais tempo para perceber degradações existentes em um quadro. As técnicas também devem se preocupar com o formato do vídeo que está sendo codificado, que pode ser progressivo ou entrelaçado.

2.2.2 Escalabilidade espacial

Escalabilidade espacial (DOMANSKI et al., 2000; SCHWARZ et al., 2007) é conceito de escalabilidade em que os métodos utilizam a variação da resolução dos quadros para criação das camadas de vídeo. Nestas técnicas, o processo de codificação tem como primeiro passo a redução da resolução espacial do quadro de entrada para uma determinada resolução estabelecida. Este quadro é então compactado pelos processos tradicionais de transformada, quantização e codificação entrópica e forma a camada base. Para geração da camada adicional, o quadro base é ampliado para a resolução original e comparado com o quadro original. As diferenças entre eles são então codificadas para formar a camada adicional. É possível a criação de mais de uma camada adicional reduzindo ainda mais o tamanho da camada base e ampliando a resolução à medida que novas camadas são criadas.

A variação na resolução do vídeo é uma característica é interessante em transmissões que são feitas simultaneamente para dispositivos que suportam diferentes resoluções, como celulares e computadores pessoais, por exemplo (VETRO; SUN, 2001). Neste caso, os usuários de PCs recebem a imagem base, fazem a ampliação para a resolução original e então incluem as camadas adicionais para melhorar a qualidade do vídeo. Já os usuários de celular poderiam receber apenas a camada base e não necessitariam ampliar a resolução desta, obtendo assim um vídeo de qualidade adequada com apenas uma camada.

Um processo muito importante na codificação espacial é a redução dos quadros originais para geração das camadas inferiores. Existem algumas técnicas propostas para realizar este processo, entre elas as pirâmides laplacianas (BURT; ADELSON, 1983). Nesta técnica, a imagem é reduzida pelo cálculo da média aritmética dos pixels de cada região. Para reduzir uma imagem pelo fator 2, por exemplo, é definida uma janela de 2x2 pixels, formando assim um grupo de 4 pixels. É calculada a média aritmética desses 4 pixels e este será o valor do pixel de posição correspondente à janela que foi processada. O processo é feito para todas as janelas da imagem (sem sobreposição de janelas), gerando assim um novo nível da pirâmide (nível inferior). O nível inferior é transmitido como a camada base e as camadas adicionais são criadas a partir da diferença entre os níveis da pirâmide. A figura 2.5 exemplifica o processo de criação da pirâmide laplaciana e mostra a criação das camadas. O item (a) da figura exemplifica a criação dos níveis inferiores da pirâmide a partir do quadro original, que são formados pela média dos elementos de cada bloco do nível superior. Os itens (b) e (c) mostram como são calculados os valores que serão transmitidos como camadas adicionais a partir dos valores da pirâmide.

Para decodificação, o primeiro passo é a decodificação da imagem base e ampliação desta para atingir a resolução da camada adicional (segunda camada caso existam mais do que duas camadas). Com o recebimento da camada adicional, ela é decodificada e adicionada à camada base. No caso de existirem mais camadas adicionais, o processo se repete: a imagem atual é ampliada, a camada adicional é decodificada e adicionada à

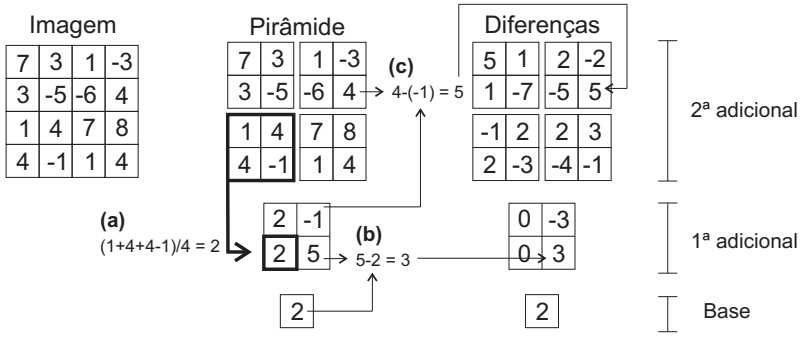


Figura 2.5: Exemplo da criação da pirâmide laplaciana para redução dos quadros.

imagem atual.

As figuras 2.6 e 2.7 mostram um codificador e um decodificador com suporte para n camadas que exemplificam o processo de escalabilidade espacial. Os blocos com setas para cima correspondem ao processo de ampliação da resolução espacial, enquanto as setas para baixo indicam a redução da resolução. Os blocos com o símbolo $\sum$ indicam a união dos quadros, contendo sempre o sinal utilizado em cada entrada, que indicará a operação realizada: dois positivos (+) indicam a soma dos valores e um positivo com um negativo (-) indica a diferença entre eles.

Para todas as camadas, exceto a última, o primeiro passo da codificação é a redução da resolução dos quadros de entrada para a resolução especificada para cada camada. No caso da primeira camada, após a redução, os quadros são codificados (com o processo tradicional) e já formam a camada base. Para formar a segunda camada, o sinal contendo as camadas anteriores (no caso, apenas a primeira camada) é decodificado, ampliado e comparado com o sinal de entrada da segunda. As diferenças entre eles são codificadas para formar os dados transmitidos como segunda camada. Para a terceira e próximas camadas, o processo é idêntico ao da segunda camada, que é exibido na área ressaltada no diagrama.

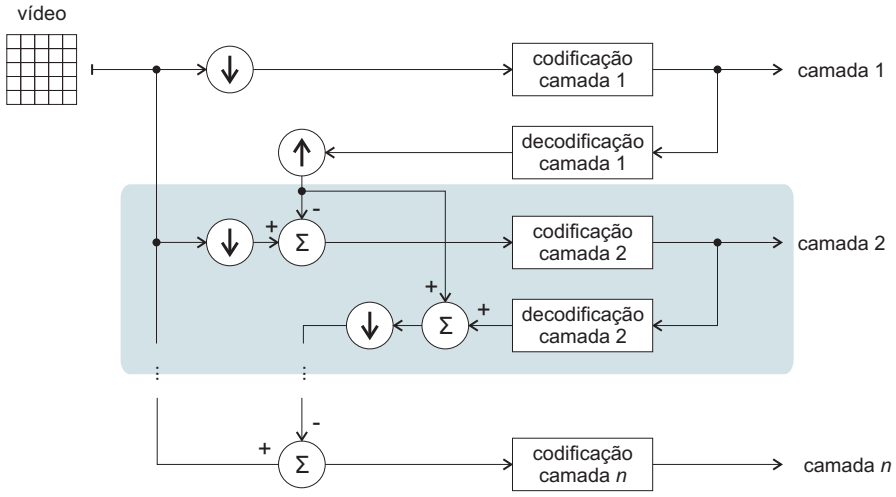


Figura 2.6: Diagrama de um codificador com escalabilidade espacial.

Esta técnica apresentada é empregada no padrão MPEG-2, que também possibilita a integração da escalabilidade SNR com a escalabilidade espacial. O padrão especifica a

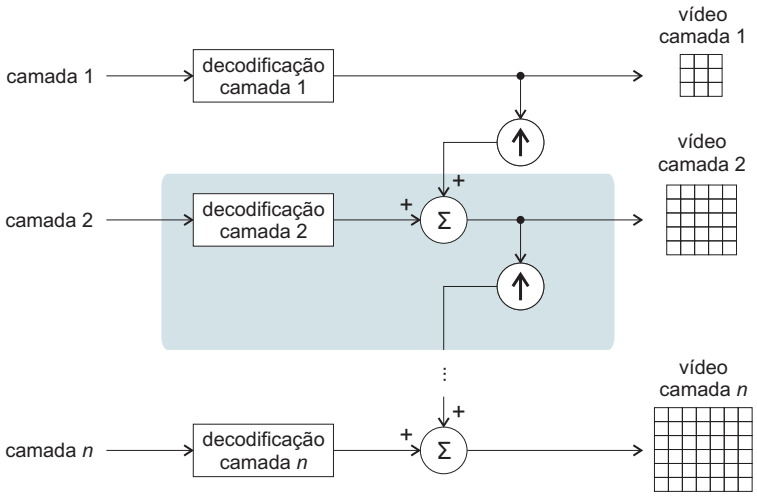


Figura 2.7: Diagrama de um decodificador com escalabilidade espacial.

criação de, no máximo, 3 camadas: uma camada base, uma codificada com escalabilidade espacial e a outra SNR. Também é possível a utilização de apenas 2 camadas, onde a camada adicional pode ser espacial ou SNR. Neste caso, o padrão especifica que um decodificador em conformidade com o perfil de escalabilidade espacial deve ser capaz de decodificar a camada adicional tenha ela sido codificada com escalabilidade espacial ou com escalabilidade SNR.

Além disso, a codificação escalável espacial descrita é semelhante ao processo utilizado pelo modo hierárquico do padrão para codificação de imagens estáticas JPEG (WALLACE, 1991), que codifica a imagem em uma resolução menor e adiciona qualidade codificando as diferenças da imagem reduzida em relação à imagem original.

2.2.3 Escalabilidade de qualidade (SNR)

A escalabilidade de qualidade (ARNOLD et al., 2000; LI, 2001), também chamada de escalabilidade por SNR, é obtida através da variação do nível de fidelidade dos quadros, que normalmente é medido pela relação sinal-ruído desses quadros. Essa variação é geralmente obtida com a mudança no processo de quantização durante a codificação de cada uma das camadas de vídeo.

O processo de codificação em geral é bastante semelhante ao processo de codificação não escalável. Para codificação da camada base, são executados os processos de transformada, quantização e entropia tradicionais, apenas modificando o grau de quantização para geração de uma camada base com SNR inferior ao que teria o vídeo gerado pela codificação não escalável (e, portanto, utilizando um número menor de bits). Esta camada pode então ser transmitida juntamente com outras informações necessárias para a decodificação (como os vetores de movimento).

A camada adicional é gerada com a codificação do restante dos dados que foram descartados pelo processo de quantização da camada base. Após a camada base ser codificada, ela é decodificada e comparada com a imagem original. A diferença entre essas imagens, que consiste na distorção introduzida pela codificação da camada base, é então novamente codificada, agora com maior precisão (nível de quantização mais baixo). Esta distorção pode então ser transmitida para formar uma camada adicional. A figura 2.8 mostra um exemplo da organização dos quadros em uma codificação escalável de qualidade com duas camadas.

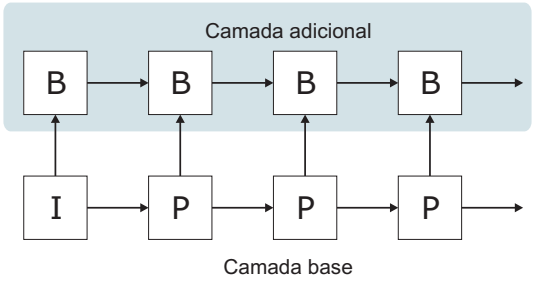


Figura 2.8: Organização dos quadros na codificação escalável de qualidade.

O processo de decodificação para a camada base não apresenta nenhuma mudança em relação à decodificação não escalável. A camada base passa pelos processos de decodificação entrópica, é inversamente quantizada, passa pela transformada inversa e pela compensação de movimento. Quanto à decodificação da camada adicional, são necessárias algumas mudanças. Inicialmente, a camada adicional passa pela decodificação entrópica e quantização inversa. Os valores obtidos são então combinados com os valores recebidos para a camada base e esta combinação é decodificada utilizando o processo padrão da camada base.

Existe a possibilidade de extensão deste método para geração de mais camadas, o que geralmente é feito através de uma redução maior na qualidade da camada base, redução na qualidade da segunda camada e codificação da distorção gerada pela camada adicional (formando a 3ª camada), por exemplo.

O padrão MPEG-2 apresenta um perfil para codificação escalável por SNR (chamado *SNRProfile*) que possibilita a criação de duas camadas de vídeo: uma camada base e uma de refinamento. O padrão especifica que a camada base deve ser codificada de forma que possa ser decodificada por qualquer decodificador que esteja de acordo com o padrão MPEG-2, seja ele um decodificador escalável ou não. O padrão H.264 SVC permite uma maior flexibilidade na codificação escalável de qualidade através de dois modos conhecidos como CGS e MGS, que serão descritos na seção 2.2.5.

2.2.4 Outras técnicas de escalabilidade

Além dos três conceitos de escalabilidade já citados, alguns outros métodos foram desenvolvidos para atingir escalabilidade de vídeo, como é o caso do método por particionamento de dados, o método para atingir escalabilidade fina (FGS — *Fine Grain Scalability*) existente no MPEG-4 (que pode ser considerado um caso de escalabilidade de qualidade) e outros que visam a escalabilidade baseada em regiões de interesse. Estes métodos não possuem conceitos necessariamente diferentes dos três anteriormente abordados. Alguns são apenas casos específicos (como é o caso do FGS) e outros são métodos que podem ser utilizados em conjunto com os métodos comentados nas seções anteriores (como é o caso dos métodos baseados em regiões de interesse).

2.2.4.1 Particionamento de dados

O método de escalabilidade por particionamento de dados atua sobre os coeficientes gerados após a aplicação da transformada, que normalmente é uma das etapas da codificação de vídeo. Este método também é conhecido como escalabilidade por frequência, por atuar no espaço das frequências. A escalabilidade por particionamento de dados é bastante simples, portanto pode ser implementada com pouca complexidade se comparada às

demais técnicas de codificação escalável (BRUNO, 2003).

As abordagens podem variar dependendo do modelo de codificação, mas, em um modelo tradicional, o particionamento de dados é feito após a ordenação em *zig-zag* que é aplicada sobre os coeficientes de cada bloco da imagem. É escolhido um ou mais pontos de particionamento no vetor de coeficientes, pontos que irão dividir esses coeficientes em camadas. Os coeficientes representantes das baixas frequências (valores mais importantes) estarão nas camadas inferiores, enquanto os coeficientes de frequências mais altas fazem parte da(s) camada(s) adicional(is). A figura 2.9 mostra um exemplo do particionamento de um bloco e divisão dos coeficientes em duas camadas.

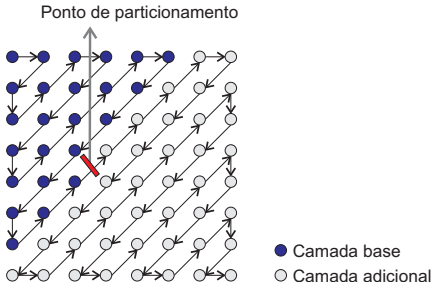


Figura 2.9: Exemplo de escalabilidade por particionamento de dados.

O padrão MPEG-2 define a criação de, no máximo, duas camadas com essa técnica, que normalmente é utilizada juntamente com outras, como escalabilidade espacial e de qualidade. O conceito de particionamento de dados também é utilizado no JPEG progressivo, onde os coeficientes de baixas frequências são os primeiros a serem enviados, o que permite uma reconstrução da imagem com qualidade relativamente boa. À medida que o restante dos coeficientes vão sendo enviados a qualidade da imagem vai melhorando até a reconstrução estar completa (FURHT, 1995).

2.2.4.2 FGS por bit-planes

Outro conceito de escalabilidade é o FGS (*Fine Grain Scalability*), que corresponde à codificação escalável que busca alcançar granularidade fina, ou seja, dispor de diversos níveis de adaptação (diversas camadas). Os métodos de escalabilidade temporal, espacial e de qualidade disponibilizados em padrões até o MPEG-4 possibilitavam a criação de apenas 2 camadas (ou 3, quando combinados). No MPEG-4, uma técnica de codificação por *bit-planes* foi especificada para atingir um número maior de camadas (LI, 2001), podendo ser considerada um caso especial de escalabilidade de qualidade, apesar das diferenças de implementação em relação à escalabilidade de qualidade tradicional.

Durante a padronização do MPEG-4, outras técnicas além da baseada em *bit-planes* foram propostas para atingir FGS: através de codificação wavelet (SCHUSTER, 1998) e uso de *matching pursuits* (MP) (AL-SHAYKH et al., 1999) para codificação dos resíduos da imagem. O método de *bit-planes* foi escolhido devido à sua simplicidade de implementação e eficiência e, portanto, será o método abordado no restante desta seção.

Na técnica de FGS por *bit-planes* do MPEG-4, o vídeo também é codificado em duas camadas: uma camada base e uma camada adicional. A camada adicional, porém, pode ser sub-dividida em diversas outras camadas. Isso é feito através da ordenação dos bits de cada coeficiente de um bloco após a ordenação em *zig-zag*, e por isso a técnica é chamada de *bit-planes*, ou planos de bits. O número de *bit-planes* é definido pelo número de bits necessários para representar o maior coeficiente da sequência.

A distribuição dos bits dos coeficientes em cada *bit-plane* é feita de maneira simples. O *bit-plane* mais significativo é chamado de MSB (*Most Significant Bit-plane*) e irá conter o bit 0, assumindo que o bit 0 de cada coeficiente é o seu primeiro bit, ou seja, o mais significativo. Os demais *bit-planes* são chamados de MSB-n e são formados pelo bit de posição n de cada coeficiente, onde n é o número do *bit-plane*. Assim, o próximo *bit-plane* após o MSB é chamado de MSB-1 e irá conter os bits de posição 1 de todos os coeficientes. Após o MSB-1 virá o MSB-2 com os bits de posição 2 dos coeficientes, e assim por diante. A figura 2.10 mostra um exemplo de uma sequência de coeficiente e os *bit-planes* gerados.

	12	5	0	3	7	0	0	2	2	0	3	0	0	0	0	0	...	0	0	Coeficientes
	↓										...								↓	
bit 0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	...	0	0	MSB
bit 1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	...	0	0	MSB-1
bit 2	0	0	0	1	1	0	0	1	1	0	1	0	0	0	0	0	...	0	0	MSB-2
bit 3	0	1	0	1	1	0	0	0	0	0	1	0	0	0	0	0	...	0	0	MSB-3

Figura 2.10: Exemplo de uma sequência de coeficientes com os *bit-planes* formados.

Após a ordenação dos *bit-planes*, cada um deles é codificação com uma variação do método RLE. A figura 2.11 mostra os valores da codificação RLE para os coeficientes da figura 2.10. Para este modo de codificação, esta técnica apresenta resultados de compressão melhores do que as técnicas RLE seguidas de VLC tradicionalmente utilizadas (LI, 2001).

(0,1)					MSB
(0,0)	(0,0)	(2,1)			MSB-1
(3,0)	(0,0)	(2,0)	(0,0)	(1,1)	MSB-2
(1,0)	(1,0)	(0,0)	(5,1)		MSB-3

Figura 2.11: *Bit-planes* da figura 2.10 codificados em RLE.

A distribuição dos *bit-planes* em camadas pode ser relacionada a diversos fatores e irá depender dos objetivos de cada implementação, ou seja, mais de um *bit-plane* podem ser alocados em uma mesma camada conforme as necessidades.

O desempenho deste método pode ser melhorado com a identificação e tratamento especial de coeficientes prioritários. Após o processo de *zig-zag*, os coeficientes mais importantes (de baixa frequência) estão localizados no início da cadeia de coeficientes. O processo de *bit-plane shifting* consiste em "rotacionar" os coeficientes mais importantes para que eles sejam transmitidos em *bit-planes* mais significativos. Rotacionar um coeficiente significa mover todos seus bits (*shift*) uma posição em direção ao bit mais significativo. O número 6 (0110) rotacionado, por exemplo, passa a valer 12 (1100). A figura 2.12 exemplifica o deslocamento de um bloco da imagem para que ele tenha maior prioridade na codificação com *bit-planes*. No decodificador, o processo de rotação inverso deve ser aplicado sobre os mesmos coeficientes. Além de se atribuir prioridade aos coeficientes mais importantes, pode-se utilizar técnicas mais avançadas que dão prioridade para determinadas regiões da imagem de acordo com sua importância na cena.

Na codificação FGS, como a camada base é a de menor qualidade e ela é a única utilizada como referência na estimativa de movimento, a eficiência da codificação tende a ser menor do que no padrão não escalável. Para amenizar esta perda de qualidade, alguns métodos foram propostos, como é o caso do PFGS (*Progressive FGS*) (WU et al., 2001)

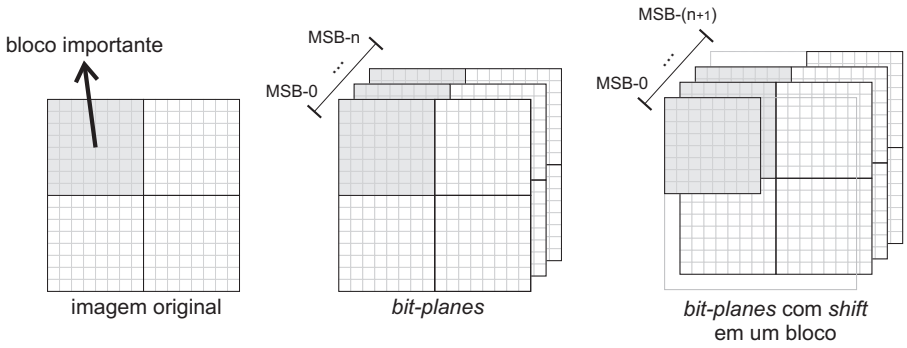


Figura 2.12: Exemplo de uso do *bit-plane shifting*.

e do IPFGS (*Improved Progressive FGS*) (GUO BAO-LONG, 2003). Basicamente, estes métodos sugerem formas de se utilizar de múltiplas camadas de referência para predição de movimento, e não mais apenas a camada base. As camadas adicionais apresentam melhor qualidade que a camada base e, portanto, possibilitam a estimativa de movimento com maior acuidade.

2.2.4.3 Região de interesse

Alguns outros modos de escalabilidade que algumas vezes são necessários são a escalabilidade por região de interesse (ROI — *Region Of Interest*) e escalabilidade baseada em objetos. Em ambos os métodos as camadas adicionais normalmente representam áreas de um quadro, contínuas e que tenham maior relevância, como, por exemplo, as regiões que envolvem o rosto de uma pessoa em uma videoconferência. Um exemplo de implementação da escalabilidade por região de interesse é com o uso de FGS no MPEG-4, onde os *bitplanes* podem ser deslocados para dar maior prioridade às regiões desejadass (LI et al., 2000), como o já citado *bit-plane shifting*.

2.2.5 Escalabilidade no H.264 SVC

A extensão escalável do padrão H.264, chamada de SVC (*Scalable Video Coding*) (SCHWARZ et al., 2007), teve sua primeira especificação finalizada em meados de setembro de 2007. Essa extensão procura unir a eficiência de compressão do H.264 com novas técnicas de codificação escalável, corrigindo algumas deficiências que dificultavam o uso da codificação escalável no passado para tornar o seu uso viável e mais atrativo do que era anteriormente.

Como já comentado anteriormente, padrões anteriores ao H.264, como o MPEG-2 e o H.263, já especificavam métodos para se atingir escalabilidade, mas eram raramente utilizados devido, principalmente, à perda de eficiência na codificação e ao aumento na complexidade do decodificador (SCHWARZ et al., 2007). As principais melhorias do SVC em relação aos padrões anteriores estão justamente na resolução dos problemas existentes no passado: (i) uso de técnicas mais avançadas de predição de movimento entre as camadas de vídeo para melhorar a eficiência da codificação, (ii) redução na complexidade requerida para decodificação e (iii) o próprio uso do H.264, que melhora consideravelmente a eficiência de codificação. O SVC possibilita a utilização dos três conceitos principais de escalabilidade citados (temporal, espacial e de qualidade) e, inclusive, a integração entre eles para codificação de um mesmo vídeo.

Apesar do objetivo final ser o mesmo, o termo “escalabilidade” pode ser interpre-

tado de diferentes maneiras. No SVC, prover escalabilidade é entendido como "permitir a remoção de partes do fluxo de vídeo para adaptá-lo às várias necessidades ou preferências dos usuários finais e às diferentes capacidades dos terminais e/ou condições de rede" (SCHWARZ et al., 2007). Não está no escopo desta seção apresentar informações detalhadas sobre cada etapa da codificação escalável do H.264/SVC, mas sim apresentar as principais características, melhorias e diferenças em relação aos padrões anteriores.

A **escalabilidade temporal** em codificadores híbridos como o H.264/SVC geralmente é feita pela restrição dos quadros que podem ser usados como referência durante a estimativa de movimento para que seja possível deslocar determinados quadros para as camadas adicionais. Quadros de camadas superiores só podem utilizar quadros da mesma camada ou de camadas inferiores como referência. Isso garante que qualquer camada só necessita dela própria e das camadas inferiores para ser decodificada.

A grande vantagem que o SVC oferece na escalabilidade temporal é a flexibilidade de permitir um número arbitrário de referências para codificação de cada quadro, o que era bastante limitado em padrões mais antigos. Esta flexibilidade torna possível a codificação escalável com diversas camadas temporais e também melhora a eficiência da codificação, já que mais referências estão disponíveis para a predição de movimento.

A escalabilidade temporal normalmente é feita no SVC com o uso de quadros B hierárquicos (*hierarquical B-pictures*) (SCHWARZ et al., 2006), onde as camadas adicionais são formadas por quadros B, com a restrição de que eles só utilizam como referência os quadros diretamente posterior ou anterior e que façam parte da camada temporal inferior (estrutura diádica). A figura 2.13 mostra a estrutura básica dos quadros, onde uma seta representa que o quadro apontado utiliza o quadro apontador como referência. O uso de quadros B hierárquicos é um caso especial, mas o padrão não é restrito apenas a este caso. A estrutura de predição também não precisa necessariamente ser diádica e múltiplos quadros podem ser usados como referência.

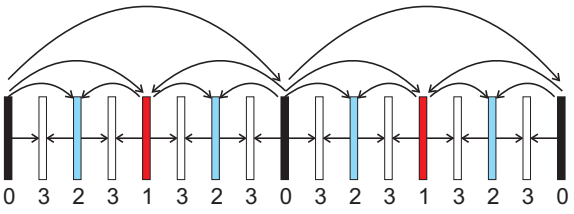


Figura 2.13: Exemplo de estrutura de quadros B hierárquicos.

Outro conceito proposto para codificação temporal é o MCTF (*Motion-Compensated Temporal Filtering*) (SCHäFER et al., 2005), que consiste numa decomposição do vídeo aplicada ao longo do eixo temporal realizada com base na transformada wavelet. A estrutura de codificação é semelhante à estrutura tradicional de um codificador híbrido, sendo que a maior diferença está na inclusão de etapas chamadas de *motion-compensated update*. Simplificadamente, a etapa de *update* consiste em uma segunda aplicação de estimativa de movimento sobre os resíduos da primeira etapa de estimativa (chamada de *prediction*). A etapa *prediction* é considerada a aplicação de um filtro passa-alta, enquanto a etapa *update* é considerada como a aplicação de um filtro passa-baixa. Os quadros codificados pelas etapas de *update* são alocados em níveis temporais inferiores, sendo que quanto menor o nível temporal, mais forte é a filtragem aplicada.

Apesar da maior complexidade, já foi demonstrado que o uso de MCTF não é mais eficiente do que o uso de quadros B hierárquicos (SCHWARZ et al., 2006). Também pode

ser dito que o uso de escalabilidade temporal não tem impacto negativo na eficiência da codificação, apesar de haver uma relação entre a perda de eficiência para manter o atraso baixo ou maximizar a eficiência mas ter a desvantagem de um atraso maior.

Para a **escalabilidade espacial**, o SVC utiliza uma abordagem semelhante à dos padrões antigos. Cada camada representa uma resolução espacial e, em cada uma, predição de movimento com os quadros vizinhos (*inter-frame*) e predição *intra-frame* são aplicadas, assim como na codificação não escalável. Além disso, uma predição adicional é feita *entre* as camadas, a chamada predição *inter-layer*. A predição *inter-layer* permite que as camadas inferiores sejam utilizadas como referência para predição das camadas superiores, ou seja, um quadro de uma camada inferior pode ser utilizado para predição deste mesmo quadro em uma camada superior. A predição pode ser feita para os vetores de movimento, para os blocos em si ou para os resíduos de blocos já codificados (quando o bloco referenciado contém apenas os resíduos de outro bloco).

O SVC permite o uso de resoluções arbitrárias entre as camadas, com a única restrição de que tanto a resolução horizontal quanto a resolução vertical devem aumentar (ou permanecer iguais) de uma camada inferior para uma superior. Além disso, também é possível que uma camada adicional represente apenas uma região da camada inferior, assim como no conceito de escalabilidade por região de interesse (ROI).

Simulações mostram que o uso do esquema completo de codificação espacial do SVC (com os três tipo de codificação *inter-layer*) normalmente atinge resultados melhores do que o uso de opções mais limitadas, que comparam-se aos padrões antigos, como o MPEG-2 e H.263, por exemplo (SCHWARZ et al., 2007).

A **codificação escalável de qualidade** no SVC pode ser considerada um caso especial da codificação espacial no conceito chamado CGS (*Coarse-Garin quality Scalability*). O mesmo processo de predição *inter-layer* pode ser utilizado, com a diferença de que a resolução espacial de todas camadas é a mesma, portanto não é necessária nenhuma etapa de redimensionamento como é feito na codificação espacial.

O CGS, porém, possibilita a criação de um número limitado de camadas. Por este motivo, o SVC também dispõe de outro método de escalabilidade de qualidade, que é chamado MGS (*Medium-Grain quality Scalability*). O MGS é alcançado de duas formas: (i) com a distribuição dos coeficientes obtidos após a transformada em diferentes *slices* ("fatias" de um quadro) e (ii) com a adaptação à nível de pacote, onde uma sinalização inserida pelo codificador permite que pacotes (mais precisamente, NALs — *Network Abstraction Layer*) das camadas adicionais sejam descartados para adaptar o fluxo de vídeo às condições impostas. Com este nível mais fino de adaptação, o MGS permite a criação de um número maior de camadas do que o CGS.

Com o uso de *key pictures* (quadros-chave) e algumas limitações impostas, o SVC consegue reduzir o *drift* relacionado à estimativa de movimento, que sempre representou um problema na codificação. Simplificadamente, o *drift* é um problema que ocorre devido às diferenças que podem existir entre a execução da estimativa de movimento no codificador e no decodificador (devido ao descarte de pacotes da camada adicional após a codificação, por exemplo). Porém, este controle acaba reduzindo a eficiência da codificação. A relação entre eficiência de codificação e *drift* pode ser controlada através da escolha do GOP ou do número de estágios hierárquicos, ou seja, o controle da quantidade e frequência das *key pictures* (que são consideradas como pontos de resincronização).

Outra característica importante da escalabilidade de qualidade do SVC é a possibilidade de inclusão de prioridades para as NALs, o que auxilia o processo de adaptação do fluxo. Em um fluxo de vídeo escalável com MGS, existem diversas possibilidades de

adaptação, ou seja, a adaptação para determinada taxa pode ser feita descartando NALs de diferentes maneiras. Um mecanismo de atribuição de prioridades auxilia este processo de adaptação, fazendo com que dados menos importantes sejam descartados antes com o objetivo de obter melhor qualidade após a extração.

Além do CGS e do MGS, o SVC inicialmente também possibilitava o uso de FGS, de forma semelhante ao FGS do MPEG-4. O FGS permite que os *slices* (basicamente, um grupo de coeficientes) do vídeo sejam truncados em diversos pontos. Esta adaptação à nível de bit oferece ainda mais flexibilidade que a adaptação de NALs feita no MGS, motivo pelo qual o FGS possibilita uma granularidade mais fina. Em razão da alta complexidade computacional do FGS, ele acabou sendo removido da especificação atual do SVC, mas análises dos métodos mostraram que o MGS consegue atingir qualidade bastante similar, mas com uma complexidade menor (SCHWARZ; WIEGAND, 2007).

O SVC também permite a integração dos três métodos de escalabilidade, ou seja, um fluxo de vídeo pode oferecer, ao mesmo tempo, escalabilidade temporal, espacial e de qualidade (seja CGS ou MGS). Isso garante ainda mais flexibilidade e maior poder de adaptapação aos vídeos.

2.3 Transmissão multimídia

A transmissão de dados multimídia pode ser feita de diversas maneiras, assim como a transmissão de dados de forma geral em redes de computadores. O foco deste trabalho está nas transmissões em redes *best-effort* (onde não há garantia de qualidade de serviço e normalmente vários fluxos disputam a banda disponível), fazendo uso de multicast, múltiplas camadas de vídeo e visando ambientes de larga escala. Um ambiente típico para uso deste tipo de transmissão é quando há necessidade de realização de uma transmissão ao vivo para diversos tipos de receptores (computadores pessoais e celulares, por exemplo) que naturalmente terão diferentes capacidades de processamento e transmissão. Os resultados apresentados no capítulo 4 não são válidos apenas para estes ambientes, mas este modelo de transmissão justifica bem a definição dos objetivos deste trabalho e portanto será abordado nesta seção.

O uso de multicast tem grandes vantagens em relação às transmissões unicast, especialmente em transmissões multimídia onde o volume de dados a serem transmitidos é grande. No unicast, a necessidade de transmissão de um fluxo independente para cada receptor exige muitos recursos, tanto de rede quanto de processamento. A principal vantagem do multicast está justamente na economia desses recursos, já que é necessário apenas um fluxo de transmissão, o que também aumenta a escalabilidade da transmissão. Apesar das vantagens, o uso de multicast cria novos desafios, sendo que o principal está relacionado à falta de um canal de retorno (do receptor para o transmissor), que é comum em transmissões unicast.

O multicast também pode ser relacionado de forma bastante interessante com a codificação de vídeo escalável. As camadas de vídeo codificadas são transmitidas cada uma em um diferente grupo multicast, o que permite que os receptores avaliem sua capacidade (rede e processamento, principalmente) e recebam apenas o número de camadas possíveis. Assim, um receptor utilizando uma rede com velocidade de 150 kbit/s receberia apenas 2 camadas (2 grupos multicast), enquanto outro receptor com rede de 2 Mbit/s receberia 4 camadas, por exemplo. Este é apenas um exemplo, mas o número de camadas para cada receptor depende, obviamente, da taxa de codificação utilizada em cada camada.

Um desafio das transmissões multimídia é possibilitar o acesso universal aos fluxos

que estão sendo transmitidos, mesmo que para receptores heterogêneos localizados em ambientes que normalmente também são muito diferentes. Para garantir que todos os usuários recebam a melhor transmissão possível, diversos fatores devem ser considerados, como a topologia da rede, o nível atual de carga da rede e as características de cada receptor. Além disso, outro fator importante é que o tráfego multimídia desta transmissão utilize uma parcela equitativa da banda, sendo imparcial e amigável com os demais tráfegos concorrentes (WIDMER et al., 2001).

Estas etapas de controle da transmissão para verificação dos fatores citados são feitas por algoritmos chamados de protocolos de controle de congestionamento (WIDMER et al., 2001; LIU et al., 2003). As pesquisas dividem estes protocolos em duas categorias: taxa única e multi-taxa. Em **protocolos de taxa única**, como, por exemplo, o PGMCC (RIZZO, 2000) e o TFMCC (WIDMER; HANDLEY, 2001), o transmissor envia os dados para todos os receptores a uma taxa ajustada dinâmicamente, baseando-se no receptor mais lento. Apesar dos pontos positivos, este método faz com que os receptores mais rápidos fiquem limitados à velocidade do receptor mais lento, desperdiçando largura de banda.

Os **protocolos multi-taxa** permitem que a transmissão seja feita em mais de uma taxa de transmissão simultaneamente. A idéia central consiste, como comentado anteriormente, na transmissão do sinal multimídia em diferentes camadas, onde cada camada é transmitida em um grupo multicast diferente. Desta forma, os receptores podem aumentar e diminuir suas taxas de recebimento através de operações de *join* e *leave* nos grupos multicast, conforme sua largura de banda disponível no momento. A operação de *join* consiste em inscrever-se em um grupo multicast, que neste caso corresponde a passar a receber os dados de uma camada de vídeo. O *leave* corresponde à operação contrária, quando o receptor finaliza o recebimento de um grupo multicast. Diversos protocolos foram especificados nesta área, incluindo o RLM (MACCANNE et al., 1996), RLC (VICISANO et al., 1998), PLM (LEGOUT; BIERSACK, 2000) e o ALMTF (ROESLER, 2003), que foi proposto no projeto SAM e será comentado na seção 2.5.

Recentemente, uma nova linha de pesquisa nesta área passou a unir conceitos de protocolos de taxa única e conceitos multi-taxa para desenvolver **protocolos híbridos**. Alguns exemplos são o GMCC (LI et al., 2007) e o SMCC (KWON; BYERS, 2003), que fornecem multi-taxas através da utilização de sub-camadas independentes de taxa única, ou seja, o protocolo disponibiliza diversas camadas, onde cada camada pode apresentar variações na sua taxa de transmissão interna.

É importante observar que não existem implementações efetivas destes protocolos disponíveis para uso. Eles normalmente são implementados e validados através do simulador de redes NS-2, mas isso não viabiliza seu uso em transmissão reais.

Diversos outros protocolos possuem propósitos semelhantes (controle sobre o congestionamento da rede), mas nem todos são voltados para transmissões multimídia, como é o caso dos protocolos desenvolvidos para transmissões com multicast confiável, por exemplo. Além disso, muitos protocolos são de difícil aplicação prática, como é o caso dos protocolos híbridos. Eles necessitam de um codificador que permita a criação de diversas camadas e permita a variação das taxas dentro destas camadas. O SVC é o primeiro padrão a fornecer esta flexibilidade, mas a tarefa de acoplar a codificação com a transmissão e maximizar o uso da banda para os receptores pode ser bastante complexa. Protocolos multi-taxa com camadas fixas facilitam esta integração por não necessitarem variações dentro de cada camada. Desta forma, os vídeos podem ser codificados em um número determinado de camadas e cada uma delas transmitida em um grupo multicast diferente.

2.4 Avaliação de qualidade de vídeo

Para os usuários de um sistema que envolve dados multimídia, a qualidade dos dados é um dos fatores fundamentais para definição de sua satisfação em relação a este sistema. Neste trabalho a qualidade é avaliada em relação aos vídeos, sem considerar os outros componentes que podem existir em um dado multimídia (como o som, principalmente).

A qualidade de um vídeo é influenciada por diversos fatores. Medidas tradicionais como PSNR (*Peak Signal-to-Noise Ratio*) e RMSE (*Root Mean Square of the Error*) fornecem uma estimativa da qualidade de cada imagem do vídeo usando apenas a diferença da luminância dos pixels em relação ao vídeo original. Porém, diversos outros fatores também influenciam a qualidade, como, por exemplo, a taxa de quadros por segundo, a resolução espacial e erros na transmissão que podem levar à geração de artefatos, perda de quadros, entre outros. Diversos estudos já estudaram a validade dessas técnicas para verificação de qualidade de imagens e vídeos (HUYNH-THU; GHANBARI, 2008; WANG; BOVIK; LU, 2002; GIROD, 1993).

O processo de avaliação da qualidade pode ser feito de forma objetiva ou subjetiva. Medidas objetivas são obtidas através de sistemas computadorizados, que analisam as entradas e calculam os resultados, ao contrário das subjetivas, que são obtidas através de experimentos com seres humanos. Algumas técnicas objetivas são desenvolvidas com base na percepção humana, o que melhora a estimativa de qualidade em relação às outras medidas puramente físicas ou técnicas. Os resultados da aplicação de técnicas objetivas são obtidos de maneira muito mais simples, enquanto a aplicação de metodologias subjetivas requer muito mais tempo, esforço e investimento. Porém, são as avaliações subjetivas que geralmente apresentam resultados mais confiáveis e precisos (KOZAMERNIK et al., 2005) se bem aplicadas.

Como comentado, algumas técnicas objetivas baseiam-se na percepção humana. Elas são chamadas de técnicas perceptuais, e são desenvolvidas em função da baixa associação verificada entre os resultados de técnicas objetivas simples e a qualidade realmente percebida pelos seres humanos. Atualmente, o grupo VQEG (*Video Quality Experts Group*), parte integrante do ITU-T, é um importante grupo de trabalho cuja pesquisa é direcionada para a análise de qualidade em dados multimídia. Um dos focos do grupo é na análise de modelos de avaliação objetiva perceptual a fim de definir métodos que possam representar fielmente uma análise subjetiva (VQEG, 2000, 2003).

A avaliação subjetiva de qualidade em dados multimídia é guiada por padrões internacionais definidos por organizações como a ITU. As normas possuem áreas de atuação específicas (televisão a cabo, broadcast, aplicações multimídia, etc.) e recomendam como devem ser realizadas as diversas etapas da análise, incluindo a configuração do ambiente, seleção dos avaliadores, metodologia de testes, entre outros. Para análise de vídeo em aplicações multimídia, a norma que melhor se aplica é a ITU-T Rec. P.910 (ITU-T, 1999). Ela corresponde a uma atualização da ITU-R Rec. BT.500 (ITU-R, 2002), que também é muito utilizada mas é direcionada para sistemas de televisão (BARONCINI, 2006).

Estas normas definem várias etapas do processo de avaliação subjetiva e apresentam diferentes metodologias possíveis para aplicação das avaliações. Essas metodologias diferem-se em relação a diversos fatores, como, por exemplo, à exibição de apenas um estímulo (*single stimulus*) ou dois estímulos (*double stimulus*) aos avaliadores e quanto à presença ou não de uma referência (vídeo original, antes de ser codificado). Neste trabalho foi utilizada a metodologia ACR (*Adjectival Categorical Rating*), que será comentada em mais detalhes na seção 2.4.1.

A seção 2.4.1 a seguir apresenta algumas metodologias de avaliação subjetiva que foram importantes para auxílio nas decisões durante o desenvolvimento do trabalho, dando destaque para as etapas mais críticas do processo. Já a seção 2.4.2 aborda os métodos objetivos, incluindo os perceptuais e os não perceptuais e comentando as suas características mais importantes.

2.4.1 Metodologias subjetivas

Como já comentado, a avaliação subjetiva de qualidade em dados multimídia é guiada por normas internacionais, que recomendam como devem ser realizadas as etapas do processo de avaliação. A tabela 2.1 apresenta algumas normas importantes em relação à avaliação de qualidade de dados multimídia (não apenas vídeo) com uma breve descrição de seu conteúdo. Como o foco deste trabalho está no vídeo, o restante desta seção trata do processo de avaliação subjetiva de qualidade apenas em vídeo.

Tabela 2.1: Normas para avaliação de qualidade.

Nome	Descrição
ITU-R Rec. BT.500	Metodologias para avaliação subjetiva da qualidade de vídeo em televisores.
ITU-T Rec. P.910	Métodos para avaliação subjetiva de vídeos em aplicações multimídia.
ITU-T Rec. P911	Métodos para avaliação subjetiva de dados audiovisuais em aplicações multimídia.
ITU-T J.144	Técnicas para avaliação objetiva de vídeo para televisão a cabo na presença de uma referência (vídeo de referência sem defeitos, erros de transmissão, etc.).
ITU-R BS.1387	Avaliação de sistemas de áudio de alta qualidade.

Para análise de vídeo em aplicações multimídia a norma que melhor se aplica é a P.910 (ITU-T, 1999), que corresponde a uma atualização e adaptação da BT.500 (ITU-R, 2002), norma direcionada para sistemas de televisão. Estas duas são as normas mais utilizadas neste trabalho e serão tratadas pelos nomes P.910 e BT.500 apenas no restante deste documento. Uma das etapas mais importantes do processo é a definição da metodologia utilizada para realização dos testes junto aos usuários. A tabela 2.2 apresenta algumas metodologias descritas nas duas normas citadas e um método mais recente, chamado SAMVIQ (EBU, 2003), que foi criado pela France Telecom R&D e padronizado pelo EBU (*European Broadcasting Union*) e ITU.

O processo de avaliação subjetiva de vídeo funciona, basicamente, como uma sequência de atividades, onde o avaliador visualiza um (ou mais de um) vídeo, interpreta-o e atribui uma nota à qualidade deste vídeo, de acordo com uma escala de valores pré-definida e de acordo com os objetivos das avaliações que devem ser previamente descritos à este avaliador.

Um aspecto importante que diferencia as metodologias é em relação à apresentação ou não de uma referência durante a avaliação. A referência é uma versão do vídeo que está sendo avaliado sem apresentar defeitos, erros ou qualquer degradação que possa estar sendo avaliada. A qualidade dos vídeos degradados será então uma comparação entre sua

Tabela 2.2: Metodologias para execução das avaliações de qualidade de vídeo.

Sigla	Nome	Referência
DSCQS	Double Stimulus Continuous Quality Scale	BT.500
DSIS	Double Stimulus Impairment Scale	BT.500
SSCQE	Single Stimulus Continuous Quality Evaluation	BT.500
ACR	Adjectival Categorical Rating	BT.500, P.910
DCR	Degradation Category Rating	P.910
PC	Pair Comparison	P.910
SAMVIQ	Subjective Assessment Method for Video Quality	EBU

qualidade e a qualidade do vídeo de referência, seja esta comparação feita de forma direta (visualização dos vídeos lado a lado) ou de forma indireta (visualização dos vídeos em momentos diferentes, mas a referência é considerada como a âncora superior, ou a maior qualidade que o vídeo pode alcançar). Uma metodologia com uso de referências precisa ter acesso tanto aos dados degradados quanto aos originais (sem distorções), enquanto metodologias que não usam referências necessitam apenas dos dados degradados. Em relação a este aspecto, as metodologias podem ser divididas em FR (*Full-Reference*) e NR (*No-Reference*). Em FR estão classificadas as metodologias que necessitam do uso de uma referência e em NR aquelas que não necessitam.

Outro aspecto importante das metodologias é em relação à exibição de apenas um estímulo (*single stimulus*) ou dois estímulos (*double stimulus*) aos avaliadores. Exibir apenas um estímulo significa exibir apenas o vídeo que está sendo analisado pelo avaliador no momento, enquanto duplo estímulo significa exibir o vídeo que está sendo avaliado e uma referência deste mesmo vídeo ao mesmo tempo. Estímulo duplo também pode ser utilizado sem envolver uma referência, com intuito de se obter uma comparação entre dois vídeos degradados, por exemplo. Enquanto estudos mostram que metodologias de estímulo duplo chegam a resultados mais precisos (ALLNATT, 1983), alguns trabalhos atuais falham em encontrar diferenças entre elas (HUYNH-THU; GHANBARI, 2005; PINSON; WOLF, 2003).

A escala de votação também é muito importante e difere-se dependendo da metodologia utilizada. A metodologia ACR, por exemplo, utiliza uma escala discreta, onde o número de valores pode variar (5 ou 11, por exemplo). Já a metodologia SAMVIQ utiliza uma escala continua entre 0 e 100. Em escalas discretas os avaliadores estão mais sujeitos a selecionar os valores extremos, o que raramente ocorre em escalas contínuas, onde existe uma variação maior de valores para seleção (CORRIVEAU et al., 1999).

Ainda outro aspecto que tem influência na avaliação de qualidade de vídeo é o número de visualizações de um mesmo vídeo. Comparando as metodologias ACR e SAMVIQ já citadas, na SAMVIQ os vídeos são dispostos em sessões e podem ser visualizados quantas vezes forem necessárias (a decisão é feita por cada avaliador individualmente). Já na ACR não cabe aos avaliadores decidir quantas vezes irão visualizar cada vídeo: a decisão é feita conforme as necessidades da avaliação, restringindo todos os avaliadores à esta definição. Na ACR, inclusive, pode ser feita apenas uma visualização de cada vídeo, o que torna a avaliação mais prática e rápida, mas pode reduzir a confiabilidade.

Em função de todas essas diferenças e para que os avaliadores se adaptem mais facilmente à avaliação, normalmente é executada uma etapa de treinamento antes da execução da avaliação em si. Nesta etapa de treinamento é apresentado o funcionamento da avaliação e normalmente é utilizado um conjunto reduzido de vídeos (preferencialmente vídeos selecionados apenas para a etapa de treinamento) que representam todas as variações que o avaliador irá encontrar durante a avaliação. Com esta etapa de treinamento os avaliadores podem se familiarizar com o processo de avaliação e, especialmente, verificar os extremos de qualidade que serão apresentados, e, portanto, podem distribuir mais facilmente seus votos ao longo de toda a escala de votação. É também durante esta etapa (ou antes dela) que os objetivos da avaliação e as instruções devem ser passadas aos avaliadores, procurando informá-los dos aspectos mais importantes aos quais eles devem ficar atentos durante a avaliação.

Pesquisas na área de avaliação subjetiva de vídeo no domínio televisivo já vêm sendo realizadas há mais tempo do que avaliações no domínio multimídia, um tópico mais recente necessário devido às diferenças existentes entre os domínios (KOZAMERNIK et al., 2005). A maior diferença está na flexibilidade existente nas aplicações multimídia (diversos *codecs*, formatos de imagens, taxas de atualização temporal, etc.) perante a arquitetura mais restrita dos sistemas de televisão (BROTHERTON et al., 2006).

A fim de analisar a qualidade em vídeos escaláveis, os três conceitos de escalabilidade tornam-se extremamente importantes e devem ser considerados como parâmetros fundamentais na avaliação. A avaliação deve levar em conta a dimensão espacial, a dimensão temporal e a qualidade (no caso a qualidade objetiva, ou medida PSNR que normalmente é utilizada) que são alterados pelo processo de codificação. Devido à flexibilidade destes parâmetros, técnicas de avaliação específicas para dados multimídia são mais aplicáveis para avaliação de vídeos escaláveis.

Apesar da existência de normas internacionais para avaliação subjetiva de vídeo, alguns tópicos ainda não estão bem definidos, como a definição de qual metodologia apresenta melhores resultados (BROTHERTON et al., 2006). Duas metodologias importantes atualmente são a ACR (ITU-R, 2002), bastante utilizada pelo grupo VQEG, e a SAMVIQ (EBU, 2003), uma metodologia mais atual que foi criada especificamente para avaliação de vídeo em ambientes multimídia. Uma interessante comparação entre estas metodologias pode ser encontrada no trabalho de Péchard *et al.* (PéCHARD et al., 2008), e elas são brevemente descritas na sequência desta seção.

2.4.1.1 ACR: Adjectival Categorical Rating

A metodologia ACR foi inicialmente apresentada na norma BT.500 mas que também aparece na norma P.910 sob o nome de *Absolute Categorical Rating*. ACR é uma metodologia de estímulo único, onde os vídeos são apresentados um de cada vez e uma nota é atribuída para cada um deles independentemente.

A figura 2.14 mostra o processo de execução da ACR, onde os vídeos ("Vid. n") são apresentados um de cada vez e seguidos por um momento para atribuição do voto ao vídeo visto. A norma P.910 estabelece que este período de votação deve ter até 10 segundos, enquanto o tempo de duração dos vídeos pode variar conforme os objetivos das avaliações. Durante a votação é exibida uma tela cinza aos avaliadores, caso a votação seja feita com material impresso, ou a própria escala de votação, caso os votos sejam atribuídos utilizando o mesmo ambiente que exibe os vídeos (em avaliações multimídia, o aplicativo que exibe os vídeos pode também coletar os votos, por exemplo).

A ordem de apresentação dos vídeos é normalmente feita de forma aleatória para cada

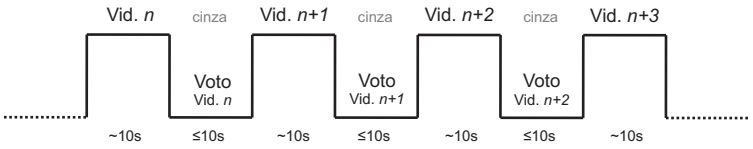


Figura 2.14: Sequência de execução de uma avaliação utilizando ACR.

avaliador, de forma que nenhum avaliador visualize os vídeos na mesma ordem que outro. Este processo ajuda a minimizar a influência do contexto na interpretação da qualidade dos vídeos (PINSON; WOLF, 2003; BARONCINI, 2006). Se, antes de visualizar o vídeo atual, o avaliador analisou um vídeo com muita degradação de qualidade, o vídeo atual provavelmente receberá uma nota superior à que receberia caso o vídeo anterior fosse um vídeo sem (ou com pouca) degradação na qualidade.

A escala de votação tradicional inclui 5 valores, cada um correspondendo à uma categoria, conforme mostra a figura 2.15 (a). Caso seja necessário uma discriminação maior, é possível utilizar uma escala de 9 ou 11 valores, como também mostra a figura 2.15 (b) e (c), respectivamente. A escala é discreta, ou seja, só podem ser escolhidos um dos 5 valores da escala (no caso da primeira escala) e mais nenhum outro valores entre eles. As 5 categorias utilizadas são: ótimo, bom, regular, ruim e péssimo. Elas correspondem, respectivamente, às categorias originais (em inglês): *excellent, good, fair, poor, bad*.

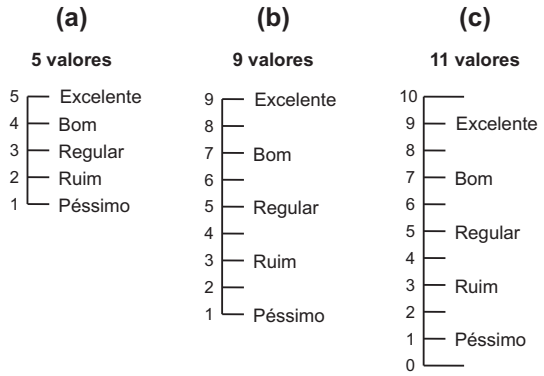


Figura 2.15: Escalas de votação para a metodologia ACR.

Quando se deseja maior confiabilidade, é possível replicar os vídeos avaliados, ou seja, exibir cada vídeo duas ou mais vezes ao longo da avaliação, com o cuidado de não exibir o mesmo vídeo em sequência. Muitas vezes também são incluídos os vídeos de referência no processo de avaliação, mas sem que os avaliadores saibam que estão avaliando uma referência. Este processo é conhecido por HRR (*Hidden Reference Removal*) e a nota atribuída a cada referência pode ser utilizada para normalização dos resultados.

A metodologia ACR é conhecida por ser simples, rápida e mesmo assim ser eficiente e, como já comentado, é bastante utilizada pelo grupo VQEG (VQEG, 2008a).

2.4.1.2 SAMVIQ: Subjective Assessment Method for Video Quality

SAMVIQ (EBU, 2003; KOZAMERNIK et al., 2005) é uma metodologia inicialmente proposta em 2003, portanto é mais recente do que as propostas nas normas BT.500 e P.910. Ela foi designada especialmente para avaliação de vídeo em ambientes multimídia, ao contrário das metodologias propostas nas normas citadas, que eram utilizadas

principalmente para avaliações em televisores.

A SAMVIQ é uma metodologia de estímulos múltiplos, onde os vídeos são organizados em sessões e podem ser visualizados múltiplas vezes, conforme as necessidades de cada avaliador. A ordem de visualização também é decidida por cada avaliador, sendo que a única limitação é que cada vídeo seja visualizado pelo menos uma vez. As sessões são limitadas a 10 vídeos (10 variações de qualidade, ou algoritmos de codificação) e a avaliação pode conter diversas sessões. Dentro de uma sessão, a ordem de visualização dos vídeos pode ser arbitrariamente decidida por cada avaliador, como já comentado. Já a ordem das sessões não é feita pelo avaliador, sendo normalmente é aleatória. Além disso, o avaliador só pode avançar para a sessão seguinte após visualizar e atribuir uma nota para todos os vídeos da sessão atual.

A escala utilizada na SAMVIQ é continua, com valores variando entre 0 e 100 e agrupados em 5 categorias, de forma semelhante à escala da ACR: excelente (80 à 100), bom (60 à 80), regular (40 à 60), ruim (20 à 40) e péssimo (0 à 20). O voto só pode ser atribuído após a primeira visualização completa do vídeo e pode ser modificado à qualquer momento (até que a sessão seja finalizada).

Outra característica importante é que a SAMVIQ utiliza duas referências em cada sessão: uma explícita e uma implícita. A referência é explícita quando os avaliadores sabem que o vídeo que estão visualizando é a referência. A presença de uma referência explícita muda a opinião dos avaliadores sobre os outros vídeos, pois a referência passa a ser utilizada como uma âncora superior e todos os vídeos passam a ser comparados a ela. Já a referência implícita é disposta entre os vídeos sem que o avaliador saiba que esta é a referência e ajuda a avaliar a qualidade real do vídeo de referência. Em avaliações da metodologia SAMVIQ, é comentado que um terço dos avaliadores atribuem notas diferentes às referências implícita e explícita, apesar de estarem avaliando exatamente o mesmo vídeo (KOZAMERNIK et al., 2005).

A figura 2.16 mostra um exemplo simplificado de como é a tela de aplicação da metodologia SAMVIQ. Como a metodologia foi criada para avaliações de vídeo em ambientes multimídia, toda avaliação é executada em um computador por uma aplicação que exibe os vídeos e obtém os votos dos avaliadores. O vídeo é exibido no centro, com os controles para iniciar, parar ou pausar sua execução. Botões na parte inferior permitem a seleção do vídeo de referência ou dos outros vídeos que serão avaliados na sessão (entre eles estará a referência implícita). À direita está a escala contínua, onde é atribuído o voto ao vídeo que está sendo visualizado atualmente.

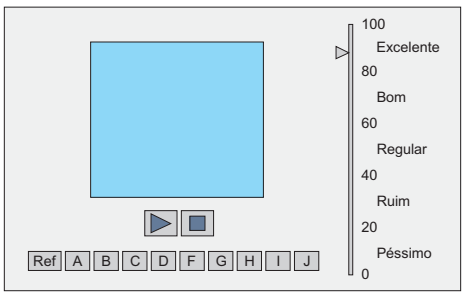


Figura 2.16: Exemplo de uma tela para aplicação da metodologia SAMVIQ.

Esta metodologia tende a ser mais precisa, pois utiliza duas referências e permite a visualização dos vídeos diversas vezes. Por estes mesmos motivos, sua aplicação costuma ser mais lenta se comparada com metodologias de estímulo único, o que força a

redução do número de vídeos utilizados em uma avaliação afim de que ela não se estenda por muito tempo. Por outro lado, novamente pelos mesmos motivos, a SAMVIQ possibilita que resultados sejam encontrados com um número menor de avaliadores do que em metodologias como a ACR, por exemplo (PéCHARD et al., 2008).

As principais diferenças entre SAMVIQ e ACR são: (a) a escala, que é discreta na ACR e contínua na SAMVIQ; (b) os vídeos são vistos apenas uma vez (podem ser repetidos, mas sempre um número pré-definido de vezes) na ACR enquanto na SAMVIQ podem ser visualizados quantas vezes forem necessárias; e (c) a SAMVIQ utiliza uma referência explícita e uma implícita, enquanto a ACR utiliza apenas uma referência implícita (ou nenhuma).

2.4.2 Métodos objetivos

Os métodos de análise de qualidade chamados objetivos são aqueles em que não é necessária a interação humana para visualização e avaliação dos vídeos. O motivo que leva ao desenvolvimento das técnicas objetivas é facilitar o processo de avaliação de qualidade em termos de tempo e custo, principalmente. Uma vez que o método objetivo foi desenvolvido, a aplicação deste método tende a ser muito mais simples que a aplicação de uma metodologia subjetiva, pois, em termos gerais, basta a execução de um aplicativo que tenha acesso aos vídeos degradados e aos originais que o resto do processo é automatizado. Métodos objetivos também podem possibilitar outras aplicações da análise de qualidade, como, por exemplo, a adaptação dinâmica de uma transmissão multimídia de acordo com a qualidade que está sendo obtida pelo receptor.

Técnicas objetivas como o erro médio quadrático (MSE, *Mean Squared Error*, e RMSE, *Root Mean Squared Error*) e a relação sinal-ruído (SNR, *Signal-to-Noise Ratio*, e PSNR, *Peak SNR*) são bastante conhecidas para a análise de qualidade em imagens e vídeos. O MSE e o PSNR são definidos pelas equações (2.1) e (2.2) abaixo, respectivamente.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(x_i - y_i)^2 \tag{2.1}$$

$$PSNR = 10\log_{10}\frac{M^2}{MSE} \tag{2.2}$$

Na equação 2.1, N corresponde ao número total de componentes sendo avaliados na amostra, ou seja, o número total de pixels da imagem. As variáveis x_i e y_i correspondem ao valor do pixel de índice i da imagem x e da imagem y, respectivamente. Na equação 2.2, M representa o valor máximo que um pixel da imagem pode assumir (com pixels de 8 bits, por exemplo, o valor será 255).

Apesar de serem importantes em diversos casos, estas técnicas são extremamente simples e não consideram diversos aspectos que influenciam a qualidade dos vídeos, como a frequência espacial e temporal dos vídeos e variações nas cores, por exemplo. Como já comentado na seção 2.4, diversos trabalhos estudam os usos e limitações dessas técnicas para avaliação de qualidade de imagens (HUYNH-THU; GHANBARI, 2008; WANG; BOVIK; LU, 2002; GIROD, 1993).

Tendo conhecimento dessas limitações, há muitos anos pesquisas vêm sendo desenvolvidas envolvendo aspectos da percepção humana para avaliação de qualidade de imagens e vídeos. O sistema visual humano é bastante complexo e a forma como percebemos cores e movimentos influenciam diretamente a definição da qualidade dos vídeos. Estes

métodos perceptivos têm como base a função espaço-temporal de sensibilidade ao contraste, como mostra a figura 2.17[3].

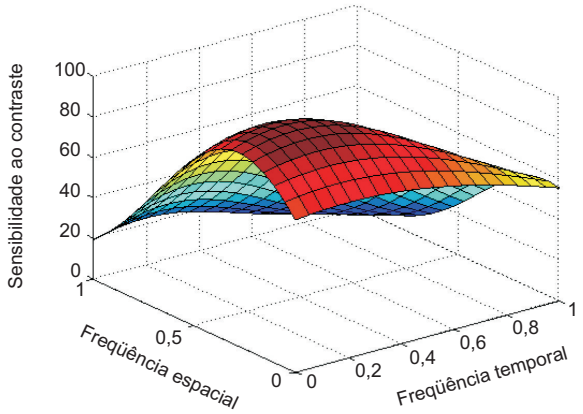


Figura 2.17: Sensibilidade do sistema visual humano.

Esta função é importante principalmente por mostrar a baixa sensibilidade humana às altas frequências temporais e espaciais. Como pode ser visto no gráfico da figura 2.17, os valores mais altos da sensibilidade encontram-se quando há a combinação de baixa frequência temporal e baixa frequência espacial (área escura do gráfico, em vermelho), enquanto os valores mais baixos da sensibilidade estão nas áreas de altas frequências temporais e altas frequências espaciais (áreas mais claras no gráfico, em azul). Este conceito é utilizado pela maioria das técnicas de compressão de imagem e vídeo para degradar as imagens em regiões menos perceptíveis ao olho humano e, portanto, comprimir a imagem reduzindo o mínimo possível a sua qualidade. Diversos outros aspectos podem ser considerados para criação de técnicas mais eficientes, como a existência da visão periférica, a adaptação do olho à luz (fraca ou forte), a possibilidade de acontecer mascaramento (ou facilitação), onde um componente da imagem pode reduzir (ou aumentar) a visibilidade de outro componente, entre outros (WANG; SHEIKH; ALAN C. BOVIK. Objective Video Quality Assessment. In: FURHT B.; MARQUES, 2003).

A figura 2.18 mostra um diagrama das etapas geralmente utilizadas nos métodos perceptuais de avaliação de qualidade de vídeo, onde o sinal degradado é comparado com o sinal de referência utilizando o conhecimento sobre o sistema visual humano. Abaixo as etapas serão descritas brevemente, e maiores detalhes podem ser encontrados nas referências deste trabalho (WANG; SHEIKH; ALAN C. BOVIK. Objective Video Quality Assessment. In: FURHT B.; MARQUES, 2003; PRESTO, 2002).

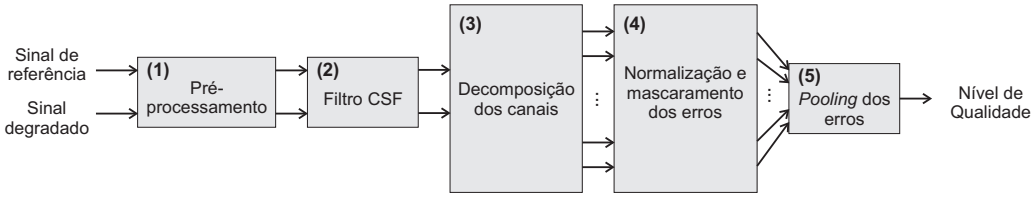


Figura 2.18: Diagrama padrão para métodos perceptuais de avaliação objetiva.

A primeira etapa (1) consiste no pré-processamento das entradas. Esta etapa é formada por diversas tarefas, como o alinhamento temporal dos dois vídeos de entrada, a conversão

[3]Imagem adaptada do documento de Feng Xiao (XIAO, 2000)

de espaço de cores, calibração para os dispositivos de exibição, adaptação à iluminação e filtro PSF (*Point Spread Function* — função que descreve a resposta de um sistema, no caso o sistema visual humano, à uma fonte de luz pontual). Esta etapa prepara os vídeos para que as técnicas perceptuais possam ser aplicadas e eles possam ser comparados.

A segunda etapa (2) consiste na aplicação de filtros lineares que aproximam os dados de acordo com as respostas do olho humano às frequências, o que é feito com o uso da função de sensibilidade ao contraste (CSF — *Contrast Sensitiviy Function*), como ilustrada na figura 2.17.

A decomposição em canais é a terceira etapa (3), utilizada para separar os dados de entrada em diferentes sub-bandas temporais e espaciais. Diversos modelos complexos podem ser utilizados para esta decomposição, porém muitas vezes os modelos que acabam sendo utilizados são mais simples, como uma transformada wavelet ou DCT (*Discrete Cosine Transform*), devido à sua facilidade de implementação e menor custo computacional. Após a decomposição, a próxima etapa (4) calcula o erro entre o sinal de referência e o sinal degradado, levando em conta os valores dos pixels de cada região que está sendo analisada e também de regiões vizinhas, afim de considerar possíveis mascaramentos de regiões.

Como o cálculo dos erros é feito para as sub-bandas separadamente, uma última etapa (5) é necessária para agrupar estes valores em uma interpretação única da qualidade do sinal. Normalmente é utilizada a equação (2.3), chamada de *Minkowski error pooling*. A função utiliza o erro dos coeficientes (e) que foram calculados independentemente em cada sub-banda, ao longo do espaço (índice k) e das frequências (índice l) — ou seja, para todas as sub-bandas —, onde β é uma constante com valor normalmente entre 1 e 4.

$$E = (\sum_{l} \sum_{k} |e_{l,k}|^{\beta})^{\frac{1}{\beta}} \tag{2.3}$$

Além das características já citadas, o uso de métodos perceptuais também inclui na avaliação de qualidade aspectos como os efeitos da transmissão (perdas de pacote, jitter), a fluidez do vídeo ao longo do tempo e degradações conhecidas, como blocagem e borramento dos quadros.

Como já comentado em relação às metodologias subjetivas, os métodos objetivos também podem ser classificados quanto à presença ou não de uma referência, o sinal não distorcido que é comparado ao sinal degradado (WANG; SHEIKH; ALAN C. BOVIK. Objective Video Quality Assessment. In: FURHT B.; MARQUES, 2003). Os métodos chamados de **FR (*Full-Reference*)** assumem que esta referência está disponível na hora da avaliação de qualidade, uma hipótese que é dada como verdadeira pela maioria dos métodos existentes.

Porém, em muitos sistemas práticos não é possível ter acesso a esta referência. Em uma videoconferência tradicional entre duas pessoas, por exemplo, o vídeo que está sendo transmitido por um participante pode ser degradado durante a transmissão, mas o vídeo de referência não estará disponível para o outro participante. Nestes casos, é interessante a existência de métodos que permitam a avaliação de qualidade sem que seja necessário o vídeo de referência. A tarefa de analisar um vídeo degradado sem acesso à referência é bastante complexa, mas existem alguns métodos desenvolvidos que são conhecidos como métodos **NR (*No-Reference*)**.

Além dos métodos FR e NR, há uma terceira classe, chamada de **RR (*Reduced-Reference*)**. Nestes métodos a referência também não está disponível, mas um canal de comunicação adicional é utilizado para transmissão de informações que auxiliam na

avaliação de qualidade. Estas informações adicionais são características extraídas do vídeo de referência e oferecem um custo muito menor de transmissão do que o custo da transmissão de todo o vídeo de referência. Pela presença destas informações adicionais, os métodos RR normalmente são mais eficientes que métodos NR, porém dificilmente possuem desempenho como o dos métodos FR, onde a referência está completamente disponível. Apesar disso, alguns métodos conseguem resultados muito próximos ou até melhores que os dos métodos FR, como é o caso da ferramenta VQM desenvolvida pelo ITS (*Institute for Telecommunication Sciences*). Esta ferramenta obteve cerca de 95% de correlação com os resultados subjetivos em avaliações do grupo VQEG, resultado melhor que o de diversos métodos FR (ITS, 2003).

Em relação a métodos objetivos, também é importante citar o trabalho do já mencionado grupo VQEG. Desde os testes da primeira fase do grupo, completados no ano 2000 (VQEG, 2000), o VQEG busca desenvolver, validar e padronizar métodos objetivos de avaliação de qualidade (principalmente os FR). O grupo executa avaliações objetivas e subjetivas sobre uma mesma base de dados e analisa os resultados para definir os métodos objetivos que representam mais fielmente as avaliações subjetivas. A segunda fase continuou o trabalho da primeira (VQEG, 2003) e, atualmente, o grupo passou a trabalhar em diversos projetos, entre eles o RRNR-TV, que tem foco em métodos RR e NR (VQEG, 2008b).

Além da precisão na estimativa de qualidade, um outro parâmetro importante que deve ser considerado no desenvolvimento de métodos objetivos é a complexidade do método. Alguns extraem características demais da imagem para aumentar a precisão da avaliação, mas o processamento necessário para isso pode inviabilizar a sua utilização em algumas aplicações (avaliações em tempo real, por exemplo). Já para outras aplicações (como a comparação entre *codecs* e avaliações como a deste trabalho), a precisão é o fator mais crítico, portanto é interessante maximizar a precisão mesmo às custas de um aumento na complexidade (e tempo de processamento) dos métodos.

Assim como para as avaliações subjetivas, alguns padrões foram criados para auxiliar a avaliação objetiva de vídeo. Há, por exemplo, a norma norte-americana ANSI T1.801.03 (**?**), que define um conjunto de parâmetros que podem ser usados para medir objetivamente a qualidade dos vídeos. Esta norma foi inicialmente especificada em 1996, mas em 2003 foi atualizada para incluir o método VQM desenvolvido pelo ITS, que, como já comentado, obteve ótimos resultados em avaliações realizadas pelo VQEG. Outras duas normas internacionais importantes nesta área são as recomendações ITU-T J.144 (ITU-T, 2004) e a ITU-R BT.1683 (ITU-R, 2004). Ambas apresentam técnicas para avaliação objetiva perceptual de vídeo, sendo que a primeira é voltada para televisão à cabo e a segunda para broadcast televisivo em *standard definition* (SD).

2.5 Projeto SAM

O trabalho que é desenvolvido dentro do projeto SAM (Sistema Adaptativo Multimídia) (ROESLER, 2003) foi o ponto de partida para as definições deste trabalho. O SAM apresenta um sistema de transmissão multimídia que tem como um dos principais objetivos atingir um grande número de receptores (universalidade da transmissão), mesmo que estes estejam localizados em ambientes heterogêneos e apresentem diferentes características (variações na capacidade de memória, processamento, entre outros). O sistema baseia-se na transmissão de vídeo escalável em camadas cumulativas utilizando IP multicast e mecanismos para adaptabilidade e controle de congestionamento.

A abordagem de transmissão em multi-taxas cumulativas do SAM tem como base a codificação escalável, onde as camadas do vídeo codificado são transmitidas em diferentes grupos multicast. As taxas de transmissão utilizadas em cada camada são fixas e pré-definidas, e as camadas são acessadas sequencialmente, ou seja, as camadas inferiores são pré-requisitos para as camadas superiores. Assim, para receber a 2ª camada, o receptor precisa estar recebendo a 1ª camada; para receber a 3ª camada, precisa estar recebendo a 1ª e a 2ª camadas; e assim por diante. Durante a codificação as camadas também são cumulativas, ou seja, as inferiores são base para as superiores, que adicionam qualidade aos dados das camadas inferiores. Para o receptor adicionar uma camada aos dados que está recebendo é necessário fazer um *join* em um grupo multicast, enquanto para reduzir uma camada é necessário um *leave* (como já comentado na seção 2.3).

A figura 2.19 mostra a arquitetura básica do SAM, onde os dados multimídia são recebidos, codificados em camadas (1) e transmitidos pela rede (2). No lado dos receptores, os fluxos são recebidos de acordo com as condições de cada receptor (3), decodificados (4) e então exibidos ao usuário.

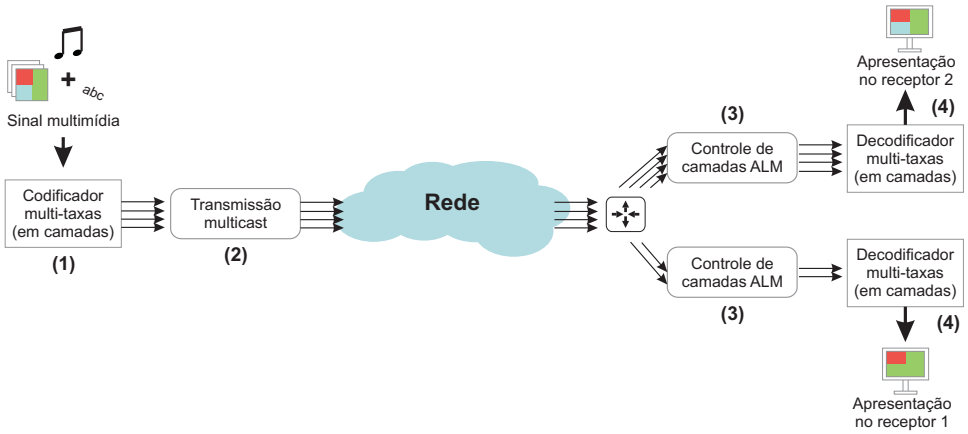


Figura 2.19: Visão geral da arquitetura do SAM.

O projeto SAM pode ser dividido em duas grandes áreas: (i) transmissão multimídia multicast em multi-taxas e (ii) codificação de vídeo escalável. A transmissão multimídia engloba todos os processos que envolvem a transmissão de vídeo e áudio em uma rede de computadores, tendo como foco o desenvolvimento de protocolos de controle de congestionamento (bloco "controle de camadas ALM" na figura 2.19). Para o SAM, o protocolo para controle de congestionamento ALM (*Adaptive Layered Multicast*) foi proposto, assim como o ALMP (*ALM for Private Networks*), uma variação do ALM para uso em redes privativas, e o ALMTF (*ALM TCP-Friendly*), outra variação do ALM designada para uso na Internet.

O ALMTF é o protocolo mais importante entre os citados, e é considerado o núcleo do sistema, um componente fundamental no SAM. Ele segue a abordagem de transmissão baseada no receptor. O controle de congestionamento fim-a-fim pode ser encontrado no receptor ou no transmissor. Quando é inserido no receptor, existe uma menor interação com o transmissor, gerando menos mensagens e, consequentemente, melhorando a escalabilidade do algoritmo. Além do controle de congestionamento, o ALMTF também realiza a adaptação automática dos receptores com relação ao número de camadas, buscando manter a estabilidade e a justiça com tráfegos concorrentes, como o TCP (o protocolo mais utilizado na Internet). Para isso, o ALMTF tenta manter um comportamento

semelhante ao do tráfego TCP, porém, com maior estabilidade. Dentro da metodologia utilizada pelo ALMTF, estão:

- O uso de dois métodos complementares para controle de fluxo no receptor: (i) modelo baseado em janela de congestionamento e (ii) modelo baseado na equação do TCP. O receptor se comunica periodicamente com o transmissor a fim de calcular o RTT, mas o número de mensagens de retorno é controlado por de mecanismos de supressão de *feedback* para permitir que o protocolo seja escalável;

- Premissa de que, quanto mais agressivo for um algoritmo, menos estável fica o tráfego gerado. Portanto, o ALMTF imita o comportamento do TCP de forma menos agressiva, tanto em momentos de aumento de banda quanto em momentos de redução. Isso aumenta a estabilidade e auxilia a manter equidade com tráfegos TCP, apesar de tornar a resposta do algoritmo mais lenta;

- Uso da técnica de par de pacotes para inferir a banda máxima suportada pela rede em que o receptor está localizado;

- Sincronismo entre os receptores inscritos na mesma sessão, para que eles só possam efetuar *join* em determinadas camadas que estão habilitadas no instante de tempo atual;

- Dessincronismo entre os receptores localizados em sessões diferentes, a fim de que eles não efetuem o *join* no mesmo instante.

Os protocolos do SAM foram inicialmente desenvolvidos e validados através do simulador de redes NS-2. Recentemente, foi feita uma implementação inicial do protocolo ALMTF com a intenção de verificar o funcionamento do mesmo em um ambiente real. Resultados satisfatórios foram encontrados (KROB et al., 2007) e o protocolo continua em processo de desenvolvimento dentro do projeto SAM.

Além do ALMTF, diversos outros mecanismos para adaptabilidade e controle de congestionamento podem ser encontrados na literatura, como os já comentados na seção 2.3. Entre as características buscadas por estes mecanismos estão: manter equidade de banda com tráfegos concorrentes, manter a estabilidade durante a transmissão, alta granularidade para melhorar a adaptabilidade, economia (ou bom aproveitamento) de banda e redução na complexidade necessária pra codificação e transmissão.

Em relação à codificação de vídeo escalável, o SAM foi inicialmente proposto para ser utilizado com o codificador em camadas Vebit (Vídeo Escalável por *Bit-planes*). Este codificador foi desenvolvido através de uma dissertação de mestrado (BRUNO, 2003), e atualmente melhorias estão sendo implementadas. A figura 2.20 mostra um exemplo de um vídeo codificado em 5 camadas com o Vebit. Como os módulos de codificação/decodificação estão separados da transmissão (módulo do ALMTF), é possível utilizar qualquer codificador escalável para criar as camadas de vídeo que posteriormente serão transmitidas, portanto o SAM não está limitado ao uso do Vebit apenas.

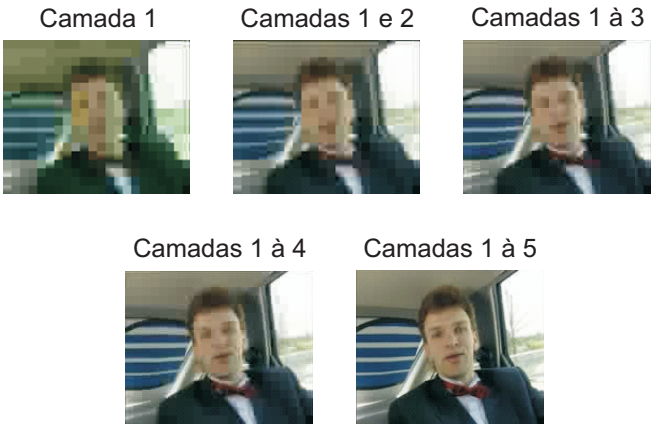


Figura 2.20: Exemplo de vídeo codificado com o Vebit em 5 camadas.

2.6 Trabalhos relacionados

Os trabalhos relacionados comentados neste capítulo envolvem pesquisas na área de codificação de vídeo, transmissão multimídia e, principalmente, avaliação de qualidade de vídeo.

Como já comentado na seção 2.1, a codificação de vídeo é guiada por padrões internacionais que estão em constante evolução ao longo dos anos. Os padrões mais reconhecidos e adotados atualmente são os padrões criados pelo grupo MPEG do ISO/IEC, pelo grupo VCEG do ITU-T e pela associação de ambos no JVT (*Joint Video Team*). O padrão MPEG-4 AVC (WIEGAND et al., 2003), também chamado H.264, é considerado o estado da arte em codificação de vídeo, sendo portanto o foco maior das pesquisas atuais. Entretanto, padrões mais antigos como o MPEG-2 e o H.263, ainda são bastante utilizados em diversas aplicações, como gravação de DVDs e videoconferências.

A codificação de vídeo escalável, apesar de ser usada numa proporção bastante menor do que a codificação tradicional, já é pesquisada há pelo menos duas décadas. Como já comentado, padrões como o MPEG-2, o H.263 e o MPEG-4 já apresentavam métodos para prover escalabilidade, mas estes eram raramente utilizados devido, principalmente, à perda de eficiência na codificação (na compressão) e ao aumento na complexidade do decodificador (SCHWARZ et al., 2007). No ano de 2007 foi finalizada a extensão escalável do H.264, chamada SVC (SCHWARZ et al., 2007). Devido às novas técnicas propostas e ao próprio uso do H.264, o SVC garante melhorias em relação aos padrões antigos, especialmente nos dois aspectos apontados como os mais graves.

Apesar do modelo de codificação híbrido adotado pelos padrões MPEG/VCEG ser o mais difundido e utilizado atualmente, diversos outros modelos já foram propostos, assim como técnicas alternativas para as etapas de codificação do modelo MPEG, como variações na transformada ou na técnica de entropia utilizada. Uma alternativa bastante estudada, especialmente interessante para codificação de imagens, é o uso de transformadas wavelet (GRAPS, 1995). Alguns métodos utilizam algoritmos como o EZW (MARTUCCI et al., 1997) e o SPIHT (KIM et al., 2000), que organizam os coeficientes da transformada de forma hierárquica para aproveitar as redundâncias existentes entre os diversos níveis hierárquicos criados.

Há também uma linha de pesquisa denominada DVC (*Distributed Video Coding*), ou DSC (*Distributed Source Coding*), que baseia-se na compressão de duas ou mais fontes relacionadas mas que não têm interação (ou comunicação) uma com a outra (GIROD et al., 2005). Esta técnica de codificação baseia-se nos teoremas de Slepian e Wolf (SLEPIAN; WOLF, 1973) e de Wyner e Ziv (WYNER; ZIV, 1976), e tem como principal característica alterar a maneira como é feita a codificação entre quadros (redundância temporal) para reduzir a complexidade do codificador (AARON et al., 2002). A redução de complexidade do codificador ocorre pois grande parte da etapa de codificação temporal passa a ser realizada no decodificador, que se torna mais complexo. Esta variação na complexidade das entidades é justificada em aplicações onde o codificador deve ser executado com recursos limitados, como em celulares, por exemplo. Além da codificação tradicional, algumas pesquisas também já investigam o uso de escalabilidade com DVC (XU; XIONG, 2006; OUARET; DUFAUX; EBRAHIMI, 2007).

Em relação à transmissão de dados multimídia, os modelos que mais se relacionam com este trabalho são modelos de transmissão multi-taxas, como o modelo do protocolo RLM (MACCANNE et al., 1996), por exemplo. Estes protocolos utilizam multicast para transmissão dos dados, onde o sinal é dividido em diversas camadas (codificação escalável) e cada uma é transmitida em um diferente grupo multicast. Diversos outros

protocolos possuem modelos semelhantes e são classificados como multi-taxa, como é o caso do RLC (VICISANO et al., 1998), do PLM (LEGOUT; BIERSACK, 2000) e do ALMTF (ROESLER, 2003).

Além dos protocolos multi-taxa, o funcionamento dos protocolos de taxa única e dos protocolos híbridos também é importante para este trabalho. Apesar de as avaliações serem baseadas em transmissões multi-taxa, os resultados também podem ser utilizados para outros ambientes onde é utilizada codificação escalável. Entre os protocolos de taxa única estão o PGMCC (RIZZO, 2000) e o TFMCC (WIDMER; HANDLEY, 2001), e, entre os protocolos híbridos, podem ser citados o GMCC (LI et al., 2007) e o SMCC (KWON; BYERS, 2003).

As características mais importantes destes protocolos e que mais influenciam as avaliações propostas neste trabalho são o comportamento similar ao TCP, que é simulado por alguns protocolos, e a instabilidade existente nas transmissões, apesar dos esforços para reduzi-la (WIDMER et al., 2001). Estas e outras características dos protocolos são importantes por influenciarem diretamente na qualidade do vídeo que será recebido, pois são estes protocolos que decidem o número de camadas (ou a banda) que estará disponível para os receptores à cada instante de tempo. Demais questões sobre transmissão nestes ambientes e protocolos já foram comentadas na seção 2.3.

Outra preocupação é em relação à integração entre a transmissão em múltiplas taxas e a codificação dos vídeos. A transmissão em múltiplas taxas utilizando multicast é especialmente interessante para transmissões feitas ao vivo, necessitando assim de codificadores que trabalhem em tempo real. Esta questão de redução da complexidade do codificador em camadas para utilização em tempo real foi investigada em trabalhos como o de McCanne *et al.* (MCCANNE; VETTERLI; JACOBSON, 1997), na época em que os padrões de codificação que permitiam escalabilidade eram limitados e complexos, como já comentado. Atualmente, o H.264 SVC é o melhor candidato para a codificação devido à sua eficiência e capacidade de prover diversos níveis de adaptação, facilitando a criação de um número maior de camadas e possibilitando diversas configurações da transmissão, mas sua alta complexidade ainda pode dificultar seu uso.

Em relação às avaliações de qualidade, é importante citar os trabalhos realizados pelo grupo VQEG, parte integrante do ITU-T, cuja pesquisa é direcionada para a análise de qualidade de vídeo. Um dos focos do grupo é na análise de modelos de avaliação objetiva a fim de definir aqueles que possam representar fielmente uma análise subjetiva (VQEG, 2008a), visto que a aplicação de métodos objetivos precisos tem a vantagem de ser menos custosa em relação à tempo e esforço.

Como já comentado na seção 2.4, a realização de avaliações subjetivas de qualidade em dados multimídia é geralmente baseada em normas internacionais, que possuem suas áreas de atuação específicas (televisão a cabo, broadcast, aplicações multimídia, etc.) e recomendam como devem ser realizadas as diversas etapas da avaliação, incluindo a configuração do ambiente, validação dos avaliadores, metodologia de testes, entre outros. Para análise de vídeo em aplicações multimídia, a norma que melhor se aplica é a ITU-T Rec. P.910 (ITU-T, 1999), que pode ser considerada uma atualização da ITU-R Rec. BT.500 (ITU-R, 2002), uma das normas mais conhecidas nesta área (BARONCINI, 2006).

Diversos trabalhos utilizam e avaliam as metodologias descritas nessas normas, como o trabalho de Brotherton *et al.* (BROTHERTON et al., 2006), que apresenta uma comparação entre as metodologias ACR e SAMVIQ, além de uma importante revisão sobre avaliação de qualidade, normas e metodologias utilizadas e sobre os trabalhos do grupo VQEG. Importantes aspectos das avaliações de qualidade de vídeo também são descritos

por Baroncini (BARONCINI, 2006), que relata sobre a evolução desta área ao longo dos anos e aponta as dificuldades encontradas e novas direções, especialmente em relação às metodologias utilizadas.

Um trabalho importante que envolve avaliações subjetivas de vídeo consiste nas avaliações do padrão MPEG-4, realizadas pelo próprio MPEG. As primeiras avaliações aconteceram em 1995, e todo o procedimento utilizado é descrito no artigo de Pereira e Alpert (PEREIRA; ALPERT, 1997). Nesta primeira etapa de avaliações, o padrão foi avaliado sob diversos aspectos, como compressão (variadas taxas de codificação), resiliência e recuperação de erros, codificação de objetos, escalabilidade, entre outros. Foram avaliadas diversas ferramentas que implementam o padrão e que foram submetidas para o grupo MPEG e, no mesmo trabalho, são encontrados alguns resultados obtidos nas avaliações.

Diversos trabalhos também podem ser encontrados em relação à análise de performance de padrões de codificação que, mesmo representando uma análise mais técnica, também relacionam-se com a verificação da qualidade desses padrões. Análises de performance do padrão H.264 são encontradas em diversos artigos (OSTERMANN et al., 2004), onde a performance é normalmente considerada como a relação entre taxa de codificação e o PSNR obtido. Avaliações de performance da extensão escalável do H.264, o SVC, também são encontradas (WIEN; SCHWARZ; OELBAUM, 2007), onde a mesma relação entre taxa de codificação e PSNR é descrita para diversas configurações de codificação e comparada aos resultados da codificação não escalável (com o padrão H.264).

A aplicação de avaliações subjetivas em vídeos codificados de forma escalável e, especialmente, codificados com o padrão H.264 SVC, pode ser encontrada em alguns trabalhos atuais. Monteiro e Nunes (MONTEIRO; NUNES, 2007) utilizam avaliações subjetivas de vídeos codificados com o H.264 SVC para construir uma ferramenta capaz de estimar a qualidade de vídeos escaláveis (e também não escaláveis), cuja principal diferença em relação às outras técnicas objetivas é considerar a resolução espacial e a resolução temporal dos vídeos, além das medidas PSNR e RMSE que usualmente já são consideradas. As avaliações deste trabalho utilizaram 4 vídeos codificados na resolução CIF com taxa de 25 fps, que foram avaliados por 21 observadores através da metodologia SAMVIQ. Como resultados, são apresentadas duas técnicas objetivas, uma considerando a resolução espacial e temporal dos vídeos e a outra considerando a medida RMSE. Ainda é apresentada uma técnica combinando ambas, mas que não foi validada através de avaliações subjetivas como as outras.

De forma semelhante ao trabalho de Monteiro e Nunes, Kim *et al.* (KIM et al., 2008) tentam definir uma métrica para avaliação de qualidade de vídeo com suporte à escalabilidade, ou seja, considerando os três parâmetros variáveis na escalabilidade: resolução espacial, temporal e SNR. Os resultados da métrica proposta são comparados com os resultados de avaliações subjetivas realizadas utilizando a metodologia DSCQS (ITU-R, 2002), na qual participaram 18 avaliadores, e apresentam boa correlação (0,93 em média). Alguns pontos que não foram considerados no trabalho são o uso de resoluções maiores do que CIF e o uso de vídeos com conteúdos e características mais variadas (apenas 3 vídeos foram utilizados), como movimento rápido ou lento.

O trabalho de Hsu e Hefeeda (HSU; HEFEEDA, 2007) tenta encontrar a granularidade ideal e as taxas de transmissão de camadas de vídeo criadas pela escalabilidade de qualidade do H.264 SVC, procurando maximizar a qualidade para determinado grupo de receptores. Inicialmente, o trabalho apresenta a formulação de um problema de otimização para definir a granularidade e as taxas pra cada camada de vídeo que maximizem uma função especificada. Foram utilizadas três funções: taxa efetiva recebida pelos clientes,

erro entre a banda do cliente e a taxa dos dados que ele está recebendo e qualidade do vídeo recebido (PSNR). Posteriormente, é apresentado um algoritmo para resolução do problema formulado. A verificação deste algoritmo proposto é feita em comparação ao modelo de alocação exponencial das camadas e apresenta melhores resultados tanto na taxa efetiva alocada quanto na qualidade, medida através do PSNR.

Outro trabalho relevante que utiliza o H.264 SVC investiga a redução no número de quadros por segundo de um vídeo para permitir o aumento da qualidade (PSNR) dos quadros restantes, ou seja, a troca entre as escalabilidades temporal e de qualidade (BARZILAY; TAAL; LAGENDIJK, 2007). O trabalho utiliza a ferramenta VQM para análise da qualidade dos vídeos, que são codificados com diferentes taxas de quadros por segundo através do uso da técnica de quadros B hierárquicos existente no padrão H.264 SVC. As taxas de codificação dos vídeos são fixas, portanto os vídeos com menor número de quadros por segundo apresentam maiores valores de PSNR. Os resultados mostram que não há ganho de qualidade (ou um ganho muito pequeno) em reduzir a taxa de quadros por segundo quando utilizada uma taxa de codificação fixa.

Há também trabalhos que, apesar de não trabalharem diretamente com vídeos escaláveis, examinam a relação entre taxa de quantização e taxa de quadros por segundo, parâmetros principais da codificação escalável de qualidade e temporal, respectivamente. O trabalho de McCarthy *et al.* (MCCARTHY; SASSE; MIRAS, 2004) propõe uma nova metodologia para avaliação de vídeos e a utiliza para examinar se uma alta taxa de quadros por segundo é realmente mais importante que a quantização. Nas avaliações realizadas, os autores verificaram que os avaliadores preferiram a alta resolução e não uma alta taxa de quadros por segundo. Outro resultado importante é que os avaliadores preferiram alta resolução mesmo em vídeos com bastante movimento, onde normalmente é priorizada a taxa de quadros por segundo. É importante observar que este último resultado é válido para telas pequenas, devido às resoluções CIF e QCIF utilizadas.

Ainda em relação à vídeos voltados para dispositivos com telas reduzidas e/ou pouca banda de transmissão, há o trabalho de Brun *et al.* (BRUN; HAUSKE; STOCKHAMMER, 2004), que investiga o desempenho do padrão H.264/AVC (não escalável) em mensagens multimídia (MMS - *Multimedia Message Service*). O principal objetivo deste trabalho foi definir o parâmetro de quantização e a taxa de quadros por segundo que garantem maior qualidade neste ambiente que requer baixas taxas de transmissão. O trabalho indica que o uso de um grau de quantização 34 (variável entre 0-51), taxa de 10 quadro por segundo e taxa de codificação inferiores a 64 kbit/s já é o suficiente para garantir boa qualidade no serviço. Além disso é indicado que, para vídeos de esportes (que apresentam bastante movimento), seja utilizada uma taxa de quadros por segundo maior (15 *fps*).

A maior diferença das avaliações realizadas neste trabalho em relação aos trabalhos citados está na avaliação das variações de qualidade que podem ocorrer durante a transmissão de vídeos escaláveis, o que neste trabalho é chamado de instabilidade, algo que não foi encontrado em outros trabalhos. Mais detalhes sobre estas variações são citadas na seção 3.1, juntamente com os outros objetivos que diferenciam este trabalho dos citados nesta seção.

3 DESENVOLVIMENTO DO TRABALHO

O processo de desenvolvimento do trabalho é formado por todas as tarefas realizadas para definição, preparação e execução das avaliações subjetivas de qualidade. Toda a metodologia utilizada será detalhada nas seguintes seções deste capítulo, que estão organizadas e podem ser resumidas da seguinte forma:

1. **Objetivos e contextualização:** Inicialmente, foram estabelecidas cinco propostas de avaliações de qualidade envolvendo codificação de vídeo escalável, das quais foram extraídos os objetivos deste trabalho. As propostas foram elaboradas com base nos problemas e características existentes nos sistemas de transmissão em camadas, dando prioridade àqueles também encontrados no projeto SAM.

2. **Definição do plano de avaliação:** O plano de avaliação contém a descrição dos objetivos específicos das avaliações e especificação de como eles serão atingidos. Neste trabalho, dois aspectos importantes nas definições das avaliações são as configurações de codificação e os padrões de instabilidade utilizados.

3. **Seleção e processamento dos vídeos:** Descreve o processo de seleção dos vídeos originais utilizados nas avaliações, onde eles foram obtidos e que características foram consideradas para sua seleção. Após obtenção dos vídeos, é descrito o processamento realizado sobre eles antes de sua utilização nas avaliações.

4. **Execução das avaliações subjetivas:** Contém a descrição e preparação do ambiente aonde foram executadas as avaliações, a especificação da metodologia utilizada, seleção dos avaliadores, descrição do processo de execução das avaliações e outras etapas necessárias.

Após a última etapa do desenvolvimento é realizada a análise e apresentação dos resultados, conteúdo que será abordada no capítulo 4.

3.1 Objetivos e contextualização

Inicialmente foram propostos 5 objetivos possíveis para as avaliações de qualidade deste trabalho. Todos eles envolvem a análise de vídeo escalável e são voltados para sistemas de transmissão multi-taxas (mas não limitados a este modelo de transmissão), conceitos descritos nas seções anteriores. A tabela 3.1 mostra um resumo desses objetivos.

Entre as propostas, o objetivo escolhido para execução foi o objetivo "I: Estabilidade vs. Instabilidade". Como será descrito nos resultados, também foram realizadas análises

Tabela 3.1: Objetivos propostos para as avaliações de qualidade.

	Objetivo	Descrição resumida
I	Estabilidade vs. Instabilidade	Estável na camada A ou instável, com variações entre as camadas A e B, sendo que a camada B é diretamente superior à A. A instabilidade utilizando camadas com qualidade superior ou a estabilidade em camadas de qualidade inferior representa melhor qualidade para os usuários? Quais os efeitos dessa instabilidade?
II	Avaliação dos métodos de escalabilidade	Para determinada banda disponível na rede, qual método (ou combinação de métodos) atinge maior qualidade subjetiva? O aumento na taxa de codificação *sempre* representa aumento na qualidade percebida? O comportamento verificado é o mesmo para todos os métodos de escalabilidade?
III	Avaliação dos métodos de escalabilidade com variações nas camadas	Qual método apresenta melhor qualidade quando ocorrem variações durante a exibição do vídeo? Alterações na dimensão espacial, temporal ou na qualidade (PSNR) do vídeo influência mais a qualidade percebida pelos usuários?
IV	Quantidade de camadas	Definição de um número de camadas que satisfaça os usuários. *Poucas* camadas geram variações bruscas, menor flutuação e pior aproveitamento de banda, enquanto *muitas* camadas podem gerar sobrecarga desnecessária e um grande número de variações entre as camadas, mas aproveitar melhor a banda disponível.
III	MGS vs. Escalabilidade espacial	Mantendo-se a mesma dimensão espacial durante a exibição dos vídeos, os dois métodos apresentam variações apenas na qualidade das imagens. Qual apresenta melhor qualidade subjetiva?

envolvendo os métodos de escalabilidade, portanto, em parte, também foi executado o objetivo "II: Avaliação dos métodos de escalabilidade". Além disso, algumas etapas para realização das avaliações do objetivo II também foram concluídas, apesar das avaliações não terem sido finalizadas. Isso será comentado no capítulo **??**. Por ter sido escolhido, o objetivo I será descrito nesta seção. Mais detalhes sobre as propostas para os outros objetivos são encontradas no apêndice A.

A escolha do objetivo que envolve a comparação entre transmissões estáveis e instáveis foi feita devido à existência de instabilidade em diversos ambientes de transmissão multimídia e às suas diferentes formas de manifestação. Variações na banda de rede disponível para o receptor podem ocorrer naturalmente devido à alteração do tráfego desta rede, e podem resultar na variação da qualidade do vídeo recebido pelo usuário. Uma das preocupações dos protocolos de controle de congestionamento é amenizar essa instabilidade, mantendo a transmissão o mais estável possível ao longo do tempo.

Em simulações de diversos algoritmos multi-taxas (ROESLER, 2003; MACCANNE et al., 1996; VICISANO et al., 1998; WIDMER; HANDLEY, 2001), foi visto que quanto maior o número de tráfegos concorrentes (normalmente tráfegos utilizando o mesmo protocolo que está sendo examinado ou o TCP), maior é a instabilidade da transmissão. Além

disso, uma característica buscada pela maioria dos protocolos de controle de congestionamento é manter a equidade de banda com tráfegos concorrentes. Para isso, normalmente acabam sendo utilizadas técnicas que, como consequência, reduzem a estabilidade dos sistemas (LI; LIU, 2003). A figura 3.1 mostra exemplos de simulações de 5 protocolos (ROESLER, 2003), onde são utilizados 4 fluxos do protocolo alvo em cada gráfico. A variação das camadas ao longo da transmissão é percebida pelas subidas e descidas das linhas dos gráficos.

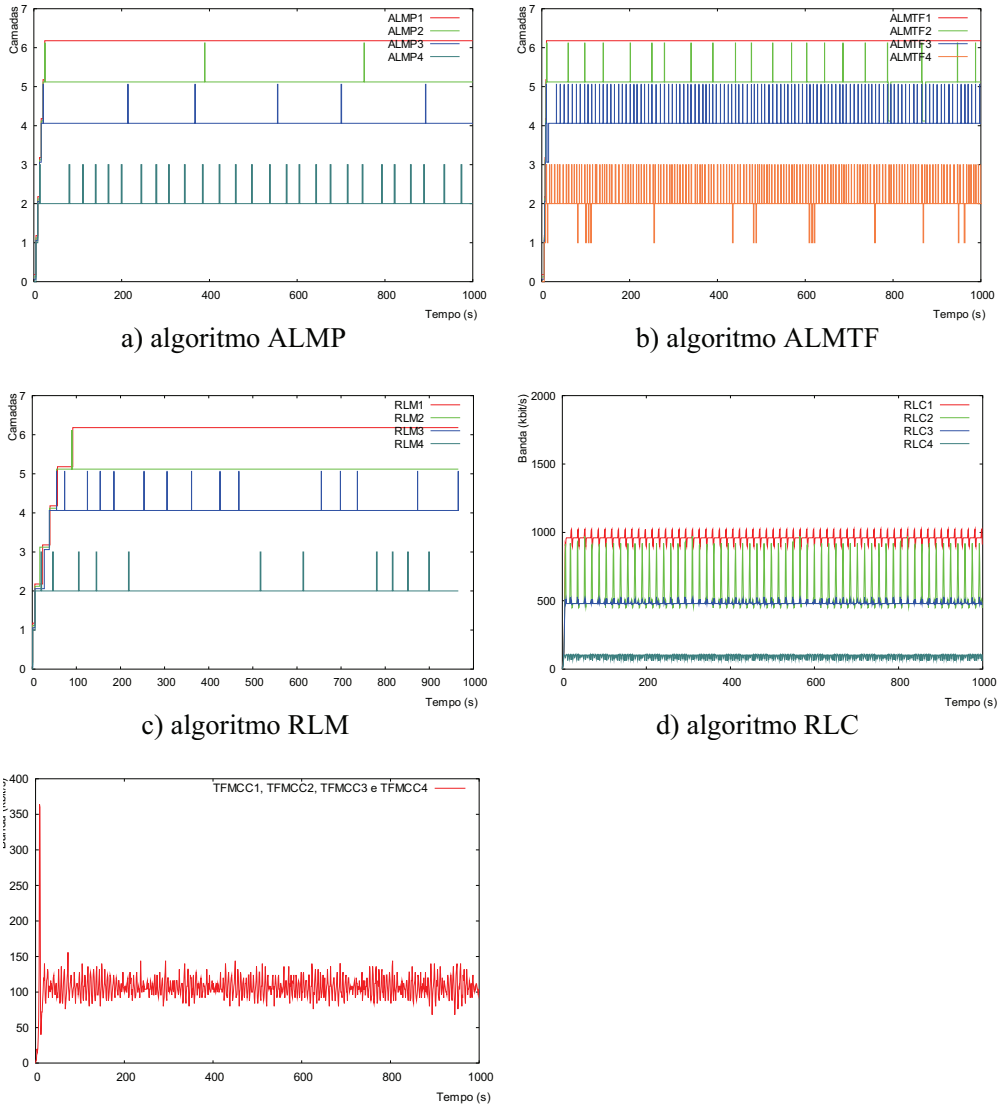


Figura 3.1: Simulação de protocolos de controle de congestionamento exibindo a variação das camadas (e banda) ao longo da transmissão.

Baseando-se nesses fatos, as avaliações foram definidas de forma a simular alterações nas camadas de vídeo conforme ocorreriam em momentos de instabilidade na transmissão e comparar a qualidade obtida nesses casos com a qualidade obtida quando um sistema permanece estável. Na forma de instabilidade utilizada, a transmissão estável permanece uma camada abaixo da instável na maior parte do tempo, ou seja, enquanto a transmissão estável está na camada 2, a instável tem variações entre as camadas 2 e 3, por exemplo.

A comparação entre os métodos de escalabilidade também é importante devido às diferentes características de cada método. A codificação escalável disponibiliza três métodos principais de criação de vídeos escaláveis e possibilita a combinação destes métodos, gerando assim diversas configurações possíveis para as camadas de vídeo. O aumento no nível dos parâmetros de cada método (aumento da resolução espacial de QCIF para CIF, por exemplo) normalmente resulta em aumento da taxa de codificação do vídeo, porém, nem sempre o aumento na banda utilizada representa aumento na qualidade percebida pelos usuários. Identificar qual método de escalabilidade e quais configurações deste método resulta na melhor qualidade subjetiva para diversos valores de banda também representa um dos objetivos das avaliações deste trabalho.

3.2 Definição do plano de avaliação

O plano de avaliação contém as definições das etapas que serão realizadas durante as avaliações, que são utilizadas como guias durante a realização destas etapas. Durante a definição do plano de avaliação, os conceitos descritos nas seções anteriores e os objetivos estabelecidos são utilizados para seleção de aspectos como a metodologia a ser utilizada nas avaliações, a quantidade de vídeos e de variações impostas a eles (processamento), quais serão estas variações, a quantidade esperada de avaliadores, entre outros. A etapa mais extensa e que tem maior impacto no restante do trabalho é a definição das variações impostas aos vídeos, que serão descritos na sequência desta seção.

Neste trabalho foram adotados os termos "SRC", "HRC" e "PVS", que são utilizados pelo grupo VQEG (VQEG, 2008a) e por outros trabalhos da área. HRCs (*Hypothetical Reference Circuits*) são as alterações às quais os vídeos originais, chamados SRCs (*Source Reference Circuit*), são submetidos para geração dos vídeos que serão avaliados. Em nosso caso, os HRCs incluem o processo de codificação escalável e a posterior variação das camadas do vídeo, etapas que serão descritas a seguir. A aplicação de um HRC sobre um SRC gera um PVS (*Processed Video Sequence*), que representa o vídeo final que será apresentado aos avaliadores. A figura 3.2 ilustra o processo em que um SRC é processado por um HRC para gerar uma PVS.

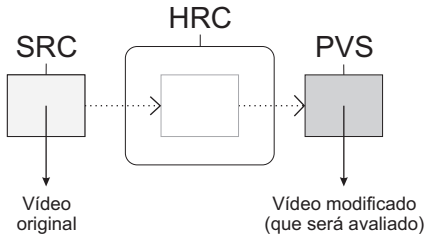


Figura 3.2: SRCs, HRCs e PVSs.

A decisão da quantidade de SRCs e HRCs utilizados tem impacto na duração das avaliações. Um número grande de SRCs é desejável para que seja possível utilizar vídeos com conteúdos variados, já que o conteúdo tem influência direta no processo de codificação. Já uma grande quantidade de HRCs é desejável para que se possa incluir mais variações sobre os vídeos e tornar os resultados mais abrangentes. Porém, o número de SRCs e HRCs deve ser equilibrado para que as avaliações não se estendam por um tempo muito longo.

De acordo com os objetivos buscados, foram especificados 18 HRCs, que serão descritos nas seções 3.2.1 e 3.2.2. Avaliações de qualidade normalmente utilizam vídeos com

duração entre 8 e 10 segundos, mas foi verificado que este tempo seria muito pequeno para nossos objetivos, portanto a duração de cada PVSs foi fixada em 14 segundos. Uma duração mais longa facilita percepção das alterações de camadas que ocorrem durante a exibição de uma PVS. Isto foi verificado após alguns ensaios iniciais, onde 2 vídeos de 10 segundos foram processados por alguns HRCs de teste e foram visualizados, simulando o que aconteceria nas avaliações.

Tendo como base os 18 HRCs e os 14 segundos que cada PVS deveria ter, foi definido o uso de 8 SRCs para as avaliações e mais 3 SRCs para a fase de treinamento. O uso de 8 SRCs é bastante razoável, pois permite a inclusão de diversos conteúdos e variações de complexidade de codificação, como será descrito na seção 3.3. Há trabalhos tais como as avaliações do grupo VQEG que normalmente utilizam um número maior de vídeos (24 vídeos, divididos em 2 grupos de 12 para cada formato escolhido (VQEG, 1999)), enquanto outros chegam a utilizar apenas 4 (NEMETHOVA et al., 2004). Além disso, é comentado na norma ITU-T Rec. P.910 (ITU-T, 1999) que para alcançar resultados confiáveis e evitar entediar os observadores, pelo menos 4 vídeos com conteúdos diferentes devem ser utilizados, ou seja, 8 vídeos representa um número adequado.

O uso de 8 SRCs e 18 HRCs produz 152 PVSs, que é o número total de vídeos avaliados por cada visualizador. Como cada vídeo possui 14 segundos, o tempo total apenas para a exibição de todas PVSs fica em cerca de 35 minutos. Incluindo o tempo para votação, cada avaliação foi prevista para durar cerca de 1 hora, mais um período opcional para intervalo.

A metodologia selecionada para execução das avaliações foi a ACR-HRR (descrita na seção 2.4.1.1), pois possibilita a aplicação de um bom número de PVSs em um período curto de tempo e por ser uma metodologia bastante difundida e utilizada que é capaz de produzir resultados válidos (BROTHERTON et al., 2006). Mais detalhes sobre como ela foi aplicada serão descritos na seção 3.4. Para codificação foi escolhido o padrão escalável H.264 SVC, por ser o estado da arte atualmente e por permitir a criação das configurações de codificação que foram selecionadas. Mais comentários sobre a codificação dos vídeos são feitos na seção 3.3.

As seções 3.2.1 e 3.2.2 a seguir descrevem as configurações de codificação e os padrões de instabilidade que formam os HRCs utilizados no trabalho.

3.2.1 Configurações de codificação

Três configurações de codificação foram criadas para a codificação escalável, cada uma utilizando apenas um dos três métodos de escalabilidade: temporal, espacial e de qualidade. Cada configuração aplicada em um SRC gera 3 camadas de vídeo, que possuem diferenças em apenas um dos seus parâmetros de codificação. Na configuração E (Espacial) apenas a resolução espacial é modificada; na T (Temporal) apenas o número de quadros por segundo; e na Q (Qualidade) apenas o PSNR, como pode ser conferido na tabela 3.2. A tabela apresenta o nome dos três métodos utilizados, os parâmetros que permanecem fixos e os parâmetros que variam em cada método.

Com estas três configurações definidas é possível analisar os três métodos de codificação escalável separadamente, pois em cada uma delas apenas um parâmetro varia, que é o parâmetro que possibilita a escalabilidade conforme o método de escalabilidade utilizado. Foi permitido que apenas um parâmetro sofresse alterações para tentar isolar os efeitos que cada método de escalabilidade tem sobre a qualidade dos vídeos.

Outra questão muito importante é a taxa de codificação (chamada *bitrate*, em inglês, ou simplesmente taxa, como será chamada na maioria das vezes no restante deste traba-

Tabela 3.2: Configurações para a codificação escalável.

Configuração	**Parâmetros**	
	Fixos	**Variáveis**
Espacial (E)	PSNR: 38 dB Quadros por segundo: 30 fps	Resoluções: QCIF (176x144), CIF (352x288) e 4CIF (704x576)
Temporal (T)	Resolução: 4CIF PSNR: 38 dB	Quadros por segundo: 3.75, 7.5 e 30 fps
Qualidade (Q)	Resolução: 4CIF Quadros por segundo: 30 fps	PSNR: De acordo com a taxa de codificação utilizada nas outras duas configurações. Última camada sempre com 38 dB.

lho) de cada uma das camadas de vídeo de cada configuração. Como as camadas serão comparadas entre as diferentes configurações, é importante que a taxa dessas camadas seja o mais similar possível, para isolar ainda mais o único parâmetro variável de cada configuração. Entretanto, notou-se na prática que a taxa das camadas pode sofrer grandes variações conforme a configuração utilizada, e também conforme o vídeo que está sendo codificado (isso será mostrado na seção 3.3). Uma alternativa seria fixar a taxa de cada camada (para todas configurações) em determinado valor, porém os parâmetros que atualmente são variáveis certamente sofreriam alterações conforme o vídeo que está sendo codificado (esses parâmetros são variáveis entre as camadas, mas são os mesmos para os diferentes vídeos). Teríamos, por exemplo, um vídeo A com a primeira camada da configuração Qualidade com PSNR 30 dB enquanto outro vídeo B teria esta mesma camada com PSNR 34 dB, apesar de ambas as camadas possuírem a mesma taxa. Em função de o objetivo principal deste trabalho ser analisar a instabilidade e fazer esta análise para cada um dos conceitos de escalabilidade individualmente, a comparação torna-se mais justa caso os valores das configurações (PSNR, fps e resolução espacial) sejam fixados entre todos os vídeos, e não suas taxas. As taxas de codificação dos vídeos utilizados neste trabalho serão exibidas na seção 3.3.4.

A decisão de se criar configurações com 3 camadas de vídeo deu-se devido, principalmente, a dois fatores:

- O número de HRCs que seriam criados: Quanto mais camadas utilizadas maior será o número de HRCs, ou seja, mais variações poderão ser analisadas, porém mais longas serão as avaliações.

- As resoluções espaciais possíveis: O uso de uma resolução menor do que QCIF foi vetada devido aos impactos que a camada base tem na codificação escalável. Com uma camada base menor do que QCIF, o desempenho da codificação das camadas superiores seria muito comprometido. Já uma resolução maior do que 4CIF não foi utilizada devido a fatores como: (i) o alto processamento requerido para codificação, (ii) as diferenças existentes no processo de avaliação subjetiva de vídeos de alta definição (HD ou próximas a isso) e (iii) a grande diferença que existiria entre a menor resolução (QCIF) e a maior (maior que 4CIF), dificultando a padronização da

exibição dos vídeos. Também seria possível o uso de uma resolução intermediária entre CIF e 4CIF, por exemplo, mas esta resolução também dificultaria a exibição dos vídeos, como será comentado na sequência.

Para a configuração Espacial, foram selecionadas as resoluções QCIF (176x144), CIF (352x288) e 4CIF (704x576) para serem usadas em cada uma das três camadas. Estas resoluções são comumente utilizadas em sistemas multimídia e facilitam a padronização da exibição dos vídeos, pois tanto QCIF quanto CIF podem ser convertidas para 4CIF apenas replicando os pixels de cada quadro, sem necessidade de alguma técnica mais elaborada, como interpolação, por exemplo. Como será comentado posteriormente, todos os vídeos foram exibidos utilizando a mesma resolução espacial (4CIF). Sem utilizar uma técnica mais complexa para redimensionar as imagens, também é reduzido o efeito que este processo pode ter na qualidade dos vídeos e, portanto, nos resultados das avaliações. As camadas de vídeo nesta configuração têm o número de quadros por segundo fixado em 30 fps e o PSNR fixado em 38 dB, pois 30 fps foi o valor máximo escolhido para o número de quadros por segundo e 38 dB foi o valor máximo para o PSNR, como será comentado posteriormente.

Para a configuração Temporal, o parâmetro variável é a taxa de quadros por segundo (fps). Sistemas multimídia normalmente utilizam taxas de 24, 25 ou 30 quadros por segundo (vídeo progressivo), enquanto sistemas televisivos utilizam 50 ou 60 campos por segundo (vídeo entrelaçado). Portanto, 30 fps representa uma boa generalização do que é utilizado pela maioria dos sistemas atuais e, por este motivo, a taxa máxima de quadros por segundo dos vídeos foi fixada em 30 fps. A taxa para a terceira camada é igual a taxa máxima, enquanto a taxa para as duas camadas inferiores são 3,75 fps e 7,5 fps. Essas taxas foram obtidas a partir do valor máximo e através da equação $n*2^x = 30 \mid x = (2, 3)$, onde n é a taxa obtida. Estas taxas são facilmente obtidas no H.264 SVC com o uso de quadros B hierárquicos, uma técnica bem definida e validada que é utilizada para atingir escalabilidade temporal (ver seção 2.2.5).

Assim como na escolha das resoluções espaciais, as resoluções temporais também foram definidas de forma a facilitar a padronização da exibição dos vídeos. Tanto a taxa 7,5 fps quanto a 3,75 fps são facilmente convertidas para a taxa máxima, 30 fps, apenas repetindo os quadros do vídeo 2^x vezes. Novamente, o uso de uma técnica simples para padronizar os vídeos reduz a influência deste processo nos resultados dos experimentos. Os parâmetros fixos para a configuração Temporal são a resolução espacial 4CIF e o PSNR com valor 38 dB.

Como pode ser visto, não foi utilizada a taxa de 15 fps, que naturalmente seria utilizada em uma camada inferior à última camada que possui 30 fps. As taxas das duas primeiras camadas são 3,75 fps e 7,5 fps, e a última camada vai diretamente para 30 fps. Esta decisão foi feita após os ensaios iniciais de codificação de alguns vídeos de teste (que posteriormente foram utilizados nas avaliações), onde foi verificado que as taxas de codificação das camadas da configuração Temporal eram mais próximas às taxas das camadas das configurações Espacial e Qualidade quando utilizadas camadas com taxas temporais 3,75 fps, 7,5 fps e 30 fps, descartando o 15 fps. Buscando justiça na comparação entre as camadas dos diferentes métodos, é importante que a taxa dessas camadas seja o mais similar possível, apesar de dificilmente ser possível mantê-las idênticas, como já foi comentado anteriormente.

A figura 3.3 mostra a diferença na codificação de dois vídeos com o uso e sem o uso de uma camada com 15 fps, onde os gráficos (a1) e (b1), à esquerda, mostram a codificação utilizando 15 fps e os gráficos (a2) e (b2), à direita, mostram a codificação sem o uso de

15 fps. A tabela 3.3 mostra os valores das taxas para cada uma das camadas (C1, C2 e C3), onde fica mais fácil comparar a configuração Temporal com a Espacial.

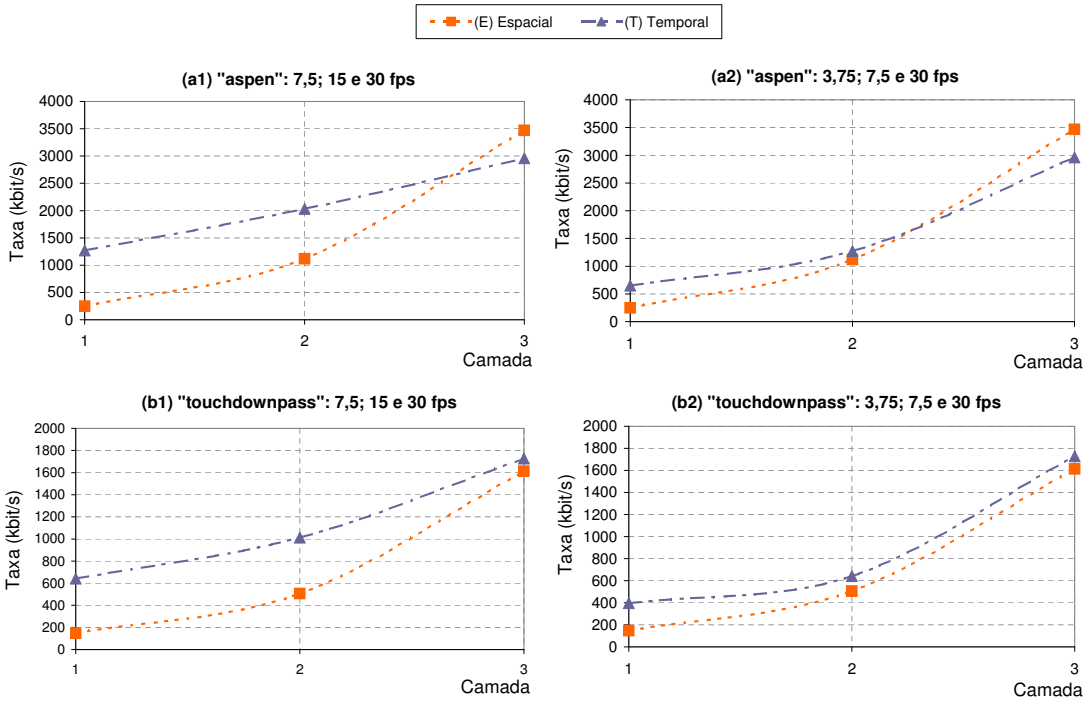


Figura 3.3: Diferença entre as camadas temporais quando utilizada uma camada com 15 fps e quando não utilizada.

Tabela 3.3: Tabela de comparação do das taxas de codificação das camadas de vídeo com e sem a camada utilizando 15 fps.

Camada	T **com 15 fps** **Taxa (kbit/s)**	T **sem 15 fps** **Taxa (kbit/s)**	E **Taxa (kbit/s)**
Vídeo: aspen			
C1	1271,07	650,79	247,38
C2	2034,02	1271,07	1118,01
C3	2960,64	2960,64	3468,66
Vídeo: touchdownpass			
C1	640,69	396,77	147,75
C2	1012,37	640,69	506,16
C3	1727,58	1727,58	1612,86

Para a configuração Qualidade, a resolução espacial foi fixada em 4CIF e a taxa de quadros por segundo em 30 fps, sendo que o parâmetro variável para esta configuração é a medida PSNR dos quadros. Inicialmente a terceira camada foi fixada com 38 dB, assim como foram todas as camadas para as outras duas configurações de codificação. O valor 38 dB foi escolhido por ser relativamente alto, normalmente representando uma imagem de boa qualidade. Com este valor é possível degradar o PSNR da terceira camada de forma facilmente perceptível (para criação da primeira e da segunda camada) e ainda

assim manter as camadas inferiores com uma qualidade razoável (ficando próximas de 32 dB). Foi admitido um erro de, no máximo, 1,5% para os valores do PSNR de todas as camada (exceto as duas primeiras da configuração Qualidade) em relação ao valor alvo 38 dB.

O PSNR para as duas primeiras camadas da configuração Qualidade (Q) foram escolhidos conforme duas premissas:

1. A taxa de codificação da primeira camada deve ser o mais similar possível à taxa da primeira camada para as configurações E e T. O PSNR da segunda camada também deve ser o mais similar possível ao PSNR da segunda camada das outras duas configurações. Isso é válido para cada vídeo individualmente (cada SRC).

2. Com a terceira camada fixada em 38 dB, o PSNR da primeira camada deve ser próximo de 32 dB. O PSNR da segunda camada deve ser o mais próximo possível da média entre o PSNR da primeira camada e o PSNR da terceira camada.

A escolha de 32 dB para a primeira camada foi feita com base nas codificações iniciais, onde percebeu-se que este valor de PSNR poderia produzir camadas respeitando a primeira premissa e com diferença de qualidade facilmente perceptível em relação à terceira camada, ou seja, a taxa da primeira camada com 32 dB é próxima à taxa das primeiras camadas das outras configurações de codificação e a diferença de qualidade entre os quadros da primeira camada com 32 dB para os quadros da terceira camada com 38 dB são facilmente perceptíveis.

A figura 3.4 mostra um exemplo do resultado da codificação de um dos vídeos, chamado "rushfieldcuts". O gráfico da esquerda mostra a taxa (kbit/s) obtida em cada camada (pontos no gráfico) de cada um dos métodos de codificação (linhas no gráfico). Note que a taxa é bastante semelhante entre as mesmas camadas dos diferentes métodos, como planejado. No gráfico da direita é exibido o Y-PSNR (valor médio do PSNR do canal Y de todos os quadros do vídeo, medido em dB) para cada camada de cada um dos métodos. É importante perceber que todas as camadas dos métodos E e T permanecem muito próximas dos 38 dB, assim como a última camada do método Q. Já as duas primeiras camadas do método Q possuem valores de Y-PNSR mais baixo, por ser justamente o objetivo da codificação escalável de qualidade. Estes gráficos são semelhantes para os outros vídeos, que serão exibidos na seção 3.3.

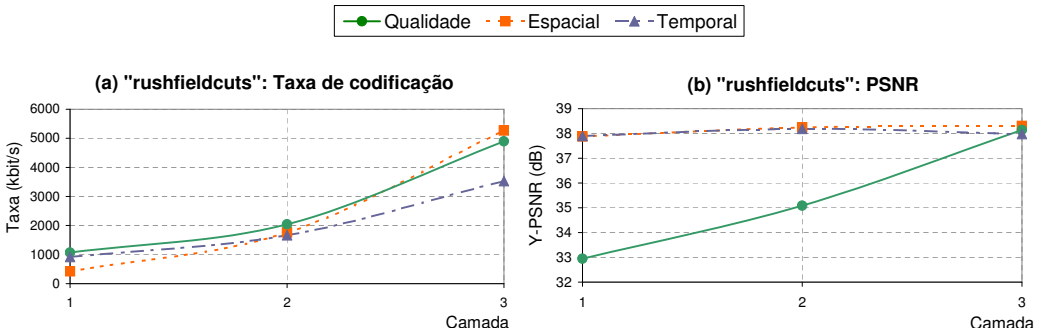


Figura 3.4: Exemplo dos resultados da codificação de um dos vídeos utilizados, chamado "rushfieldcuts".

3.2.2 Padrões de instabilidade

Como comentado na seção 2.3, os protocolos de controle de congestionamento normalmente são implementados e validados em simuladores de redes, o que não viabiliza seu uso em transmissões reais. Devido à não existência de um sistema de transmissão em camadas tal qual o modelo utilizado como base neste trabalho, não foi possível utilizar transmissões reais para verificar a instabilidade dos protocolos. Por este motivo, a instabilidade dos vídeos teve que ser simulada com base em resultados obtidos nas simulações de alguns protocolos de controle de congestionamento.

A instabilidade foi simulada com variações entre duas camadas de vídeo: o receptor se cadastra em uma camada superior à atual (processo de *join*, quando utilizado multicast, aumentando a qualidade do vídeo) e, após um curto período de tempo, deixa esta camada (processo de *leave*). Foram definidos três padrões de variação das camadas, ou padrões de instabilidade, que são descritos na tabela 3.4. A figura 3.5 mostra graficamente esses três padrões de instabilidade.

Tabela 3.4: Definição dos três padrões de instabilidade.

Nome		**Número de variações**
$p0$	Padrão 0	Sem variações, estável
$p4$	Padrão 4	16 variações por minuto ou 4 variações em 14 segundos
$p8$	Padrão 8	32 variações por minuto ou 8 variações em 14 segundos

Na figura 3.5, a linha em cada gráfico mostra a camada na qual a transmissão/vídeo está a cada instante de tempo durante os 14 segundos de vídeo. Os gráficos superiores (a) mostram os três padrões aplicados entre as camadas 1 e 2 de vídeo, enquanto os gráficos inferiores (b) mostram os mesmos padrões aplicados entre as camadas 2 e 3. Ou seja, cada um dos três padrões foi aplicado duas vezes para cada configuração de codificação (Q, E e T). Com estas configurações, além de comparar a estabilidade com a instabilidade, é possível verificar se variações em camadas superiores (melhor qualidade) apresentam resultados melhores (ou piores) do que variações em camadas inferiores, por exemplo. Por este motivo é que os padrões de instabilidade foram aplicados tanto entre as camadas 1 e 2 quanto entre as camadas 2 e 3.

Nos padrões de instabilidade, cada processo de adicionar (*join*) ou deixar (*leave*) uma camada é considerado uma variação, ou seja, como pode ser visto na figura 3.5, o padrão $p4$ é formado por dois *joins* e dois *leaves* (totalizando 4 variações) e o padrão $p8$ é formado por quatro *joins* e quatro *leaves* (totalizando 8 variações). Os processos de *leave* são feitos 1 ou 2 segundos após o *join* que o precedeu, pois este é um intervalo normalmente utilizado pelos protocolos de controle de congestionamento como um período de estabilização (ROESLER, 2003).

Estes padrões foram definidos com base nas simulações dos protocolos de controle de congestionamento comentados na seção 2.3. Os casos mais instáveis vistos chegam a pouco mais que 25 variações de camada por minuto, enquanto a maioria dos casos se encontra na faixa de 5 à 20 variações por minuto. Na tese de doutoramento de Roesler (ROESLER, 2003), onde foi proposta a base do projeto SAM, diversas simulações foram realizadas envolvendo os protocolos ALMP, ALMTF, RLM, RLC e TFMCC (que já foram

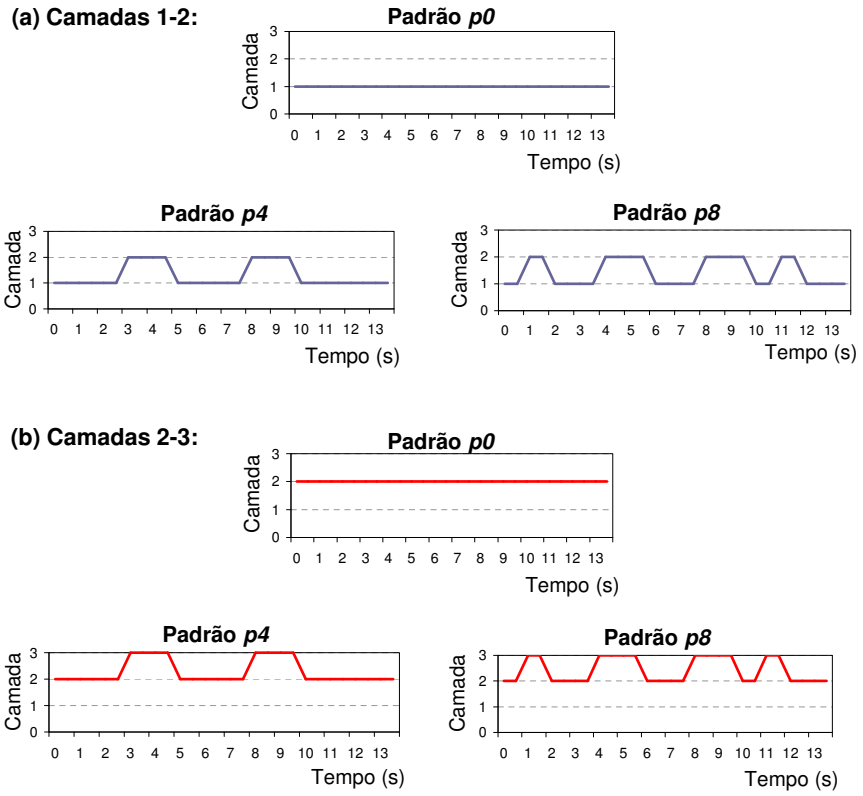


Figura 3.5: Variações das camadas ao longo do tempo nos padrões de instabilidade.

comentados na seção 2.3). A tabela 3.5 mostra um exemplo dos resultados encontrados em relação ao número de variações de camada por minuto (VCM) para cada um dos protocolos examinados. Estes resultados foram calculados em um cenário de transmissão com apenas um transmissor e 4 receptores ($rec1$, $rec2$, $rec3$ e $rec4$), cada um localizado em um local diferente, ou seja, em um enlace de rede diferente. A velocidade do enlace dos receptores é 2,1 Mbit/s ($rec1$), 1,05 Mbit/s ($rec2$), 525 kbit/s ($rec3$) e 105 kbit/s ($rec4$). As camadas de vídeo utilizadas foram 6 camadas exponenciais, com taxas de 30 kbit/s, 60 kbit/s, 120 kbit/s, 240 kbit/s, 480 kbit/s e 960 kbit/s.

Tabela 3.5: Número de variações de camadas por minuto para alguns protocolos de controle de congestionamento.

Protocolo	**Variações por minuto**			
	rec1	**rec2**	**rec3**	**rec4**
ALMP	0,00	0,24	0,60	3,12
ALMTF	0,00	2,00	13,90	21,60
RLM	0,00	0,00	1,80	0,90
RLC	0,00	7,10	0,00	0,00
TFMCC	0,00	0,00	0,00	0,00

Em outras simulações com uma maior quantidade de transmissores, receptores e tráfegos concorrentes, o número de variações de camadas aumenta consideravelmente, como pode ser visto na tabela 3.6. Estes resultados foram obtidos em simulações com 10 fluxos

do mesmo algoritmo sendo transmitidos em um enlace comum aos 10 receptores e 10 transmissores, onde cada receptor recebia o fluxo de um transmissor diferente. As camadas de vídeo são as mesmas camadas exponenciais citadas anteriormente e a banda do enlace é de 5 Mbit/s.

Tabela 3.6: Variações de camadas por minuto em um ambiente com 10 fluxos concorrentes.

Protocolo	**Variações por minuto**		
	Maior	**Menor**	**Média**
ALMP	0,60	0,00	0,20
ALMTF	12,20	8,60	10,10
RLM	3,60	1,80	2,80
RLC (oito fluxos)	12,90	2,70	4,20
TFMCC	21,60	14,40	19,02

Com as configurações de codificação e os padrões de instabilidade definidos, foram criados 18 HRCs, além do vídeo de referência. O vídeo de referência pode ser considerado um HRC onde nenhuma alteração é imposta ao vídeo original, portanto ele permanece com, teoricamente, sua maior qualidade. Esses 18 HRCs são resultado da combinação dos três métodos de escalabilidade (as configurações de codificação T, E e Q) e três padrões de instabilidade ($p0$, $p4$ e $p8$), que são aplicados duas vezes (camadas 1-2 e 2-3) para cada método de escalabilidade, ou seja, 3 métodos $\times$ 3 padrões de instabilidade $\times$ 2 aplicações de cada $=$ 18 HRCs.

A aplicação dos padrões de instabilidade produz novos vídeos, onde os momentos em que as trocas de camadas são feitas normalmente são vistos como uma variação na blocagem, borramento ou na fluidez do fluxo de vídeo (“trancadas”). Por exemplo, uma variação entre as camadas 2 e 3 para a configuração Temporal resulta na troca da taxa de quadros por segundo do vídeo de 7,5 fps para 30 fps e vice-versa. Quando há uma redução de 30 fps para 7,5 fps, a variação é vista como uma “trancada” no fluxo de vídeo (*frame freezes*, como conhecido em inglês), e isto normalmente reduz a qualidade do vídeo que será percebida pelo observador. Já para as configurações Espacial e Qualidade, as variações entre as camadas resultam em aumento ou redução da blocagem e/ou borramento dos quadros.

É importante observar que nestas avaliações a troca de camadas é instantânea e não implica em nenhuma perda de qualidade adicional além daquela existente entre uma camada e outra. Ou seja, não são consideradas perdas de pacotes que provavelmente aconteceriam para que o sistema precisasse deixar a camada atual (os protocolos normalmente verificam que aconteceram perdas e então deixam de receber determinada(s) camada(s)).

3.3 Seleção e processamento dos vídeos

Nesta seção é descrito o processo de seleção dos vídeos que foram utilizados nas avaliações e o processamento que foi aplicado a eles, incluindo as etapas práticas da codificação escalável e da simulação da instabilidade, cujas definições já foram descritas na seção 3.2. O conteúdo está dividido em 5 subseções: pré-seleção, pré-processamento, seleção final, codificação escalável e simulação da instabilidade. Cada subseção descreve

uma das etapas do processo e a ordem em que elas são descritas representa a ordem em que as etapas foram realizadas durante o desenvolvimento do trabalho.

3.3.1 Pré-seleção

A primeira etapa para a seleção dos vídeos originais que seriam utilizados nas avaliações foi a realização de um levantamento de vídeos que já são utilizados para avaliações subjetivas em outros trabalhos. Cerca de 80 vídeos com conteúdos diferentes foram inicialmente considerados para uso. Diversos deles foram obtidos dos repositórios do grupo VQEG, que correspondem aos vídeos utilizados pelo grupo nas diversas fases de seus projetos. Eles serão identificados por VQEG I-II (utilizados na fase 1 e/ou 2 dos testes do grupo) e pelo prefixo HDTV (fase atual, divisão HDTV). Outros vídeos foram obtidos dos repositórios do JSVM, o aplicativo que foi utilizado para codificação dos vídeos (será comentado na seqência desta seção), e da norma ANSI T1.801.01. Por fim, um filme chamado Elephants Dream, um curta metragem criado utilizando computação gráfica, também foi utilizado. Abaixo as fontes utilizadas são listadas junto com os endereços onde os vídeos podem ser encontrados na Internet[1].

- **VQEG: Video Quality Experts Group**

  Os vídeos obtidos do grupo VQEG estão identificados por: VQEG I-II, HDTV NTIA, HDTV SVT-ex e HDTV SVT-mf. Todos podem ser encontrados no website do grupo, em <http://www.its.bldrdoc.gov/vqeg>. Abaixo segue uma breve descrição de cada grupo de vídeos do VQEG com um endereço para facilitar o acesso:

  VQEG I-I: Vídeos utilizados na primeira ou na segunda fase dos trabalhos do VQEG.

  Endereço: <ftp://ftp.crc.ca/crc/vqeg/>

  Endereço alternativo: <http://media.xiph.org/vqeg/TestSeqences/Reference/>

  HDTV: Vídeos da fase atual do grupo, utilizados pela divisão que trabalha com vídeos em alta definição (HD). Eles possuem três fontes diferentes: NTIA (NTIA_source), SVT-ex (SVT_exports), e SVT-mf (SVT_MultiFormat).

  Endereço: <ftp://vqeg.its.bldrdoc.gov/HDTV/>

- **JSVM: Joint Scalable Video Model**

  JSVM é o nome dado ao modelo e implementação de referência do padrão de codificação de vídeo escalável H.264 SVC. Este modelo foi desenvolvido pelo JVT, e alguns vídeos estão disponibilizados em seus servidores, identificados como vídeos do JSVM ou do SVC.

  Endereço: <ftp://ftp.tnt.uni-hannover.de/pub/svc/testsequences/>

- **ANSI T1.801.01**

  O padrão ANSI T1.801 foi criado pelo ITS (*Institute for Telecommunication Sciences*) para auxiliar na avaliação de qualidade de sistemas de vídeo digital. O padrão é composto por 3 partes, sendo que a primeira, a ANSI T1.801.01, provê um conjunto padrão de vídeos digitais que podem ser utilizados para execução de avaliações subjetivas ou objetivas.

  Endereço: <ftp://vqeg.its.bldrdoc.gov/SDTV/ANSI_T1_801_01/>

[1] Todos endereços foram acessados em fevereiro de 2009.

- **Elephants Dream**

  Elephants Dream é o nome dado ao filme que é considerado o primeiro "filme aberto"(*open movie*) já criado. O filme é um curta metragem de animação, que foi construído inteiramente utilizando software livre. Todo o material utilizado para produção do filme está disponível, inclusive o filme completo em formato original, ou seja, sem nenhuma compactação. Ele foi considerado para uso por estar disponível no formato HD, ser bastante extenso (foi possível remover alguns trechos para usar nas avaliações) e por conter animações em computação gráfica (nas outras fontes existem poucos vídeos de animação).

  Endereço: <http://orange.blender.org/>

Todos os vídeos coletados são vídeos não compactados e estavam armazenados no formato YUV (amostragem 4:2:2 ou 4:2:0) ou RGB, com pelo menos 8 bits para cada componente de cada pixel. A maior parte deles estava disponível em mais de um formato, normalmente em mais de uma resolução espacial e/ou temporal. A tabela 3.7 mostra um resumo dos formatos dos vídeos de cada uma das fontes utilizadas, exibindo as maiores resoluções temporais e espaciais encontradas. Algumas fontes também possuem vídeos com resoluções menores, como é o caso da HDTV SVT-ex, que possui todos os vídeos também com 25 fps e 29.997 fps. A tabela 3.7 apresenta apenas os casos mais importantes para os propósitos deste trabalho.

Tabela 3.7: Formatos dos vídeos coletados.

Fonte	Quant.	fps	Quadros	Tempo	i/p	Resolução	Formato
VQEG I-II	10	60	260	8,8s	i	720x486	NTSC
	10	50	220	8,8s	i	720x576	PAL
	15	30	360	12s	p	640x480	VGA
HDTV NTIA	8	30	570	19s	p	1280x720	HD 720p
HDTV SVT-ex	3	60	604	10s	p	1280x720	HD 720p
		50	504	10s	p	1280x720	HD 720p
HDTV SVT-mf	5	50	500	10s	p	3840x2160	HD 2160p
JSVM	5	30	300	10s	p	704x576	4CIF
		60	600	10s	p	704x576	4CIF
	4	30	300	10s	p	352x288	CIF
ANSI T1.801.01	1	60	499	16,6s	i	720x486	NTSC
	6	60	450	15s	i	720x486	NTSC
	15	60	389	12,9s	i	720x486	NTSC
Elephants Dream	1	30	>10.000	≈10m	p	1920x1080	HD 1080p

Na tabela 3.7, a coluna “Quant.” indica a quantidade de vídeos com conteúdos diferentes disponíveis. No caso do VQEG I-II, por exemplo, 10 vídeos estão disponíveis em um formato, 10 vídeos em outro formato e outros 15 vídeos em outro formato, enquanto no caso do HDTV SVT-ex, os mesmos 3 vídeos estão disponíveis em dois formatos diferentes. A coluna “*fps*” informa o número de quadros por segundo do vídeo, “Quadros” contém o número total de quadros e “Tempo” mostra o tempo total de execução dos vídeos. A próxima coluna, “i/p”, indica se o vídeo é entrelaçado (i) ou progressivo (p),

enquanto “Resolução” mostra a resolução espacial dos vídeos e “Formato” apresenta o nome sob o qual este formato é conhecido.

Nota-se que a maioria dos vídeos possui duração menor do que os 14s impostos no plano de avaliação (seção 3.2). Apesar disso, eles foram examinados para verificar as características de um conjunto grande de vídeos (como as medidas TI e SI, que serão comentados na seção 3.3.3), mesmo se acabassem não sendo utilizados. Além disso, uma possibilidade seria reduzir a duração dos vídeos no plano de avaliação caso não houvesse um número suficiente de vídeos com características adequadas, o que não foi necessário.

3.3.2 Pré-processamento

Como visto na tabela 3.7, os vídeos obtidos estavam em formatos variados. Antes de utilizá-los, e antes do processo de seleção de quais seriam utilizados, eles foram convertidos para o formato padrão que seria utilizado nas avaliações. A maior resolução espacial usada neste trabalho é 4CIF e a maior taxa de quadros por segundo é 30 fps, portanto todos os vídeos foram convertidos para este formato.

Para realizar a conversão, foram utilizados os aplicativos AviSynth (versão 2.5), VirtualDub (versão 1.7.8) e FFmpeg (versão SVN-r11870, fevereiro de 2008). O aplicativo AviSynth foi utilizado para realizar a maioria das tarefas, incluindo o corte de regiões dos quadros (*crop*), redimensionamento, conversão do formato original para YUV com amostragem 4:2:0 (FOURCC I420), remoção de quadros, desentrelaçamento e redução da taxa de quadros por segundo (de 60 fps para 30 fps, por exemplo). O AviSynth é um aplicativo que funciona como um servidor de vídeo (*frameserver*), onde os vídeos são processados instantaneamente quando solicitados por outra aplicação, utilizando comandos especificados em arquivos de configuração. Por este motivo, neste trabalho foi utilizado o VirtualDub, que solicita o processamento do vídeo para o AviSynth e armazena os arquivos processados. Como o VirtualDub faz a gravação dos arquivos no formato AVI, foi utilizado o FFmpeg para remover os cabeçalhos dos arquivos AVI e transformá-los em vídeos YUV, prontos para serem utilizados pelo codificador escalável. O apêndice C.1 contém alguns exemplos práticos da utilização destes aplicativos, assim como alguns arquivos de configuração utilizados.

O processamento foi feito de forma a modificar o mínimo possível os vídeos originais, e os aplicativos (AviSynth e VirtualDub) e filtros (Lanczos para redimensionamento, por exemplo) utilizados são recomendados para uso nas avaliações do VQEG (VQEG, 2008a).

As operações citadas (as realizadas pelo AviSynth) não foram necessárias em todos os casos, pois cada formato de vídeo de entrada exigiu um tratamento diferente. Em relação à resolução espacial, para os vídeos em formato HD, inicialmente eram removidas áreas da esquerda e da direita do vídeo para mudança do aspecto de 16:9 (1,78:1) para 1,22:1 (aspecto da resolução 4CIF), e então eles eram redimensionados para 4CIF (704x576). Para vídeos com resoluções próximas à 4CIF, como alguns vídeos do VQEG I-II que possuem resolução 720x576, não foi necessário o redimensionamento, bastou a remoção de algumas colunas à esquerda e à direita dos vídeos (8 colunas de cada lado no exemplo dado) para adaptá-los à 4CIF. Vídeos com resoluções menores do que 4CIF *não* foram redimensionados, pois o redimensionamento para ampliar a resolução dos vídeos poderia prejudicar sua qualidade. Estes vídeos tiveram sua resolução mantida mas acabaram não sendo utilizados nas avaliações.

Quanto à resolução temporal, inicialmente os vídeos entrelaçados foram transformados para vídeos progressivos através do AviSynth. Posteriormente, vídeos com 50 ou 60 quadros por segundo foram reduzidos para a taxa de 30 quadros por segundo e os vídeos

com duração mais longa do que 14 segundos foram reduzidos para exatos 14 segundos. A localização destes 14 segundos dentro do vídeo foi escolhida conforme o conteúdo de cada um, de forma que a sequência de quadros fosse contínua e representasse bem o conteúdo do vídeo completo. Por ser o vídeo mais extenso, do vídeo Elephants Dream foram extraídas 3 sequências de 14 segundos, chamadas de “ed1”, “ed2” e “ed3”.

Após o processamento, todos os vídeos foram verificados para ver se as operações não incluíram artifícios ou pioraram a qualidade de alguma maneira. Nos casos dos vídeos entrelaçados, foi possível perceber que a etapa de transformação para vídeos progressivos piorou a qualidade consideravelmente em alguns vídeos. Estes poucos vídeos que tiveram sua qualidade perceptivelmente reduzida já foram descartados do processo de seleção. Vale observar que durante as avaliações os vídeos originais foram utilizados como referência, ou seja, foram apresentados aos avaliadores juntamente com o restante dos vídeos. Portanto, por mais que o pré-processamento tenha reduzido um pouco a qualidade de alguns vídeos, esta diferença não aparece nos resultados, que são calculados com base na qualidade atribuída aos vídeos de referência.

3.3.3 Seleção final

Dentro do conjunto de vídeos já pré-selecionados e pré-processados, três fatores principais foram considerados para seleção final dos vídeos que realmente foram utilizados. Esses fatores são: (i) as resoluções espaciais e temporais mínimas disponíveis, (ii) as medidas TI (*Temporal Information*) e SI (*Spatial Information*), e (iii) o conteúdo dos vídeos.

Em relação às resoluções temporal e espacial, inicialmente os vídeos com resolução menor do que 4CIF (704x576) foram eliminados, para que não fosse necessário redimensionar esses vídeos para aumentar sua resolução. Já os vídeos com menos de 14 segundos não foram inteiramente descartados, até mesmo os vídeos com duração menor (que são a maioria, como pode ser visto na tabela 3.7) foram considerados nas próximas etapas da seleção. Caso não fosse possível encontrar o número estabelecido de vídeos (8 para avaliação e 3 para treinamento), seria considerada uma modificação no plano de avaliação para reduzir a duração dos vídeos. Esta modificação, porém, não foi necessária.

A próxima etapa da seleção foi o cálculo das medidas TI e SI de cada um dos vídeos. Estas medidas são descritas na norma ITU-T Rec. P.910 (ITU-T, 1999), e são definidas como as medidas da complexidade temporal e espacial dos vídeos, respectivamente. A complexidade temporal e espacial dos vídeos é crítica para a seleção dos vídeos, pois elas determinam o nível de compressão que será possível atingir durante a codificação e, portanto, o nível de degradação que os vídeos irão sofrer no processo.

Os valores das medidas TI e SI são calculados individualmente para cada quadro dos vídeos e, após o cálculo dos dois valores para todos os quadros, o valor máximo de cada sequência é considerado como o valor do TI e SI do vídeo como um todo. A variabilidade de cada medida ao longo dos quadros também pode ser considerada caso seja necessária uma análise mais detalhada das características dos vídeos. Abaixo são detalhados os processos para cálculo das medidas TI e SI conforme descritos na ITU-T Rec. P.910.

SI (Informação Espacial): O cálculo da informação espacial é feito quadro-a-quadro com base no filtro de Sobel (detalhes sobre este filtro podem ser encontrados em diversos livros, como no livro de Gonzalez e Woods (GONZALEZ; WOODS, 1987)). Inicialmente, é aplicado o filtro de Sobel ($Sobel()$) no canal de luminância (F) de cada quadro (localizado no instante de tempo n no vídeo). Após a aplicação do filtro, é calculado o desvio padrão ($std()$) dos pixels do quadro F_n filtrado. Esta

operação é feita para todos os quadros da imagem, resultando em um vetor contendo um valor de SI para cada quadro. Finalmente, o valor máximo deste vetor ($max()$) é utilizado como o valor SI para o vídeo inteiro. A equação 3.1 representa o cálculo do SI de um vídeo.

$$SI = max\{std[Sobel(F_n)]\} \tag{3.1}$$

O filtro de Sobel consiste na aplicação de duas máscaras de tamanho 3x3 sobre um quadro e posterior cálculo da raiz quadrada da soma dos quadrados dos resultados das convoluções. Mais detalhes são explicados na própria ITU-T Rec. P.910 (ITU-T, 1999). A norma também recomenda que o cálculo seja feito apenas em uma sub-área da imagem, eliminando as partes superior, inferior e laterais, que são áreas normalmente não visíveis em monitores e televisores CRT.

Não há um limite máximo para os valores de SI, mas eles normalmente encontram-se no intervalo entre 0 e 250.

TI (Informação Temporal): A informação temporal é baseada na diferença dos valores entre os pixels (do canal de luminância) localizados na mesma posição espacial mas em diferentes quadros do vídeo. Inicialmente, esta diferença ($M_n(i,j)$) é calculada para todos os pixels de todos os quadros consecutivos da imagem, como mostra a equação 3.2. Na função, $F_n(i,j)$ representa o pixel da linha i e coluna j do canal de luminância do quadro localizado na posição temporal n no vídeo.

$$M_n(i,j) = F_n(i,j) - F_{(n-1)}(i,j) \tag{3.2}$$

Assim como no cálculo da medida SI, cálculo do TI final é feito através do desvio padrão ($std()$) de todos os valores $M_n(i,j)$ calculados. Ou seja, para cada posição espacial (i,j), é calculada a diferença M_n dos pixels entre todos os quadros consecutivos, gerando um vetor de diferenças para esta posição espacial. Por fim, é calculado o desvio padrão deste vetor.

O valor final do TI será o valor máximo ($max()$) entre os valores de desvio padrão de todas as posições espaciais. A equação 3.3 representa o cálculo do TI.

$$TI = max\{std[M_n(i,j)]\} \tag{3.3}$$

Quanto maior o movimento entre quadros consecutivos, maior será o valor final do TI. Por este motivo, em vídeos que possuem cortes de cena, podem ser utilizados dois valores de TI: um considerando os momentos de corte e um sem considerar estes momentos.

Para o TI também não há um limite máximo, mas os valores normalmente encontram-se no intervalo entre 0 e 100.

Uma aplicação foi desenvolvida para cálculo das medidas TI e SI de acordo com as metodologias descritas acima. Esta e outras aplicações implementadas (que serão comentadas no decorrer das próximas seções) são descritas em maiores detalhes no apêndice B.

Como comentado, a aplicação do filtro de Sobel (utilizada no cálculo do SI) normalmente é realizada em uma subárea dos quadros, descartando as laterais e partes superior

e inferior. A ITU-T Rec. P.910 sugere a remoção de colunas de 20 pixels das laterais e de linhas também de 20 pixels das partes superior e inferior. O filtro deve ser aplicado somente sobre a região central, como ilustrado na figura 3.6. Em função disso, o aplicativo desenvolvido realiza o cálculo do SI apenas nesta área central, utilizando os mesmos 20 pixels como largura das colunas e linhas descartadas. Alguns ensaios também foram realizados sem descartar nenhuma região dos quadros, porém, a alta correlação entre os resultados dos valores de SI com e sem o descarte (correlação de Pearson resultou em 0,998) mostrou que os valores de SI foram muito parecidos, portanto o descarte foi mantido.

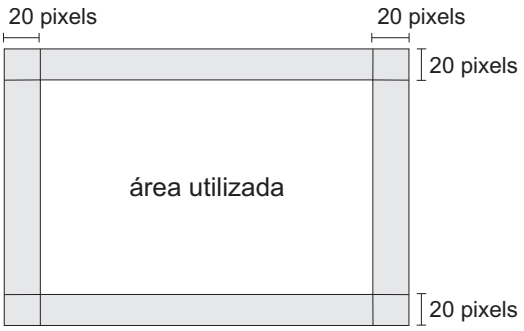


Figura 3.6: Área utilizada para aplicação do filtro de Sobel e cálculo da medida SI.

Após a obtenção do TI e SI de todos os vídeos, é construído um gráfico onde cada vídeo representa um ponto na matriz espaço-temporal. A seleção deve incluir vídeos dispersos ao longo de todas as áreas da matriz, de acordo com os propósitos do trabalho que está sendo realizado. Por exemplo, quando deseja-se realizar comparações entre codificadores de vídeo em relação a vídeos com alta complexidade de codificação temporal, seria interessante selecionar apenas vídeos com valor de TI maiores do que 60, mas que incluíssem diversas variações de valores de SI. Neste trabalho, a seleção é feita incluindo alguns valores extremos (TI ou SI bastante baixo ou bastante alto) e mantendo a distribuição dos vídeos selecionados semelhante à distribuição de todos os vídeos no gráfico.

A figura 3.7 mostra dois gráficos construídos com os valores TI e SI dos vídeos coletados. O primeiro gráfico (a) mostra a distribuição dos valores para todos os vídeos coletados, que são praticamente todos os vídeos da tabela 3.7 *após* o pré-processamento. Para o vídeo Elephants Dream, foram utilizadas as três sequências de 14 segundos que foram extraídas (como comentado na seção 3.3.2). Neste gráfico, cada ponto (símbolo $+$) representa um dos vídeos. No segundo gráfico (b), os mesmos vídeos do gráfico (a) são exibidos, porém os vídeos selecionados para o treinamento e para a avaliação estão destacados e identificados com números de 1 a 11. Esses números serão utilizados posteriormente para identificar os vídeos.

Como pode ser visto, a maioria dos vídeos selecionados está em meio à "nuvem" de pontos centrais, que representam a localização da maioria dos vídeos. Já alguns vídeos, como os pontos identificados por 1, 7 e 8, estão em extremidades com valores de informação espacial bastante alta ou informação temporal bastante baixa, por exemplo.

A figura 3.7 já exibe os 11 vídeos selecionados conforme os valores de TI e SI, porém, além dessas medidas, também foi considerado o conteúdo dos vídeos, como comentado no início desta seção. Uma das preocupações em relação à esta escolha está em selecionar a maior variação possível de conteúdos, com o propósito de manter a atenção dos observadores durante a avaliação (para não entediar os observadores, como comentado na ITU-T Rec. P.910) e de abranger os diversos conteúdos que podem ser transmitidos em

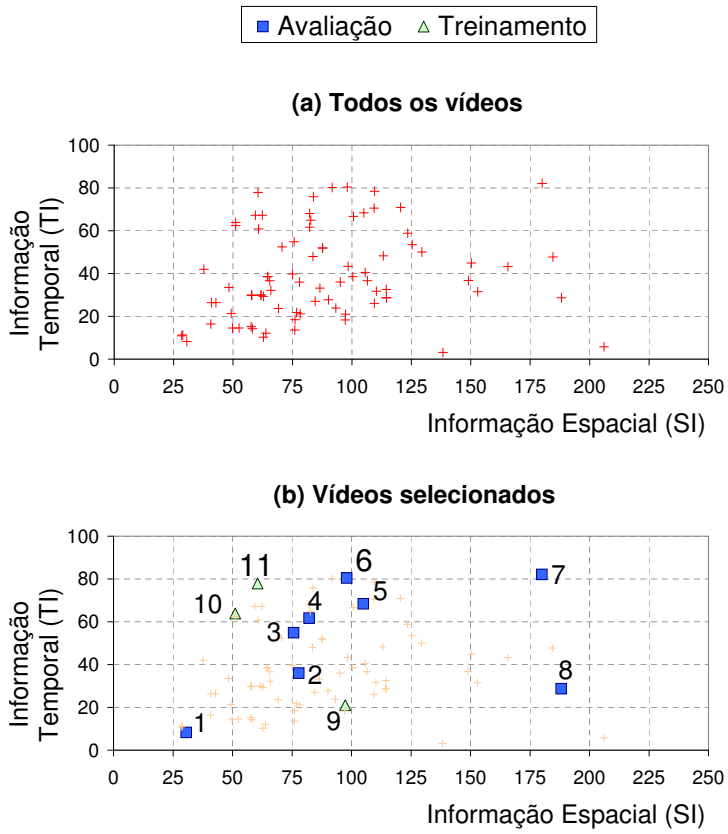


Figura 3.7: Valores TI e SI para os vídeos coletados, com destaque para aqueles que foram selecionados.

sistemas de propósito geral. Essas variações de conteúdo incluem variações nos diversos elementos que um vídeo pode possuir, como a presença ou não de água, multidões, rostos, letras, animações, movimento rápido ou lento, entre outros, fatores que também influenciam nos valores de TI e SI calculados anteriormente.

Apesar de ser desejável uma grande variedade de conteúdos, é importante que estes conteúdos sejam comumente utilizados em ambientes de transmissão similares aos que estão sendo avaliados. Neste trabalho, as avaliações abrangem sistemas de transmissão que podem ter diversos propósitos, portanto o conteúdo dos vídeos não foi limitado a alguma área específica. Eles foram escolhidos de forma a representar conteúdos normalmente vistos em sistemas de televisão, com cuidado para não selecionar vídeos com conteúdos que pudessem dispersar a atenção dos observadores (vídeos engraçados, por exemplo). Como a maioria dos vídeos coletados já são utilizados em avaliações subjetivas por grupos como o VQEG, seus conteúdos já foram previamente selecionados em relação a estas últimas restrições comentadas.

Os vídeos finalmente selecionados acabaram sendo apenas das fontes HDTV NTIA e Elephants Dream. Estes eram os vídeos com duração mais longa entre os inicialmente coletados, e a distribuição dos valores de TI e SI deles ficou extremamente adequada para que todos fossem selecionados. Além disso, o conteúdo desses 11 vídeos é bastante variado e está de acordo com os outros requisitos comentados anteriormente. A variação dos conteúdos dos vídeos poderia ser ainda maior caso vídeos de outras fontes fossem selecionados, mas esta pequena vantagem não justificaria uma mudança, visto que os vídeos selecionados estão de acordo com os outros requisitos e, especialmente, devido à

boa distribuição de valores TI e SI que eles tiveram.

A tabela 3.8 mostra o nome dos 8 vídeos selecionados para a avaliação, os valores de TI e SI e uma breve descrição do conteúdo de cada um deles. A tabela 3.9 mostra os mesmos campos, mas agora para os 3 vídeos selecionados para o treinamento. Além das tabelas, a figura 3.8 exibe uma imagem de cada um dos 11 vídeos selecionados para ajudar na sua caracterização.

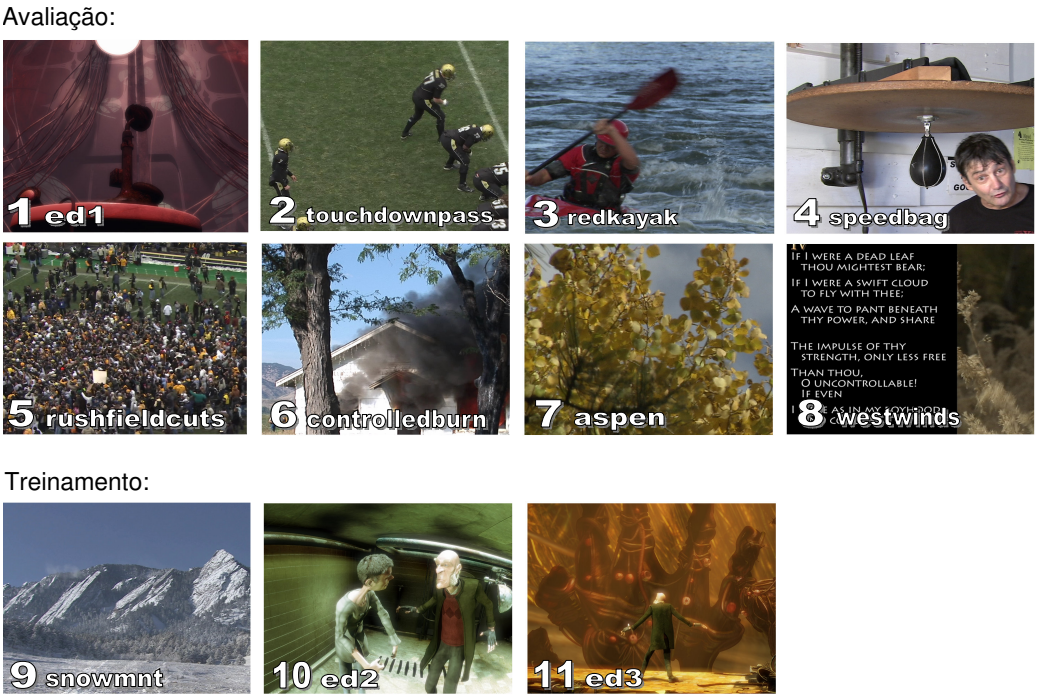


Figura 3.8: Um quadro de exemplo para cada um dos 11 vídeos selecionados.

3.3.4 Codificação escalável

Antes da codificação escalável dos vídeos, cada um deles foi expandido com a inclusão de 2 segundos (mais precisamente, 64 quadros) em seu início e fim. Os quadros incluídos são os mesmos quadros do início e do fim do vídeo, replicados em ordem inversa. Ou seja, são selecionados os primeiros 2 segundos do vídeo, a sequência de execução deste trecho é invertida e ele é inserido no início do vídeo. O mesmo acontece em relação aos últimos 2 segundos do vídeo. Este processo é feito para eliminar as variações existentes no período inicial e final da codificação enquanto o codificador ainda não está estável e a influência que isso pode ter nos resultados. Com a expansão dos vídeos, também é simulada uma transmissão mais longa, onde a codificação está sendo feita durante um longo período e um trecho central desta transmissão é removido para avaliação. O uso dos mesmos quadros em ordem inversa também auxilia nesta simulação de uma transmissão mais longa, pois se mantém um padrão espacial semelhante ao do vídeo não expandido (os quadros adicionais são semelhantes, pois são os mesmos que estão no vídeo) e mantém-se a variação temporal também semelhante à do vídeo não expandido, ou seja, não há uma "quebra" entre os quadros adicionais e os quadros do vídeo original, portanto a transição é suave (o que influência na criação dos vetores de movimento durante a codificação). Esses 4 segundos adicionais foram incluídos apenas para a codificação. Logo após a codificação eles foram removidos, ou seja, as avaliações e todos os cálculos

Tabela 3.8: Descrição dos vídeos selecionados para a **avaliação**.

ID	Vídeo	TI	SI	Descrição
1	ed1	8,31	30,56	Animação. Imagem de um portão metálico com movimento da câmera lentamente para baixo. Na metade da exibição aparecem alguns objetos em destaque próximos à câmera. Cores fortes mas sem muitas variações. Sem cortes.
2	touchdownpass	36,01	77,92	Mostra uma jogada em um jogo de futebol americano, com o gramado verde e diversos jogadores inicialmente parados e, em seguida, em movimento rápido. Sem cortes.
3	redkayak	54,90	75,74	Rio com água em movimento e um homem remando em um caiaque vermelho. Apresenta bastante movimento da água e do vento e possui poucos cortes.
4	speedbag	61,63	82,24	Câmera próxima a um aparelho para treinamento de boxe que se movimenta muito rápido. Em seguida, apresenta a parte superior de um homem falando e então volta para a câmera inicial (ou seja, possui cortes). Apresenta grande variedade de cores.
5	rushfieldcuts	68,35	104,97	Grande quantidade de pessoas em movimento em um gramado verde. No início a imagem é distante, mas, a partir da metade, torna-se bastante próxima das pessoas (com um corte de cena).
6	controlledburn	80,38	98,02	Casa em chamas, no início com movimento de água (espirrada de uma mangueira) e no restante com chamas de fogo e cores intensas. Contém alguns cortes.
7	aspen	82,10	180,02	Imagens aproximadas de folhas e árvores, com o movimento do vento e o céu azul em alguns momentos. Pouca variação de cores mas diversos cortes ao longo da sequência.
8	westwindeasy	28,69	188,17	Imagem dividida em duas partes. No lado esquerdo o texto de um poema desliza para cima lentamente e, no lado direito, a imagem de uma planta com movimento rápido do vento. Sem cortes.

Tabela 3.9: Descrição dos vídeos selecionados para o **treinamento**.

ID	Vídeo	TI	SI	Descrição
9	snowmnt	21,07	97,32	Imagens de montanhas à distância com árvores em meio à neve. Possui dois cortes lentos (*fades*) para troca de cena e muito pouco movimento de câmera. Muito pouca variação de cores.
10	ed2	77,95	60,55	Animação. Dois personagens conversando em uma sala pequena. Cenas próximas e afastadas da câmera, mostrando o espaço da sala inteira e também detalhes dos personagens. Pouco movimento e alguns cortes.
11	ed3	63,83	51,13	Animação. Mostra um personagem interagindo com diversos objetos mecânicos, apresentando cores fortes e bastante iluminação. Possui movimento moderado, com poucos cortes.

(taxa de codificação e PSNR, por exemplo) foram realizados sobre os vídeos originais de 14 segundos.

Para inclusão e remoção dos quadros adicionais, foi desenvolvido um aplicativo em linguagem C (chamado *lyuv*). É informado ao aplicativo o número de quadros que devem ser inseridos e ele obtém esses quadros, inverte a ordem deles e os insere no início e fim do vídeo de entrada. O mesmo aplicativo também faz a remoção dos quadros adicionais do início e do fim dos vídeos. Alguns detalhes práticos e exemplos de como o aplicativo é utilizado são encontrados no apêndice B.

Para a codificação escalável dos vídeos, foi utilizado o JSVM (*Joint Scalable Video Model*), um conjunto de aplicativos desenvolvidos pelo JVT (*Joint Video Team*) como referência para o padrão H.264 SVC. As ferramentas do JSVM são utilizadas para converter os vídeos para CIF e QCIF, codificar, extrair camadas, decodificar e calcular o PSNR e taxa de codificação, normalmente executadas nesta ordem. Foi utilizada a versão 9.11, compilada a partir do código fonte disponível no servidor do JVT em março de 2008. No apêndice C.2 são exibidos exemplo e outras informações técnicas sobre o uso do JSVM. A figura 3.9 mostra o processo de execução das etapas da codificação, que estão identificadas pelos número de 1 à 5.

Como pode ser visto na figura 3.9, são 5 etapas principais que formam todo o processo de codificação: “1. DownConvert”, “2. Codificador”, “3. Extrator”, “4. Decodificador” e “5. Análise do PSNR e taxa de codificação”. A figura também mostra em que etapas entram os SRCs, HRCs e PVSs, já descritos anteriormente.

A etapa 1 é utilizada para redimensionar os vídeos (SRCs) de 4CIF para CIF e para QCIF. Esse processo é necessário pois o codificador do JSVM requer que os vídeos sejam previamente redimensionados quando são utilizadas camadas com múltiplas resoluções espaciais. Neste trabalho, isso acontece quando é feita a codificação utilizando a configuração Espacial, portanto todos os vídeos passaram por esta etapa. Para o redimensionamento foi utilizada a própria ferramenta do JSVM, que realiza a redução da dimensão

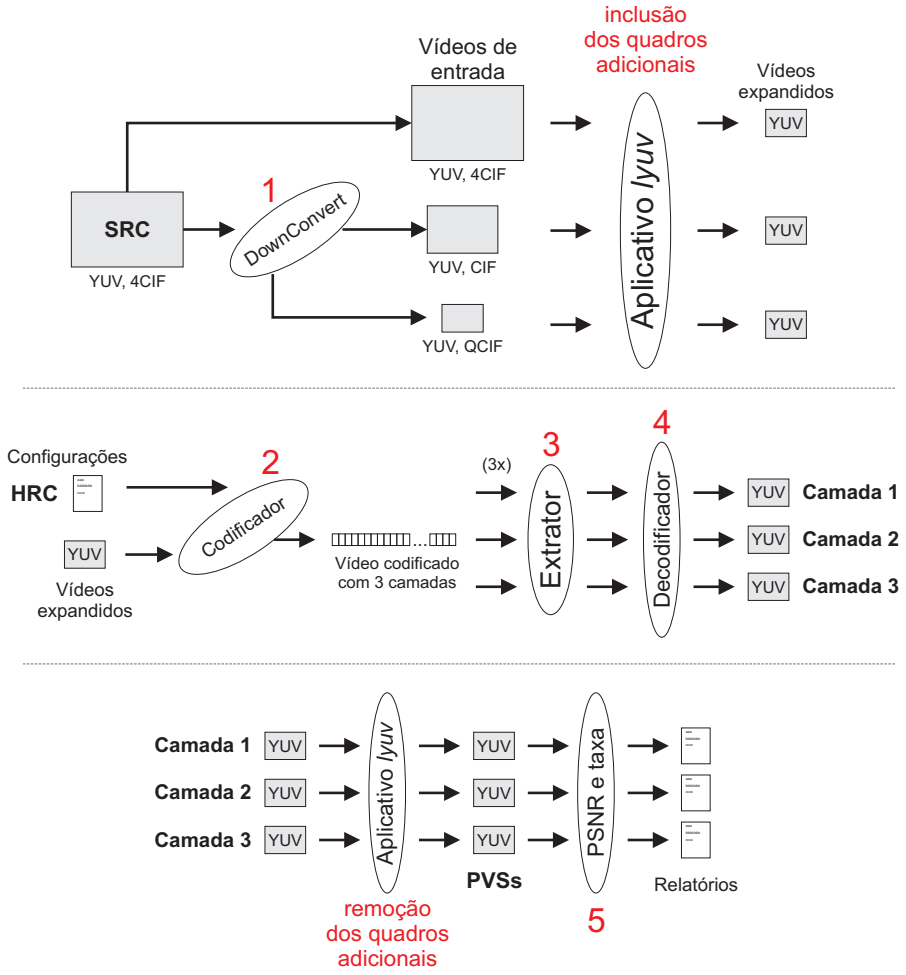


Figura 3.9: Etapas do processo de codificação escalável utilzando o JSVM.

espacial de forma especificada pelo padrão H.264 SVC. Vale observar que esta etapa é necessária para facilitar o uso do JSVM, pois o codificador poderia realizar este redimensionamento automaticamente, descartando a necessidade do redimensionamento prévio. Após o redimensionamento, cada vídeo é processado pelo aplicativo *lyuv* para inserção dos quadros adicionais, como já descrito no início desta seção.

A segunda etapa é a codificação escalável dos vídeos. O vídeo de entrada (ou os vídeos de entrada, no caso da configuração Espacial), é informado juntamente com as configurações de codificação, os HRCs (apenas as informações de codificação dos HRCs, já que eles também são formados pelas informações para simulação da instabilidade). O codificador do JSVM codifica os vídeos e cria um arquivo único (uma *bitstream*) com o vídeo codificado, contendo todas as camadas. A maior parte dos parâmetros de codificação permaneceu com o valor padrão estabelecido pelo JSVM. Já os parâmetros de quantização (chamados QP) variaram muito de vídeo para vídeo para que fosse possível manter as especificações dos objetivos (ver seção 3.2.1). Citando um exemplo, os QPs para a configuração Q do vídeo "aspen" foram 43, 37 e 27 (camadas 1, 2 e 3, respectivamente). Já para o vídeo "controlledburn", os QPs foram 40, 36 e 31.

Um aspecto importante na codificação é o uso de um GOP de tamanho 16, com período de repetição de quadros I igual a 32. Um GOP maior dificultaria a transição de camadas em um sistema real (SVC possui algumas restrições na transição entre camadas espaciais, que podem acontecer somente em quadros marcados como quadros-chave), que

nos experimentos desse trabalho podem ocorrer em intervalos espaçados em, no mínimo, 1 segundo. Alguns outros parâmetros importantes estabelecidos foram o uso CABAC para codificação entrópica, predição *inter-layer* sempre de forma adaptativa, *FastSearch* com range 96 para predição de movimento (SAD-YUV e SAD-Y para medidas de distorção) e quadros B hierárquicos para codificação temporal. Tentou-se utilizar técnicas avançadas (como o caso do CABAC) para conseguir um bom desempenho de codificação, mas sem exageros, evitando complexidade desnecessária que aumentaria o tempo de processamento. Foge do escopo deste trabalho detalhar cada um desses parâmetros, mas mais detalhes podem ser encontrados no apêndice C.2 (que mostra os arquivos de configuração e, portanto, os parâmetros utilizados nas codificações), em artigos que descrevem o padrão H.264 SVC (SCHWARZ et al., 2007) e no manual de utilização do JSVM (JSVM, 2008).

Apesar de não ser um objetivo direto deste trabalho, é interessante observar o tempo de duração do processo de codificação. Em média, para um vídeo de 14 segundos, o tempo de codificação foi: 1 hora para a codificação Temporal, 4 horas para a codificação Espacial e 10 horas para a codificação Qualidade. Este tempo é apenas uma estimativa média da duração das codificações, que foram feitas em computadores com configurações variadas (mas equivalentes): algumas máquinas com processadores Pentium 4 3.0 GHz e 1GB de memória e outras com configurações similares. Este longo tempo de duração para vídeos com duração bastante curta é explicado pela complexidade do padrão H.264 SVC e pelas características do JSVM, que é implementado em software e não é otimizado em relação à velocidade de codificação (como a maioria dos codificadores são), justamente por ser uma implementação de referência.

A diferença entre o tempo de codificação das 3 configurações ocorre, principalmente, devido à dimensão espacial das camadas. Na configuração Temporal, a codificação é feita em apenas uma camada e as 3 camadas temporais são posteriormente extraídas desta camada (mais alguns detalhes sobre o modo de operação do JSVM podem ser vistos no apêndice C.2). Ou seja, o vídeo é codificado com apenas uma camada mas com suporte para escalabilidade temporal, portanto camadas temporais podem ser extraídas após a codificação do vídeo. Esta camada da configuração Temporal tem resolução 4CIF, mas, por ser apenas uma camada, o tempo de codificação é bastante menor do que o das outras configurações. Já as configurações Espacial e Qualidade são codificadas com 3 camadas, com a diferença de que as 3 camadas da Qualidade possuem resolução espacial 4CIF, enquanto as camadas da Espacial possuem resolução QCIF, CIF e 4CIF. É esta diferença das resoluções que causa a diferença do tempo de codificação entre essas configurações. A predição de movimento realizada entre as camadas (predição *inter-layer*) também provoca aumento no tempo de codificação das configurações Espacial e Qualidade (a Temporal não possui predição *inter-layer*, já que é codificada em apenas uma camada).

A terceira etapa do uso do JSVM é a extração das camadas a partir do vídeo codificado (da *bitstream*), como é visto no item 3 da figura 3.9. A extração é um processo simples e rápido, onde o extrator recebe o vídeo codificado e parâmetros informando qual camada deve ser extraída. Cada camada extraída gera uma nova *bitstream*, que contém a camada solicitada e todas as camadas inferiores. Ou seja, a extração da camada 2 gera um arquivo com a camada 1 e a camada 2 integradas; a extração da camada 3 gera um arquivo com a camada 1, 2 e 3; e assim por diante. Como a codificação gera uma *bitstream* com todas as camadas, o extrator é sempre executado 3 vezes, uma para a extração de cada camada.

A quarta etapa é a decodificação de cada uma das 3 camadas extraídas na etapa anterior. A decodificação é mais simples que a codificação, mas não chega a ser mais simples

que o processo de extração, e é a etapa responsável pela transformação da *bitstream* de cada camada em um vídeo descompactado no formato YUV. Após a decodificação, cada vídeo é processado pelo aplicativo *lyuv* para remoção dos quadros adicionais que foram inseridos antes da codificação. Os vídeos gerados após esta etapa correspondem às PVSs utilizadas nas avaliações.

A quinta e última etapa do processo é a análise do PSNR e taxa de codificação dos vídeos, o que é feito através de um aplicativo existente no JSVM. Cada PVSs é comparada com o seu SRC para verificação do PSNR, e a taxa é calculada através da *bitstream* da PVS, ou seja, do mesmo arquivo obtido após a extração da camada que gerou a PVS que está sendo analisada. É importante mencionar que o PSNR usado nos cálculos deste trabalho é o PSNR médio de todos os quadros do vídeo para o seu componente Y (luminância), chamado PSNR-Y.

O processo ilustrado na figura 3.9 é para a codificação de apenas um SRC utilizando apenas uma configuração de codificação. Como foram definidas 3 configurações de codificação, o processo foi realizado 3 vezes para cada SRC. Na prática, o processo foi realizado mais vezes, pois, para encontrar os parâmetros adequados (principalmente os parâmetros de quantização) para que os vídeos respeitassem as definições feitas (ver seção 3.2.1), cada SRC teve que ser codificado diversas vezes utilizando a mesma configuração de codificação.

Apesar de o PSNR estar fixo em 38dB (maioria das camadas) para todos os vídeos, a taxa de cada camada teve grandes variações. Por outro lado, se a taxa fosse fixada entre os diferentes vídeos, o valor do PSNR que sofreria variações. Como já comentado na seção 3.2.1, em função de o objetivo principal deste trabalho ser analisar a instabilidade e fazer esta análise para cada um dos conceitos de escalabilidade individualmente, foi decidido que a comparação seria mais justa caso os valores de PSNR estivessem fixos, e não as taxas. As figuras 3.10 e 3.11 mostram os valores das taxas de codificação e do PSNR de cada camada para todos os SRCs utilizados para a avaliação, e a figura 3.12 mostra os mesmos gráficos para os SRCs utilizados para treinamento. Nestes gráficos é interessante observar a proximidade entre os valores das taxas de cada camada, apesar de não serem valores idênticos (o que seria ideal), e a proximidade que os valores de PSNR têm do valor 38 dB (garantindo o erro máximo de 1,5% conforme estipulado). Na tabela 3.10 são exibidos os valores exatos das taxas e PSNR de cada camada para todos SRCs e HRCs, onde a coluna "C." indica a configuração de codificação e as colunas "C1", "C2" e "C3" indicam as camadas 1, 2 e 3, respectivamente..

Como já comentado na seção 3.2.1, a taxa de codificação das camadas de vídeo foi controlada entre as configurações de codificação aplicadas à um mesmo SRC, mas não foram controladas entre os diversos SRCs. Ou seja, as taxas tiveram grandes variações entre os SRCs para se atingir os objetivos especificados na seção 3.2.1. Esta variação pode ser vista nos gráficos das figuras 3.10, 3.11 e 3.12, e também na tabela 3.10. Como exemplo, pode-se observar a taxa das camadas da configuração T para o vídeo "westwindeasy", que são 235 kbit/s, 377 kbit/s e 907 kbit/s, enquanto as mesmas taxas para as camadas do vídeo "rushfieldcuts" são 916 kbit/s, 1665 kbit/s e 3531 kbit/s.

Em relação às taxas de codificação dos vídeos, também pode-se observar duas exceções que foram identificadas nos resultados das avaliações. Essas exceções são os vídeos "ed1" e "speedbag", que são os dois vídeos cuja configuração Q apresentou taxas mais afastadas das taxas das outras configurações. Além disso, eles foram os dois vídeos que apresentaram menores taxas de codificação, apesar de também estarem de acordo com as especificações propostas. Estas particularidades afetam os resultados das avaliações e

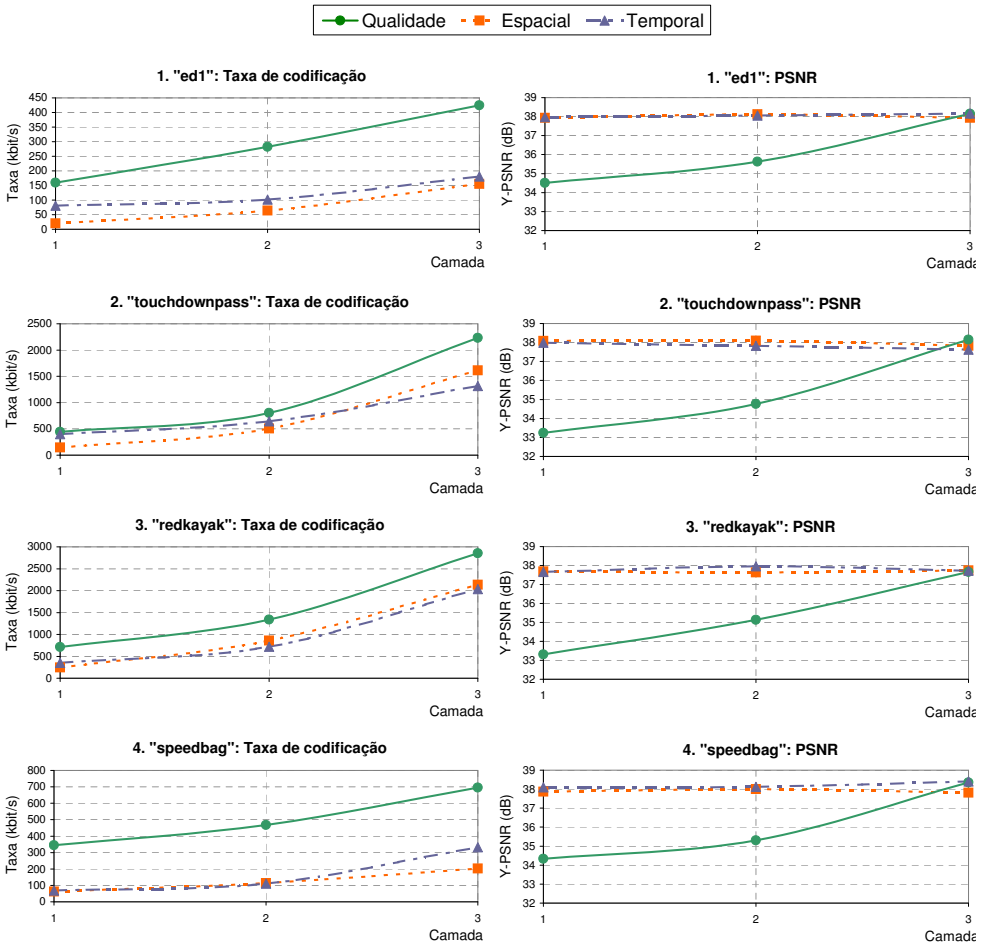


Figura 3.10: Gráficos do PSNR e taxas de codificação para os primeiros 4 SRCs utilizados na avaliação.

serão comentadas novamente na apresentação desses resultados no capítulo 4.

O apêndice C.2 mostra alguns exemplos de arquivos de configuração criados para utilização do JSVM, juntamente com os parâmetros de codificação de alguns vídeos e algumas outras informações técnicas de utilização do JSVM.

3.3.5 Simulação da instabilidade

Após a codificação, a segunda etapa para aplicação dos HRCs é a simulação da instabilidade. Os padrões de instabilidade já foram especificados na seção 3.2.2 e foram aplicados com o uso do aplicativo *lyuv*. Este aplicativo recebe como entrada os vídeos que serão utilizados e um arquivo de configuração que especifica em que momentos ocorrem as variações de camada. Os vídeos de entrada são então dispostos em um arquivo de saída conforme o padrão de instabilidade especificado. Por exemplo, para a aplicação do padrão $p4$ entre as camadas 2 e 3 da configuração T do SRC "aspen" (ver figura 3.5), é passado o vídeo da camada 2 e o vídeo da camada 3 da configuração T deste SRC para o aplicativo *lyuv* e eles serão organizados da seguinte maneira no arquivo de saída: 3 segundos do vídeo da camada 2, 2 segundos da camada 3, 3 segundos da camada 2, 2 segundos da camada 3 e 4 segundos da camada 2. Os HRCs que utilizam o padrão estável ($p0$) não necessitam deste procedimento, pois não possuem variações de camadas.

Durante a simulação da instabilidade, os vídeos também foram normalizados para a

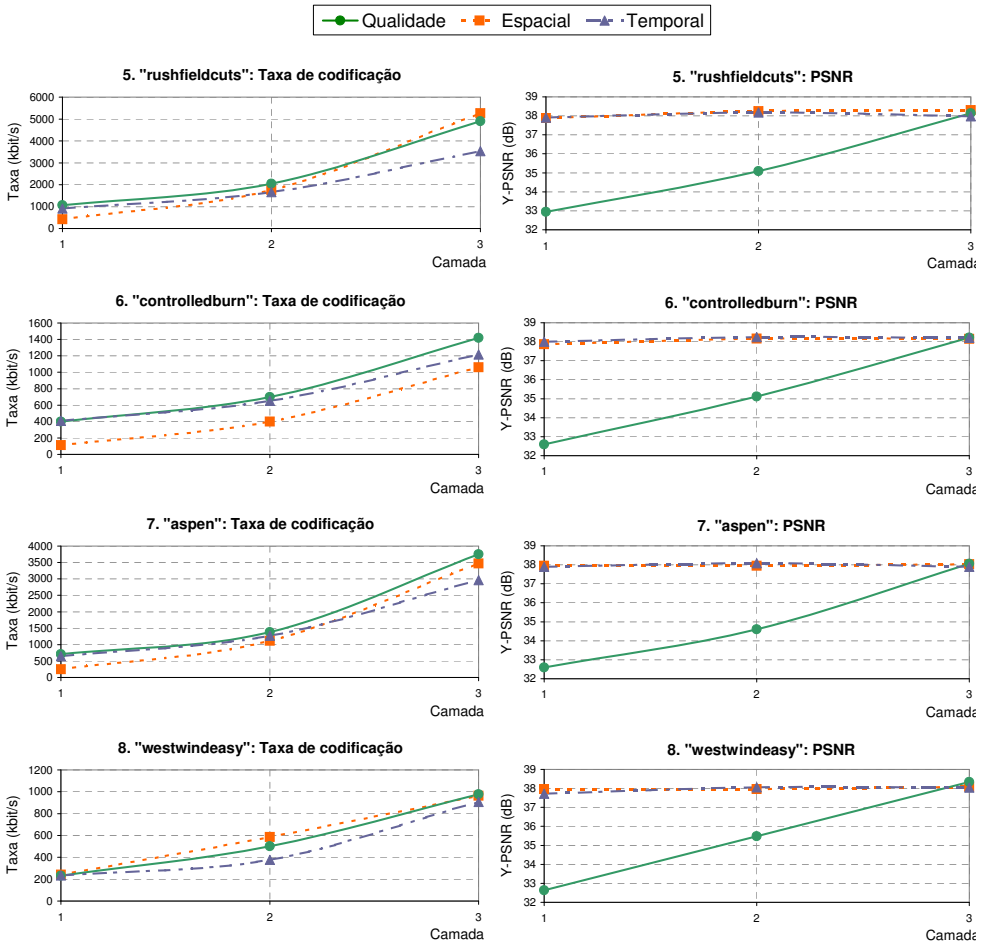


Figura 3.11: Gráficos do PSNR e taxas de codificação para os últimos 4 SRCs utilizados na avaliação.

resolução 4CIF e 30 fps. Aqueles que apresentavam resolução CIF ou QCIF foram ampliados utilizando replicação de pixels e aqueles que apresentavam um número menor de quadros por segundo foram ampliados para 30 fps com a repetição de quadros. O aumento da resolução espacial foi feito para manter um ambiente mais realista e padronizar a avaliação. Se as camadas com resolução QCIF e CIF não fossem ampliadas, a resolução do vídeo iria mudar durante a variação de camadas, mas dificilmente um aplicativo real varia a resolução do vídeo ao longo de sua exibição, ele é reduzido ou ampliado para a resolução que está sendo usada. A replicação de pixels utilizada consiste em replicar um pixel 16 vezes (quando a resolução inicial é QCIF) ou 4 vezes (quando a resolução inicial é CIF). A figura 3.13 exemplifica este processo de ampliação. Já o aumento da resolução temporal não altera em nada a maneira como os vídeos seriam exibidos caso estivessem com um número menor de quadros por segundo, apenas aumenta o número de quadros do arquivo. Na ampliação de um vídeo com 7,5 fps para 30 fps, por exemplo, cada quadro é apenas replicado 4 vezes.

Sistemas que exibem vídeo normalmente utilizam técnicas mais avançadas de interpolação de pixels quando é necessário o redimensionamento das imagens ou interpolação temporal para aumento do número de quadros por segundo. Neste trabalho, foram utilizadas técnicas mais simples, a replicação de pixels e de quadros, para que este processo influenciasse o mínimo possível nos resultados das avaliações.

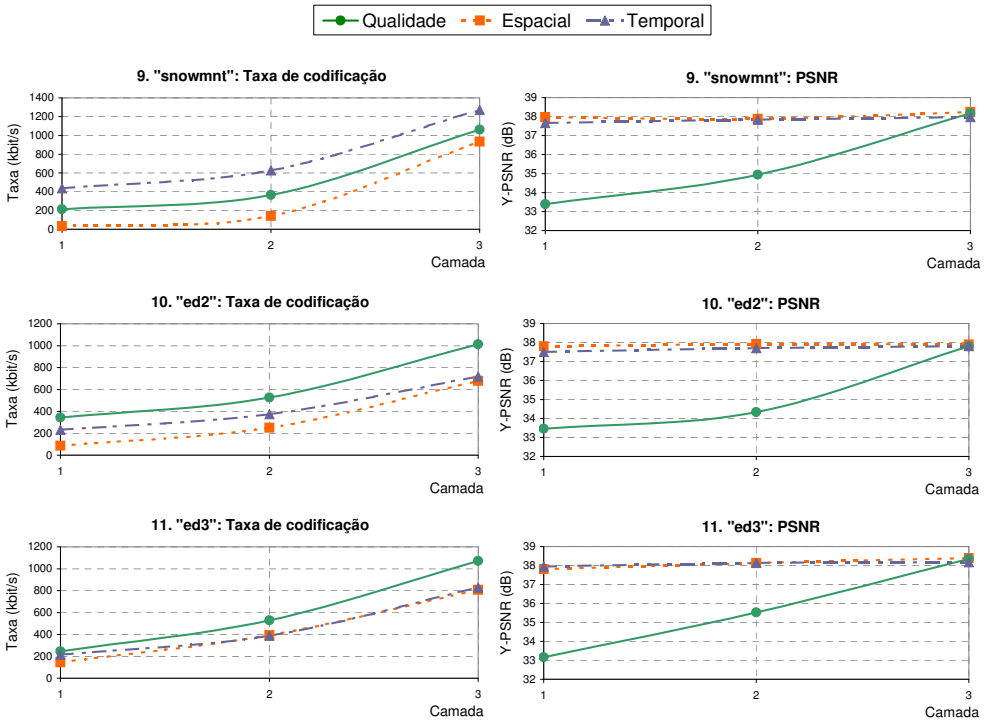


Figura 3.12: Gráficos do PSNR e taxas de codificação para todos os SRCs utilizados para treinamento.

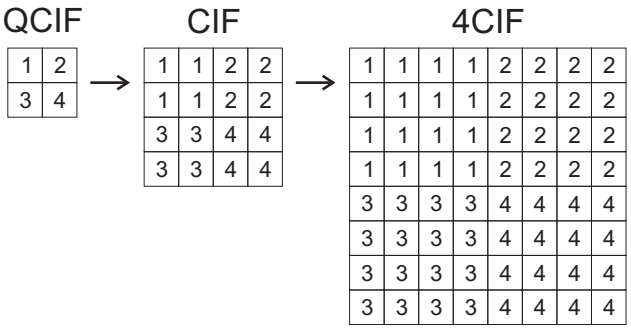


Figura 3.13: Exemplo da replicação de pixels para ampliação da resolução espacial.

As alterações na qualidade dos vídeos provocadas pelas variações das camadas normalmente são vistas pelos usuários como variações na blocagem, no borramento ou na suavidade do fluxo do vídeo, como já comentado na seção 3.2.2. A figura 3.14 mostra um exemplo de como ficaram as camadas das configurações Qualidade e Espacial para o SRC "redkayak". Esta figura exemplifica a diferença de qualidade entre as camadas da configuração Qualidade e a diferença de resolução e qualidade entre as camadas da configuração Espacial. Todas as imagens exibidas representam exatamente o mesmo quadro, mas em camadas diferentes. Na figura, "C1" indica a primeira camada, "C2" a segunda camada e "C3" a terceira camada. A coluna da esquerda mostra as três camadas da configuração Qualidade e a coluna da direita mostra as três camadas da configuração Espacial. A coluna central exibe a relação entre a resolução das camadas na configuração Espacial. As alterações nas camadas da configuração Temporal não são exemplificadas pois ocorrem apenas eixo temporal.

A aplicação dos padrões de instabilidade encerra a fase de processamento dos vídeos.

Tabela 3.10: Valores de PSNR e taxa de codificação para todos os SRCs.

			PSNR (dB)			Taxa (kbit/s)		
ID	**SRC**	**C.**	**C1**	**C2**	**C3**	**C1**	**C2**	**C3**
1	ed1	T	37,97	38,06	38,16	80,94	101,88	180,99
		E	37,94	38,12	37,94	19,92	64,38	155,58
		Q	34,51	35,62	38,15	159,81	282,75	424,71
2	touchdownpass	T	37,99	37,82	37,62	398,28	641,25	1316,85
		E	38,08	38,10	37,83	147,75	506,16	1612,86
		Q	33,24	34,77	38,14	441,72	805,20	2233,02
3	redkayak	T	37,66	37,96	37,73	349,05	718,80	2034,39
		E	37,68	37,63	37,74	245,31	860,94	2135,73
		Q	33,32	35,13	37,65	715,74	1340,13	2850,93
4	speedbag	T	38,09	38,13	38,41	67,86	110,67	332,91
		E	37,87	38,00	37,82	62,40	113,28	202,86
		Q	34,33	35,30	38,36	345,30	467,49	694,80
5	rushfieldcuts	T	37,90	38,18	37,98	916,98	1665,54	3531,09
		E	37,88	38,24	38,30	424,05	1758,51	5267,94
		Q	32,95	35,08	38,14	1068,42	2044,14	4898,58
6	controlledburn	T	38,01	38,24	38,21	409,98	651,02	1215,15
		E	37,87	38,17	38,15	112,23	399,15	1059,93
		Q	32,59	35,11	38,22	399,81	698,04	1420,59
7	aspen	T	37,89	38,09	37,90	650,79	1271,07	2960,64
		E	37,94	37,95	38,02	247,38	1118,01	3468,66
		Q	32,59	34,60	38,05	708,45	1379,76	3750,60
8	westwindeasy	T	37,72	38,05	38,05	235,29	377,37	907,29
		E	37,96	37,96	38,08	241,89	585,06	963,78
		Q	32,64	35,48	38,33	232,44	501,51	976,80
9	snowmnt	T	37,67	37,84	37,97	435,84	627,63	1271,31
		E	37,97	37,89	38,24	31,41	140,82	933,45
		Q	33,38	34,94	38,18	212,51	365,76	1060,70
10	ed2	T	37,51	37,71	37,81	230,43	374,49	718,17
		E	37,80	37,92	37,89	87,33	251,16	679,32
		Q	33,45	34,33	37,80	343,80	525,57	1013,97
11	ed3	T	37,94	38,14	38,17	214,77	387,78	827,61
		E	37,80	38,13	38,39	144,60	394,02	808,11
		Q	33,167	35,52	38,34	244,71	528,87	1070,64

Depois desta etapa, as 152 PVSs da avaliação e as demais PVSs do treinamento estavam geradas e prontas para utilização. Em relação às PVSs utilizadas para treinamento, os 3 SRCs foram codificados com todos HRCs, assim como os vídeos da avaliação, mas nem todas PVSs geradas foram utilizadas. Foi selecionado um subconjunto destas PVSs, como será comentado na seção 3.4.

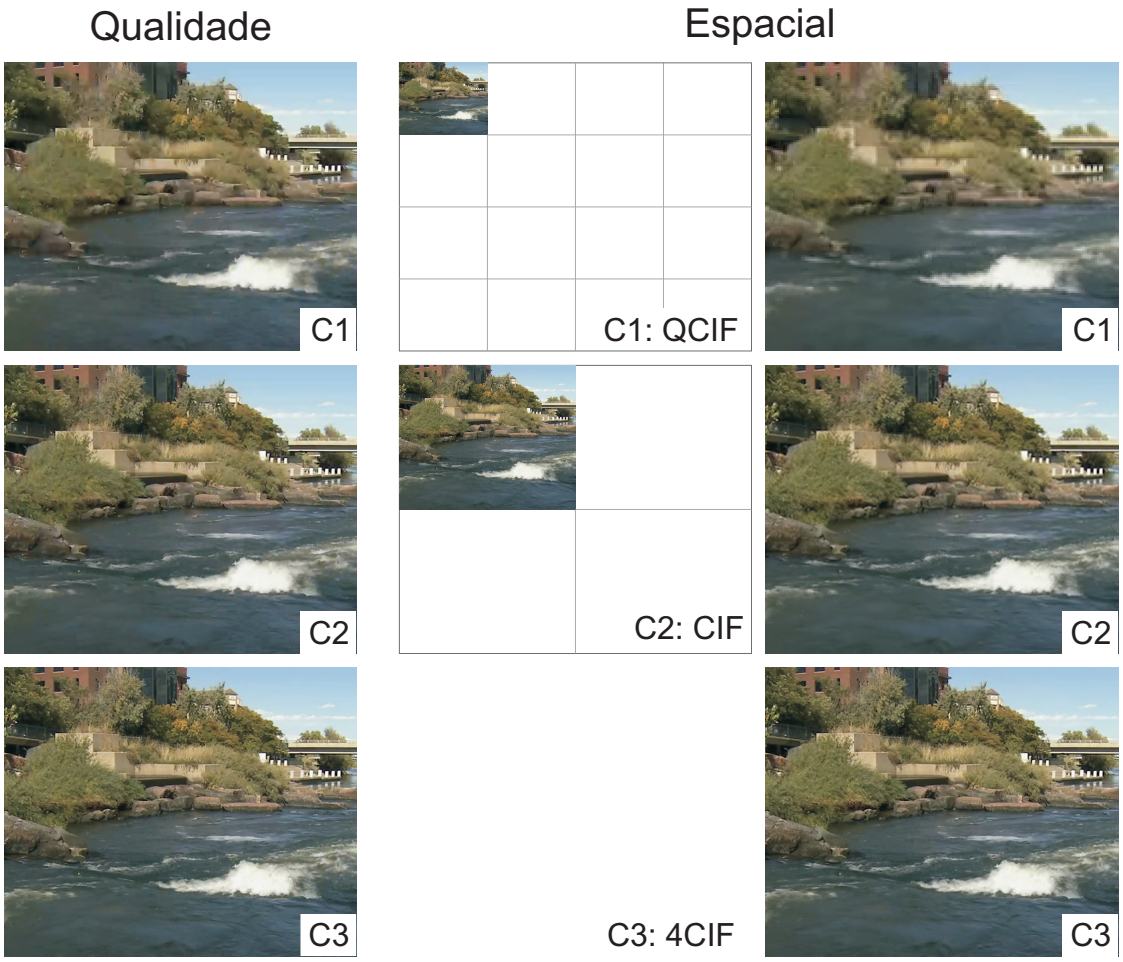


Figura 3.14: Exemplo de quadros codificados do SRC “redkayak”.

3.4 Execução das avaliações subjetivas

O processo de avaliação foi realizado em um ambiente de acordo com as recomendações da norma ITU-T P.910. Entre as principais recomendações para as condições de visualização, estão os itens da tabela 3.11. Apesar de estabelecer estes parâmetros, a norma permite certa flexibilidade em diversos deles. Exemplificando, a norma comenta que a distância de visualização não deve ser selecionada somente de acordo com o tamanho da tela como especificado (que tem relação com o parâmetro H exibido na tabela), mas também de acordo com o tipo de tela, tipo da aplicação e os objetivos do experimento.

Foi usado um luxímetro digital ICEL Manaus (LD-550) para verificação da luz ambiente e, para calibração do monitor, foram utilizadas as ferramentas Video Essentials DVD, que consistem em um conjunto de testes executados para calibrar parâmetros do monitor, como o contraste, brilho e cores. O Video Essentials DVD é uma das ferramentas selecionadas pelo VQEG na execução de suas avaliações (VQEG, 2008a). Os vídeos foram exibidos em um monitor LCD de 19"(LG L1952H) utilizando a resolução 1024x768. Esta resolução foi escolhida devido aos objetivos dos testes, pois permite uma melhor percepção das mudanças que ocorrem nas simulações de variação das camadas dos vídeos de resolução 4CIF. Resoluções maiores reduziriam a área de apresentação dos vídeos na tela, dificultando a percepção dos detalhes. A tabela 3.12 sumariza as condições do ambiente e as especificações do monitor que foi utilizado. A utilização das recomen-

Tabela 3.11: Condições do ambiente segundo a norma P.910.

Parâmetro	**Valores**
Distância de visualização[2]	1-8 H
Luminância máxima da tela	100-200 cd/m
Razão entre luminância da tela inativa e luminância máxima	$\leq$ 0,05
Razão entre luminância da tela quando exibindo uma tela preta em uma sala completamente escura e luminância máxima de um ponto branco	$\leq$ 0,1
Razão entre luminância do ambiente atrás da tela e luminância máxima dos vídeos.	$\leq$ 0,2
Cromaticidade do fundo	D_{65}
Iluminação da sala	$\leq$ 20 lux

dações da norma P.910 e especificação dos parâmetros utilizados no trabalho auxiliam a validação dos resultados e permitem que estes possam ser comparados com resultados de outras avaliações.

Tabela 3.12: Condições do ambiente e especificações do monitor utilizado nas avaliações.

Parâmetro	**Valores**
Distância de visualização	3H ($\pm$ 90 cm)
Tamanho da tela	19" diagonal
Resolução da tela	1024x768
Dot pitch	0.294
Taxa de atualização	60 Hz
Razão de contraste	40-50
Tempo de resposta	8ms
Método de calibração	Video Essentials DVD
Temperatura de cor	6500K

22 avaliadores participaram das avaliações, incluindo 19 homens e 3 mulheres, a maioria na faixa etária entre 18-35 anos, dos quais 27% possuem entre 21-23 anos e 27% entre 27-35. A faixa etária e o gênero de todos os avaliadores podem ser vistos no apêndice D.2. Nenhum dos avaliadores havia participado de avaliações de qualidade de vídeo antes e a maioria trabalha na área da computação. A quantidade de avaliadores utilizados (22) é superior ao mínimo recomendado (15) para obtenção de resultados válidos segundo a norma BT.500.

Antes da avaliação, os avaliadores foram submetidos a dois testes de visão, um para acuidade e um para visão de cores, como é recomendado na norma P.910. Para visão

[2]H indica a altura do vídeo na tela (ou a altura da tela, se o vídeo é exibido em tela cheia)

de cores, foram utilizadas as placas de Ishihara e o teste foi aplicado com a visualização das placas no próprio computador no qual foram executadas as avaliações. Já o teste de acuidade foi realizado com uma cartela *Rosenbaum Pocket Screener*, uma versão reduzida dos painéis de teste de visão normalmente utilizados profissionalmente para exames de visão. A figura 3.15 mostra um exemplo de 4 placas de Ishihara e da cartela para teste de acuidade. Nenhum avaliador apresentou problemas na visão durante esses testes.

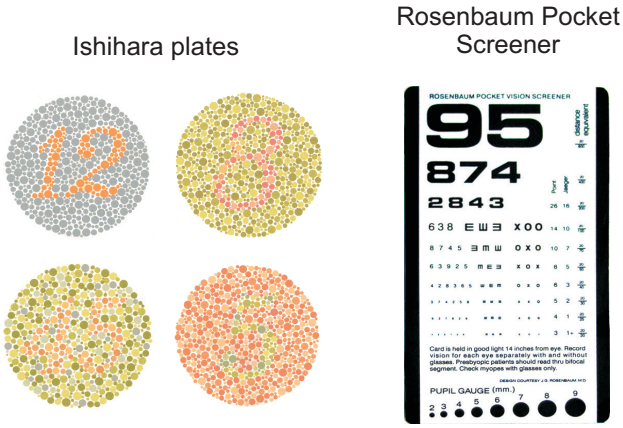


Figura 3.15: Exemplo dos documentos utilizados para teste visão.

A distância de visualização foi fixada em 3H (entre 80 cm e 90 cm). Todos avaliadores foram instruídos através de um documento impresso (que representa um conjunto de instruções padrão) e de explicações orais para resolução de dúvidas ainda existentes. O texto do documento impresso encontra-se no apêndice D.1. Foi solicitado claramente a cada avaliador para tentar perceber os detalhes dos vídeos, principalmente em relação às variações existentes ao longo da visualização de um mesmo vídeo, e para atribuir uma nota representando sua interpretação da qualidade daquele vídeo.

Em uma avaliação de vídeos que apresentam variações de qualidade ao longo de sua exibição normalmente são utilizadas metodologias de votação contínua (como a metodologia DSCQS, descrita na norma BT.500, por exemplo), onde diversos votos são atribuídos durante toda a execução do vídeo. Neste trabalho utilizou-se um conceito chamado *service acceptance*, onde são usados vídeos com duração mais longa que o tradicional (que é de 8-10 segundos), mas a votação não é contínua: é atribuído apenas 1 voto ao final da sua exibição, que corresponde à "qualidade do serviço" obtido (BARONCINI, 2006). O conceito foi utilizado pois optamos por não ter medidas de qualidade para os momentos exatos das variações de qualidade, mas sim uma avaliação do serviço como um todo.

O processo de avaliação foi feito com uso da metodologia ACR com HRR (*Hidden Reference Removal*), já descrita na seção 2.4.1.1. Esta metodologia é de estímulo único, onde os vídeos são apresentados em sequência e alternados com intervalos para votação. HRR indica o uso de uma referência "escondida", sem que o avaliador saiba que o vídeo que está avaliando é a referência. É utilizada uma referência para cada SRC, que consiste no vídeo original deste SRC. A nota desta referência é utilizada como âncora superior para normalização dos votos, como será descrito na seção 4.1.

A votação é feita em uma escala de 11 valores, de 0 a 10, identificadas por 5 marcadores: ótimo, bom, regular, ruim e péssimo. Também foi dada a opção ao avaliador de assistir novamente o vídeo (apenas mais uma vez) caso, por algum motivo, tenha perdido

a primeira exibição. A ordem de apresentação dos vídeos é gerada de forma aleatória para cada avaliador com o objetivo de minimizar a influência do contexto nos resultados.

A execução das avaliações foi realizada inteiramente através de um computador. Foi desenvolvida uma aplicação em C/C++ para exibição dos vídeos e interatividade com o avaliador, especialmente para obtenção das notas que os avaliadores atribuem aos vídeos. Esta aplicação possui maneiras de verificar se os vídeos foram exibidos de forma adequada, ou seja, se os quadros foram exibidos no tempo em que deveriam ser exibidos, o que é importante especialmente pelo motivo de as avaliações incluírem vídeos com alterações na taxa de quadros por segundo. Mais alguns detalhes sobre esta aplicação podem ser encontrados no apêndice B. Como já comentado na seção 3.3.5, todos os vídeos foram reproduzidos com resolução 4CIF usando 30 fps.

A figura 3.16 mostra um exemplo de duas telas do aplicativo usado para execução das avaliações subjetivas: a da esquerda mostra uma PVS durante sua exibição e a da direita mostra a tela de votação. A tela de votação apresenta um título no topo, onde é indicado o número do vídeo visualizado (que indica a posição do vídeo entre as 152 PVSs). A escala de votação de 11 valores é exibida ao centro e, na parte inferior, estão os botões para repetir o último vídeo visto (apenas uma vez para cada PVSs) e para prosseguir a avaliação (só habilitado após uma nota ser atribuída ao vídeo atual). Os vídeos foram exibidos com uma borda preta de 1 pixel de largura e com fundo de cor R=G=B=128, com a aplicação permanecendo sempre em tela cheia.

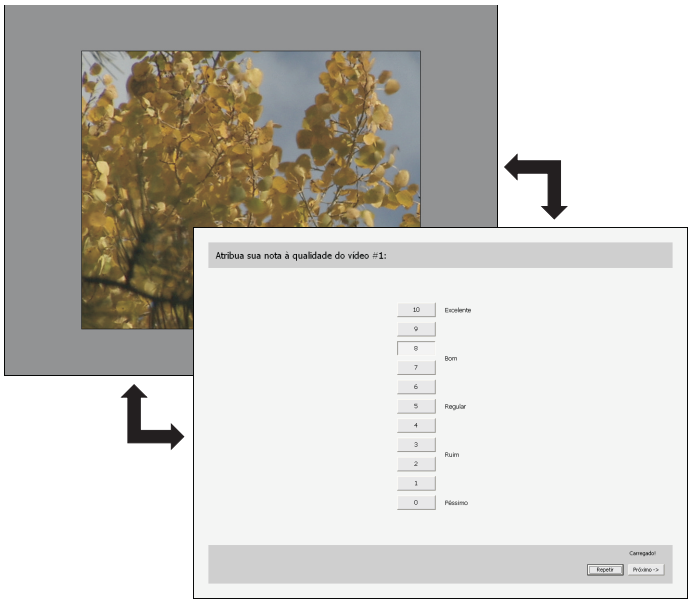


Figura 3.16: Exemplo de telas do aplicativo para execução das avaliações subjetivas.

Após o processo inicial, que inclui os testes de visão e o processo de instrução dos avaliadores, era executada a fase de treinamento. Esta fase é utilizada para que os avaliadores possam se familiarizar com o aplicativo utilizado e com as variações (HRCs) às quais os vídeos foram submetidos. Como comentado na seção 3.3, foram selecionados 3 vídeos específicos para esta fase de treinamento, que não foram utilizados na avaliação efetiva. Eles foram codificados com todas as 18 HRCs, assim como o restante dos vídeos, mas não foram todas as PVSs geradas que foram utilizadas no treinamento. Apenas 15 PVSs foram selecionadas, para restringir o tempo de duração desta fase. Essas 15 PVSs foram selecionadas de acordo com algumas premissas: (i) incluir o maior número de HRCs possível; (ii) dividir as 15 PVSs justamente pelos 3 SRCs (5 PVSs de cada

SRC, idealmente); (iii) dividir igualmente o número de HRCs utilizando cada uma das 3 configurações de codificação (T, Q e E); e (iv) selecionar alguns casos considerados mais extremos (qualidade muito boa ou muito ruim), através de visualizações prévias dos vídeos.

As avaliações iniciavam logo após a fase de treinamento, onde as 152 PVSs eram exibidas em um período de cerca de 1 hora, com um intervalo opcional na metade da avaliação. As sessões foram realizadas para cada avaliador individualmente. A figura 3.17 ilustra a ordem em que as fases da avaliação foram executadas.

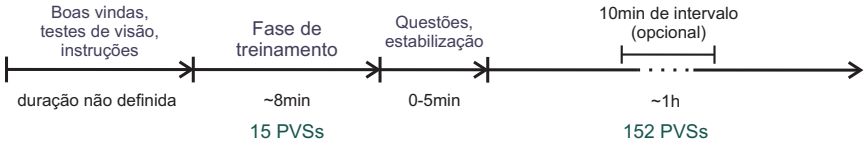


Figura 3.17: Fases das avaliações de qualidade.

Ao final do processo, foi entregue um questionário bastante informal aos avaliadores para obter a opinião dos mesmos sobre questões como a duração dos vídeos e da avaliação, a qualidade geral e o conteúdo dos vídeos e a dificuldade em se perceber as variações ao longo de um mesmo vídeo. As questões incluídas neste questionário podem ser vistas no apêndice D.2 e as respostas obtidas são apresentadas no apêndice D.4.

4 APRESENTAÇÃO DOS RESULTADOS

Este capítulo mostra os resultados obtidos com as avaliações de qualidade realizadas. Inicialmente, a seção 4.1 mostra o processamento aplicado sobre os dados antes da análise, enquanto as seções 4.2 e 4.3 mostram os votos gerais de todos os avaliadores para todos HRCs e SRCs e a análise dos valores médios desses votos. As próximas seções analisam os dados de acordo com os três principais objetivos do trabalho: em relação à instabilidade (seção 4.4), em relação à taxa de codificação (seção 4.5) e em relação aos métodos de escalabilidade (seção 4.6).

4.1 Análise inicial

O primeiro passo para a análise dos resultados é a normalização dos votos conforme o voto dado ao vídeo de referência. A normalização é feita para cada avaliador e para cada PVSs, através de um método semelhante ao utilizado pelo grupo de multimídia do VQEG em seu último plano de testes (VQEG, 2008a). A equação 4.1 mostra o método utilizado para normalizar os votos. Nesta equação (e nas próximas que serão apresentadas), a variável V indica o voto atribuído ao vídeo indicado pelo índice da variável, que pode ser: a, o avaliador que atribuiu o voto; h, o HRC; s, o SRC; p, a PVS (também representada por sh); r, a referência. Assim, e V_{ash} indica o voto atribuído pelo avaliador a ao SRC s e HRC h, o que pode também ser entendido como o voto do avaliador a para a PVS sh. A variável V_{asr} representa o voto do avaliador a para o vídeo de referência do SRC s e V' representa o voto normalizado.

$$V'_{ash} = \begin{cases} 1 & \text{se } V_{ash} > V_{asr} \\ V_{ash}/V_{asr} & \text{caso contrário} \end{cases} \tag{4.1}$$

A equação 4.1 é aplicada para os votos de todas PVSs de todos avaliadores antes de qualquer outra análise e os votos, originalmente no intervalo entre 0 e 10, estarão no intervalo $0 \leq V_{a,s,h} \leq 1$. Se o voto para determinada PVS for maior do que o voto dado ao vídeo de referência do mesmo SRC, o voto normalizado será 1 (apesar deste caso não ter acontecido nas avaliações). Duas outras formas semelhantes de normalização também foram aplicadas para comparação, sendo uma delas chamada DMOS (*Differential Mean Opinion Score*) (VQEG, 2008a), que é obtida através da equação 4.2. Na equação, V_{ash} corresponde ao voto atribuído à determinada PVS e V_{asr} o voto atribuído ao vídeo de referência do mesmo SRC utilizado para codificar esta PVS. A soma do valor 10 ao final é realizada pois a escala de valores utilizada está entre 1 e 10. A outra forma de normalização utilizada foi a equação 4.3 (MONTEIRO; NUNES, 2007), onde as variáveis usadas são as mesmas já comentadas. As diferenças nos resultados obtidos com esses

diferentes métodos foram muito pequenas, portanto não justificavam a troca da forma de normalização definida na equação 4.1.

$$V'_{ash} = V_{ash} - V_{asr} + 10 \tag{4.2}$$

$$V'_{ash} = \frac{V_{ash}}{V_{asr}} \tag{4.3}$$

Após a normalização e antes da análise estatística, foi realizada uma validação dos votos de acordo com a especificação da norma BT.500. Para cada avaliador, o processo verifica o número de votos que estão muito acima (serão chamados de P) ou muito abaixo (chamados Q) da média e também a relação entre esses dois valores. Este processo é realizado para cada PVS. Se 5% ou mais dos votos estão muito acima ou muito abaixo da média *e* se a relação entre os dois valores é próxima de 1 (o que indica que ambos os valores P e Q são muito altos, e não apenas um deles), o avaliador deve ser descartado.

As equações 4.4 e 4.5 mostram como são calculados os valores de P e Q. Nas equações, β_{2sh} representa o coeficiente de kurtosis, que indica se a distribuição dos votos é normal ou não (sua equação pode ser encontrada na norma BT.500). Entre as variáveis ainda não apresentadas, $\overline{V}_{sh}$ representa a média dos votos de todos avaliadores para o SRC s e HRC h (ou PVS sh) e S_{sh} representa o desvio padrão para os votos deste mesmo conjunto.

$$\text{se } 2 \leq \beta_{2sh} \leq 4\text{: } \begin{cases} \text{se } V_{ash} \geq \overline{V}_{sh} + 2S_{sh} & \text{então } P_a = P_a + 1 \\ \text{se } V_{ash} \leq \overline{V}_{sh} - 2S_{sh} & \text{então } Q_a = Q_a + 1 \end{cases} \tag{4.4}$$

$$\text{senão: } \begin{cases} \text{se } V_{ash} \geq \overline{V}_{sh} + \sqrt{20}S_{sh} & \text{então } P_a = P_a + 1 \\ \text{se } V_{ash} \leq \overline{V}_{sh} - \sqrt{20}S_{sh} & \text{então } Q_a = Q_a + 1 \end{cases} \tag{4.5}$$

Após o cálculo de P e Q, o avaliador a deve ser rejeitado se ambas as condições das equações 4.6 e 4.7 forem satisfeitas. Na equação 4.6, J representa o número total de HRCs (incluindo a referência) e K representa o número total de SRCs.

$$\frac{P_a + Q_a}{J.K} > 0,05 \tag{4.6}$$

$$\left| \frac{P_a - Q_a}{P_a + Q_a} \right| < 0,3 \tag{4.7}$$

O resultado deste processo indicou um avaliador irregular, portanto os resultados que serão apresentados foram analisados para os outros 21 avaliadores, excluindo os votos do avaliador irregular.

Os resultados das avaliações são exibidos em função da medida MOS (*Mean Opinion Score*), que corresponde à média dos votos de todos os avaliadores, como definido na norma BT.500. O MOS pode ser calculado para cada PVS, para cada HRC ou para cada SRC. O MOS de uma PVS é a média dos votos atribuídos por todos avaliadores para determinada PVS e será chamado MOS_p. O MOS de um HRC (MOS_h) é a média dos votos de todos avaliadores e todos SRCs processados pelo HRC alvo, ou seja, os 21 votos para os 8 SRCs que foram processados pelo HRC alvo são somados e o resultado é dividido por 168 (21*8=168). Por fim, o MOS de um SRC (MOS_s) representa a média dos votos de todos avaliadores e todos HRCs que foram usados para processar o SRC alvo. O MOS_h é importante para verificar a qualidade média percebida nos diferentes tipos de

alterações aplicadas aos vídeos (métodos de escalabilidade e padrões de instabilidade), enquanto o MOS_s mostra a qualidade média de cada vídeo. A figura 4.1 mostra um exemplo de como são calculados os valores MOS a partir de 2 SRCs, 3 HRCs e com os votos de 3 avaliadores (cada x representa o voto de um avaliador para uma PVS).

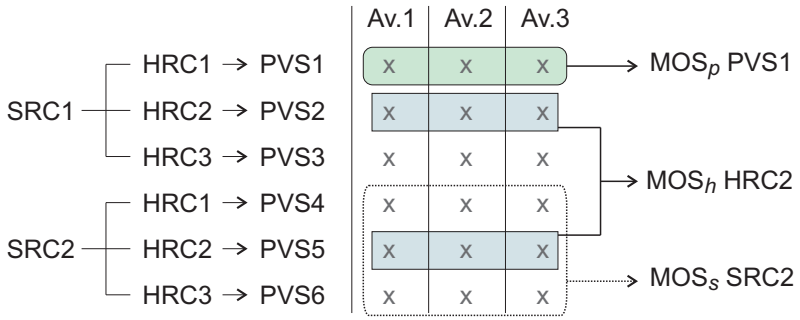


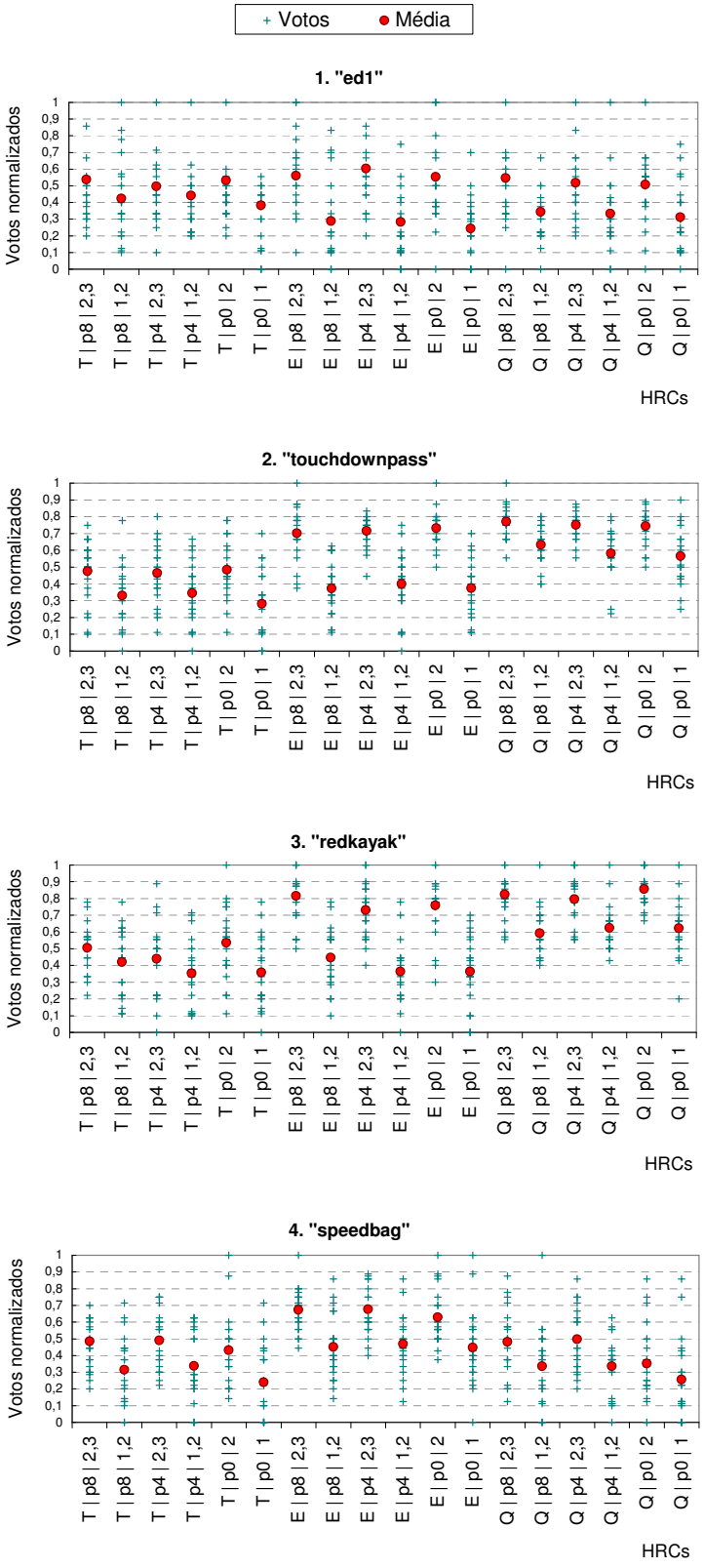
Figura 4.1: Exemplo dos votos utilizados para cálculo dos valores MOS.

Para identificação dos HRCs, será utilizada uma legenda representada da seguinte forma: "configuração | padrão | camadas". Configuração diz respeito à configuração de codificação utilizada (ver seção 3.2.1): Temporal (T), Espacial (E) ou Qualidade (Q). O "padrão" indica qual padrão de instabilidade foi utilizado (ver seção 3.2.2): estável ($p0$), com pouca variação ($p4$) ou com muita variação ($p8$). Por fim, "camadas" indica em que camadas de vídeo o padrão de instabilidade foi aplicado: entre as camadas 1 e 2 ou entre 2 e 3 (para o padrão estável é apresentada somente uma camada, obviamente). Assim, a legenda "T | p8 | 2-3", por exemplo, representa a configuração de codificação Temporal, utilizando o padrão com muita variação de camadas aplicado entre as camadas 2 e 3.

4.2 Apresentação de todos os votos atribuídos

As figuras 4.2 e 4.3 mostram os gráficos para todos os votos (já normalizados) atribuídos para cada SRC, com a média dos votos de cada HRC em destaque. Cada coluna representa um dos 18 HRCs e contém 21 votos, um atribuído por cada avaliador (símbolos +). Para cada HRC também é especificado o valor médio dos 21 votos (círculo vermelho). Na tabela ao lado dos gráficos, são exibidos os valores da média dos votos ("Média"), o intervalo de confiança de 95% ("Int.Conf.") e o desvio padrão ("Desv.P.") para todos HRCs. O intervalo de confiança de 95% (ITU-R, 2002) indica com 95% de precisão que a diferença entre os valores encontrados na avaliação e os valores "reais" (quando calculados com "todos" avaliadores, ou, na prática, com um número enorme de avaliadores) se encontra dentro do intervalo $[M_h - I_h, M_h + I_h]$, onde M_h é a média dos votos e I_h é o intervalo de confiança do HRC h. Um relatório dos votos atribuídos por todos avaliadores para cada uma das PVSs avaliadas pode ser encontrado no apêndice D.5.

Analisando as figuras 4.2 e 4.3, pode ser visto que os HRCs temporais (T), especialmente aqueles com os padrões de instabilidade $p4$ e $p8$, possuem os votos mais dispersos da média do que os outros HRCs. O desvio padrão médio para as PVSs processadas pela configuração Temporal é 0,214, enquanto para a configuração Qualidade é 0,175 e para a configuração Espacial é 0,186. Estes dados podem ser interpretados como uma maior dificuldade dos observadores em avaliar a qualidade de vídeo onde as variações ocorrem no eixo temporal ou então como a existência de opiniões mais diversas sobre estas varia-



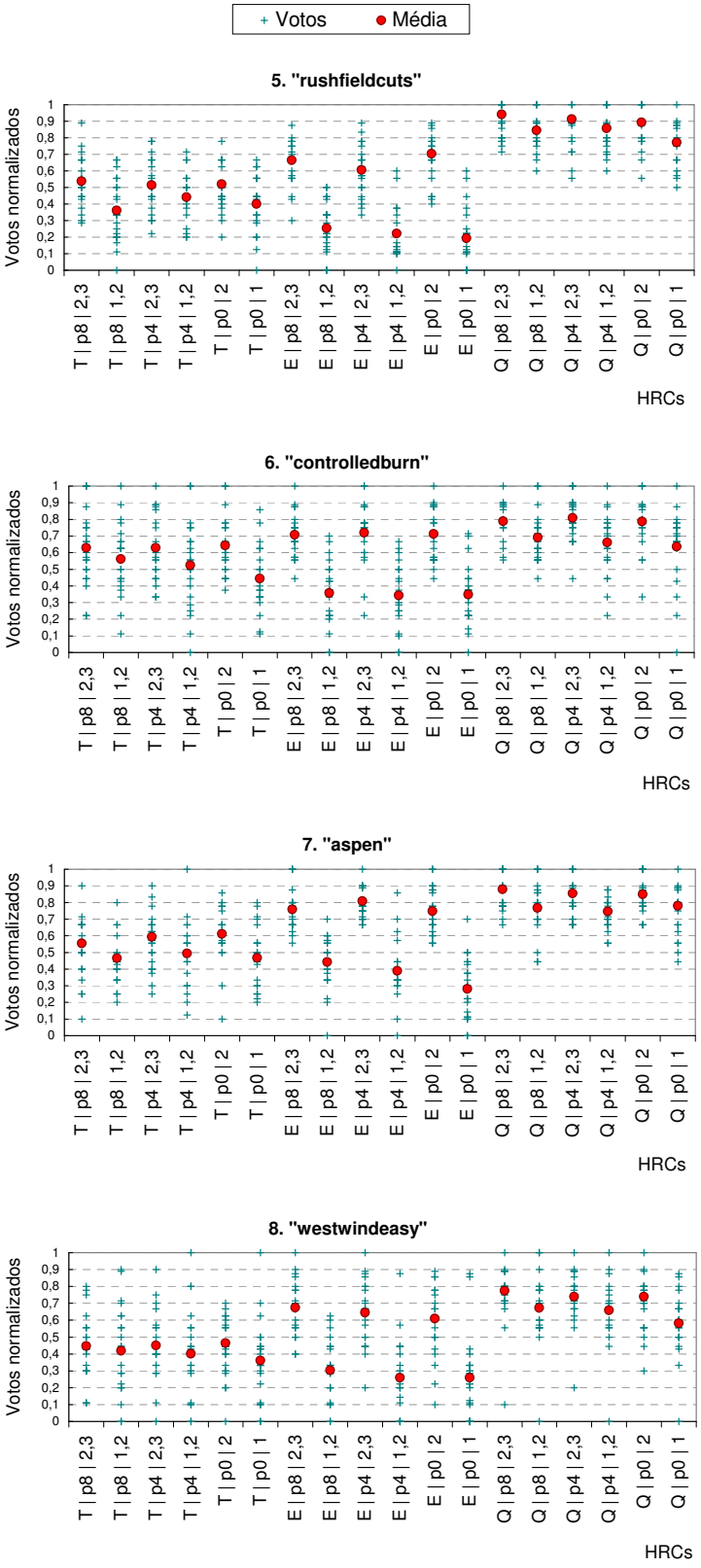
HRC	Média	Int.Conf.	Desv.P.
Q | p0 | 1	0,31	0,09	0,20
Q | p0 | 2	0,51	0,11	0,26
Q | p4 | 1,2	0,33	0,09	0,22
Q | p4 | 2,3	0,52	0,11	0,26
Q | p8 | 1,2	0,34	0,11	0,25
Q | p8 | 2,3	0,55	0,13	0,30
E | p0 | 1	0,24	0,08	0,19
E | p0 | 2	0,55	0,13	0,29
E | p4 | 1,2	0,28	0,12	0,28
E | p4 | 2,3	0,60	0,13	0,29
E | p8 | 1,2	0,29	0,11	0,26
E | p8 | 2,3	0,56	0,10	0,23
T | p0 | 1	0,38	0,13	0,31
T | p0 | 2	0,53	0,13	0,31
T | p4 | 1,2	0,44	0,12	0,28
T | p4 | 2,3	0,50	0,11	0,26
T | p8 | 1,2	0,42	0,11	0,25
T | p8 | 2,3	0,54	0,16	0,37

HRC	Média	Int.Conf.	Desv.P.
Q | p0 | 1	0,57	0,07	0,16
Q | p0 | 2	0,74	0,04	0,10
Q | p4 | 1,2	0,58	0,06	0,15
Q | p4 | 2,3	0,75	0,03	0,08
Q | p8 | 1,2	0,63	0,05	0,12
Q | p8 | 2,3	0,77	0,04	0,10
E | p0 | 1	0,37	0,07	0,17
E | p0 | 2	0,73	0,05	0,11
E | p4 | 1,2	0,40	0,08	0,20
E | p4 | 2,3	0,72	0,04	0,09
E | p8 | 1,2	0,37	0,07	0,16
E | p8 | 2,3	0,70	0,07	0,16
T | p0 | 1	0,28	0,09	0,20
T | p0 | 2	0,48	0,07	0,17
T | p4 | 1,2	0,34	0,08	0,19
T | p4 | 2,3	0,46	0,08	0,18
T | p8 | 1,2	0,33	0,08	0,18
T | p8 | 2,3	0,48	0,08	0,19

HRC	Média	Int.Conf.	Desv.P.
Q | p0 | 1	0,62	0,07	0,17
Q | p0 | 2	0,86	0,05	0,11
Q | p4 | 1,2	0,62	0,06	0,14
Q | p4 | 2,3	0,80	0,06	0,14
Q | p8 | 1,2	0,59	0,06	0,15
Q | p8 | 2,3	0,83	0,07	0,16
E | p0 | 1	0,36	0,10	0,22
E | p0 | 2	0,76	0,09	0,20
E | p4 | 1,2	0,36	0,08	0,18
E | p4 | 2,3	0,73	0,07	0,16
E | p8 | 1,2	0,45	0,08	0,18
E | p8 | 2,3	0,82	0,08	0,18
T | p0 | 1	0,36	0,09	0,21
T | p0 | 2	0,54	0,09	0,22
T | p4 | 1,2	0,35	0,09	0,20
T | p4 | 2,3	0,44	0,09	0,22
T | p8 | 1,2	0,42	0,09	0,20
T | p8 | 2,3	0,50	0,06	0,14

HRC	Média	Int.Conf.	Desv.P.
Q | p0 | 1	0,26	0,10	0,24
Q | p0 | 2	0,35	0,10	0,23
Q | p4 | 1,2	0,34	0,08	0,18
Q | p4 | 2,3	0,50	0,08	0,19
Q | p8 | 1,2	0,34	0,10	0,24
Q | p8 | 2,3	0,48	0,09	0,20
E | p0 | 1	0,45	0,10	0,23
E | p0 | 2	0,63	0,07	0,16
E | p4 | 1,2	0,47	0,08	0,18
E | p4 | 2,3	0,68	0,08	0,18
E | p8 | 1,2	0,45	0,08	0,19
E | p8 | 2,3	0,67	0,05	0,13
T | p0 | 1	0,24	0,10	0,23
T | p0 | 2	0,43	0,09	0,22
T | p4 | 1,2	0,34	0,09	0,22
T | p4 | 2,3	0,49	0,08	0,18
T | p8 | 1,2	0,32	0,09	0,20
T | p8 | 2,3	0,49	0,09	0,21

Figura 4.2: Votos, média, intervalo de confiança e desvio padrão dos SRCs.



HRC	Média	Int.Conf.	Desv.P.
Q | p0 | 1	0,77	0,07	0,15
Q | p0 | 2	0,89	0,08	0,19
Q | p4 | 1,2	0,86	0,07	0,16
Q | p4 | 2,3	0,91	0,09	0,21
Q | p8 | 1,2	0,84	0,05	0,12
Q | p8 | 2,3	0,94	0,09	0,20
E | p0 | 1	0,19	0,08	0,18
E | p0 | 2	0,70	0,09	0,21
E | p4 | 1,2	0,22	0,07	0,15
E | p4 | 2,3	0,60	0,07	0,16
E | p8 | 1,2	0,25	0,07	0,16
E | p8 | 2,3	0,66	0,07	0,17
T | p0 | 1	0,40	0,07	0,17
T | p0 | 2	0,52	0,09	0,20
T | p4 | 1,2	0,44	0,06	0,15
T | p4 | 2,3	0,51	0,07	0,16
T | p8 | 1,2	0,36	0,08	0,19
T | p8 | 2,3	0,54	0,07	0,16

HRC	Média	Int.Conf.	Desv.P.
Q | p0 | 1	0,64	0,10	0,23
Q | p0 | 2	0,79	0,07	0,16
Q | p4 | 1,2	0,66	0,08	0,19
Q | p4 | 2,3	0,81	0,06	0,13
Q | p8 | 1,2	0,69	0,06	0,15
Q | p8 | 2,3	0,79	0,05	0,11
E | p0 | 1	0,35	0,08	0,18
E | p0 | 2	0,71	0,07	0,16
E | p4 | 1,2	0,34	0,08	0,19
E | p4 | 2,3	0,72	0,08	0,19
E | p8 | 1,2	0,36	0,08	0,20
E | p8 | 2,3	0,71	0,06	0,13
T | p0 | 1	0,44	0,08	0,19
T | p0 | 2	0,64	0,08	0,18
T | p4 | 1,2	0,52	0,11	0,27
T | p4 | 2,3	0,63	0,08	0,20
T | p8 | 1,2	0,56	0,09	0,22
T | p8 | 2,3	0,63	0,10	0,23

HRC	Média	Int.Conf.	Desv.P.
Q | p0 | 1	0,78	0,08	0,19
Q | p0 | 2	0,85	0,05	0,12
Q | p4 | 1,2	0,75	0,06	0,13
Q | p4 | 2,3	0,85	0,07	0,16
Q | p8 | 1,2	0,77	0,07	0,16
Q | p8 | 2,3	0,88	0,06	0,13
E | p0 | 1	0,28	0,08	0,18
E | p0 | 2	0,75	0,06	0,14
E | p4 | 1,2	0,39	0,08	0,19
E | p4 | 2,3	0,81	0,06	0,14
E | p8 | 1,2	0,44	0,07	0,17
E | p8 | 2,3	0,76	0,05	0,13
T | p0 | 1	0,47	0,08	0,19
T | p0 | 2	0,61	0,07	0,17
T | p4 | 1,2	0,49	0,08	0,20
T | p4 | 2,3	0,59	0,09	0,21
T | p8 | 1,2	0,47	0,09	0,22
T | p8 | 2,3	0,55	0,10	0,23

HRC	Média	Int.Conf.	Desv.P.
Q | p0 | 1	0,58	0,08	0,19
Q | p0 | 2	0,74	0,09	0,21
Q | p4 | 1,2	0,66	0,09	0,20
Q | p4 | 2,3	0,74	0,07	0,17
Q | p8 | 1,2	0,67	0,09	0,21
Q | p8 | 2,3	0,77	0,08	0,20
E | p0 | 1	0,26	0,11	0,25
E | p0 | 2	0,61	0,10	0,24
E | p4 | 1,2	0,26	0,10	0,22
E | p4 | 2,3	0,64	0,08	0,20
E | p8 | 1,2	0,30	0,08	0,19
E | p8 | 2,3	0,67	0,07	0,16
T | p0 | 1	0,36	0,10	0,24
T | p0 | 2	0,46	0,10	0,23
T | p4 | 1,2	0,40	0,11	0,27
T | p4 | 2,3	0,45	0,09	0,21
T | p8 | 1,2	0,42	0,11	0,27
T | p8 | 2,3	0,45	0,08	0,19

Figura 4.3: Votos, média, intervalo de confiança e desvio padrão dos SRCs (continuação).

ções do que em relação às outras variações. Ou seja, os dados mostram que os avaliadores têm maior concordância na avaliação de qualidade onde a variação está no eixo espacial (o que acontece nas configurações Qualidade e também na Espacial, pois os vídeos foram todos exibidos com a mesma resolução espacial) do que vídeos onde as variações ocorrem no eixo temporal.

Para verificar a relação entre os votos atribuídos aos diferentes SRCs, foi utilizada a correlação de Pearson, assim como adotada nos testes do VQEG para comparação entre resultados objetivos e subjetivos (VQEG, 2008a). A equação 4.8 mostra como é encontrada a correlação de Pearson entre dois conjuntos de dados X e Y, onde N é o número total de elementos e $\overline{X}$ e $\overline{Y}$ são as médias para os conjuntos X e Y, respectivamente. O resultado será um número entre -1 e 1, onde 1 indica conjuntos iguais (correlação máxima), -1 indica conjuntos inversos e 0 indica conjuntos sem nenhuma correlação.

$$R = \frac{\sum_{i=1}^{N}(X_i - \overline{X}) * (Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^{N}(X_i - \overline{X})^2} * \sqrt{\sum_{i=1}^{N}(Y_i - \overline{Y})^2}} \tag{4.8}$$

Para cálculo da correlação, foi criado um vetor para cada SRC com a média dos votos dos 18 HRCs (na mesma ordem em que são apresentados nas figuras 4.2 e 4.3) e esses vetores foram comparados em pares. A maioria das comparações resultaram em valores acima de 0,85, mostrando que a resposta dos avaliadores para os HRCs tende a ser similar para todos os SRCs. Entretanto, dois SRCs apresentaram baixa correlação quando comparados aos outros SRCs: "ed1" e, especialmente, "speedbag". A tabela 4.1 mostra os valores encontrados em todas as comparações, destacando aqueles com baixa correlação. Os SRCs são identificados nas linhas e colunas através dos números de 1 à 8, conforme os mesmos identificadores já apresentados anteriormente.

Tabela 4.1: Tabela de correlação entre os SRCs.

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1. ed1	-	*0,58*	*0,61*	*0,58*	*0,45*	*0,74*	*0,58*	*0,57*
2. touchdownpass		-	0,97	*0,50*	0,85	0,86	0,92	0,92
3. redkayak			-	*0,44*	0,84	0,87	0,92	0,92
4. speedbag				-	*0,03*	*0,27*	*0,23*	*0,22*
5. rushfieldcuts					-	0,91	0,95	0,96
6. controlledburn						-	0,94	0,95
7. aspen							-	0,97
8. westwindeasy								-

A baixa correlação apenas para os vídeos "ed1" e "speedbag" ocorreu pois os HRCs que usam a configuração Qualidade (chamados Q-HRCs) apresentaram votos bastante abaixo da média quando aplicados a esses dois vídeos em comparação aos votos obtidos pelos mesmos HRCs quando aplicados aos demais vídeos, como pode ser visto nos gráficos dos respectivos vídeos na figura 4.2. Devido a isso, esses dois vídeos foram os que apresentaram menor MOS_s, como será visto na seção 4.3. O motivo para os Q-HRCs aplicados a esses vídeos apresentarem votos abaixo da média está relacionado à taxa de codificação apresentada por eles, que foram os dois vídeos com menores taxas entre os 11

codificados, como será comentado na seção 4.5. Apesar dos valores baixos para as taxas de codificação, é válido reafirmar que ambos foram codificados de acordo com as especificações da seção 3.2, assim como os demais vídeos. A figura 4.4 mostra graficamente a relação entre os dois vídeos que resultaram na maior correlação (a) e entre os dois vídeos que tiveram menor correlação (b).

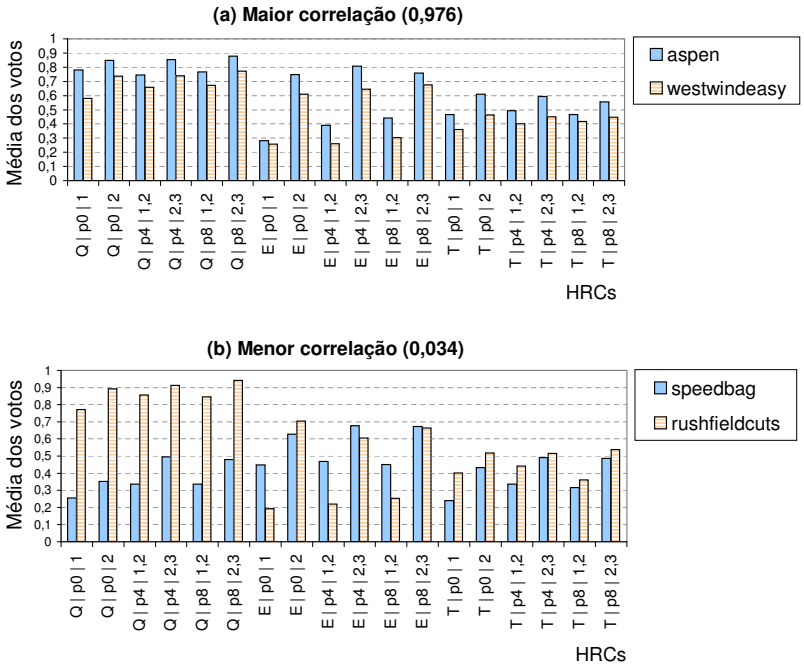


Figura 4.4: Maior e menor correlação entre os SRCs.

Essas exceções na verificação da correlação também mostram a importância do conteúdo dos vídeos e a influência que ele tem no processo de codificação. A remoção desses 2 vídeos considerados como exceções teve pouco impacto nos resultados se comparados aos resultados utilizando os 8 SRCs. A diferença principal foi que o MOS_h dos Q-HRCs passou a ser 0,087 mais alto, em média. Apesar desta diferença, todos os resultados foram calculados com os 8 SRCs avaliados.

4.3 Média dos votos para SRCs e HRCs

Os valores MOS são apresentados para cada SRC e cada HRC na figura 4.5. No gráfico (a), os valores estão sendo apresentados para cada SRC, ou seja, cada barra representa um MOS_s, a média dos votos normalizados de todos avaliadores para todos HRCs que processaram o SRC alvo. Da mesma forma é calculado o gráfico (b), mas agora cada barra representa um MOS_h, ou seja, a média dos votos normalizados de todos os avaliadores para todos SRCs que foram processados com o HRC alvo. Os valores reais do MOS são apresentados numericamente dentro de cada barra.

Os gráficos da figura 4.5 também mostram os intervalos de confiança de 95% como uma linha preta no limite de cada barra, que representam os possíveis valores "reais" com 95% de precisão. Ao lado direito de cada gráfico é exibido o valor de cada um destes intervalos de confiança. Para o gráfico dos SRCs (a), os intervalos de confiança ficaram entre 0,023 e 0,030, com média igual a 0,026, e, para o gráfico dos HRCs (b), ficaram entre 0,026 e 0,039, com média 0,033. Esses valores são pequenos, indicando, portanto,

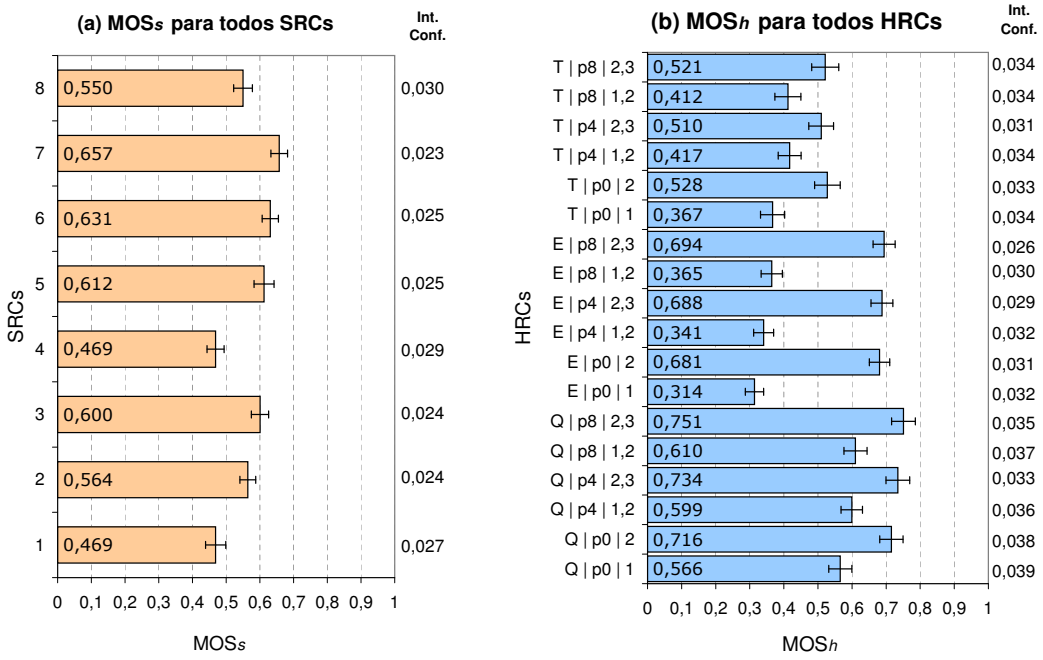


Figura 4.5: MOS para todos SRCs e HRCs.

uma boa precisão dos resultados.

O gráfico (a) da figura 4.5 mostra que o MOS dos SRCs está perto do centro da escala para todos os SRCs, pois a maioria dos valores está entre 0,5 e 0,6. Esta é uma evidência de que os HRCs usados nas avaliações geraram PVSs com grande variação de qualidade (ou seja, qualidade de excelente à péssima) e que os avaliadores conseguiram dispor os seus votos ao longo de toda a escala, sem concentrá-los nos limites inferior (muito pessimistas) ou superior (muito otimistas). Esta afirmação pode ser reforçada utilizando o MOS mínimo e máximo dos HRCs de alguns vídeos (podem ser vistos nas figuras 4.2 e 4.3). Por exemplo, o vídeo "rushfieldcuts" (SRC 5) apresenta seu menor MOS_h com valor igual a 0,192 e seu maior MOS_h com valor 0,941, que são valores próximos aos limites da escala de votação (0 e 10). Calculando a média do MOS de todos HRCs deste vídeo, o resultado é 0,59, que está próximo ao valor central da escala. Já o vídeo "speedbag" que possui valores entre 0,24 e 0,676, por exemplo, é o que apresenta a menor variação.

4.4 Resultados da instabilidade

A figura 4.6 apresenta o mesmo gráfico da figura 4.5 (b), mas ordenado para facilitar a análise da relação entre qualidade e instabilidade. O gráfico (a) mostra os resultados somente para a configuração de codificação Temporal, o (b) para a Espacial e o (c) para a Qualidade. As primeiras 3 barras dos gráficos representam, respectivamente, os padrões de variação de camadas estável ($p0$), com pouca variação ($p4$) e com muita variação ($p8$), aplicados nas camadas 1 e 2 dos vídeos (no padrão estável, somente na camada 1). As 3 últimas barras representam os mesmos padrões aplicados nas camadas 2 e 3 (e o padrão estável somente na camada 2). Abaixo das barras é exibido o nome do HRC e cada barra também apresenta seu intervalo de confiança de 95%, assim como na figura 4.5 (b).

Acima de cada dupla de barras é exibido um arco com a diferença entre os valores (valor da barra da direita menos o valor da barra da esquerda), para auxiliar a verificação da diferença de qualidade entre os padrões de instabilidade.

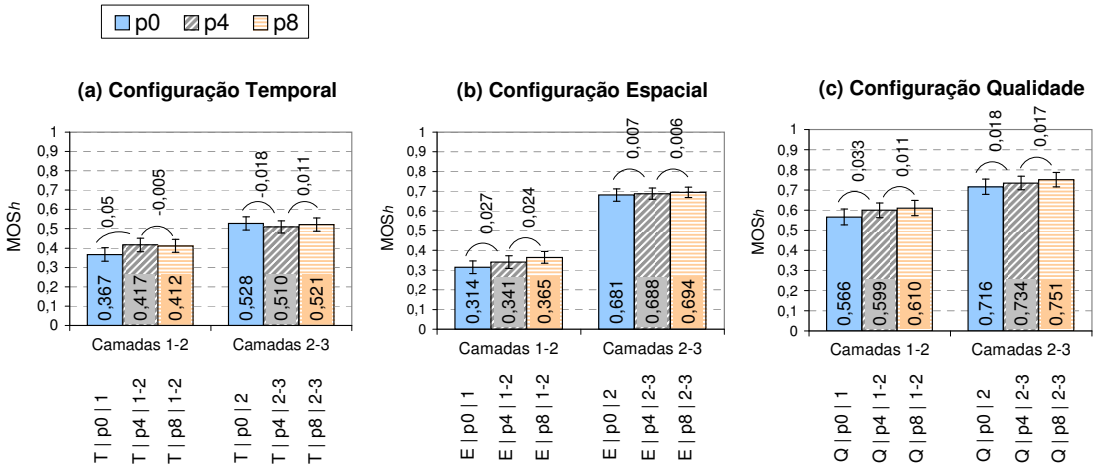


Figura 4.6: Análise do MOS_h em relação à instabilidade.

A tabela 4.2 mostra as diferenças de qualidade entre os padrões de instabilidade para cada um dos SRCs utilizados, calculados a partir da média dos votos de cada HRC para os SRCs individualmente, como já apresentado nas figuras 4.2 e 4.3. Cada linha da tabela está associada a um HRC, indicado na coluna "Configuração". A terceira coluna indica entre quais padrões de instabilidade está sendo feita a comparação: diferença entre o padrão $p4$ e o padrão $p0$ ($p4$-$p0$) ou diferença entre o padrão $p8$ e o padrão $p4$ ($p8$-$p4$). Os SRCs estão numerados conforme já apresentado anteriormente (ver tabela 3.8) e a última coluna apresenta a média das diferenças entre todos SRCs. É importante observar que esta coluna das médias não apresenta valores exatamente iguais às diferenças apresentadas na figura 4.6. Isso ocorre pois, na tabela, inicialmente foram calculadas as diferenças para cada SRC individualmente e depois foram calculadas as médias, enquanto na figura, inicialmente foram calculados os valores do MOS_h (ou seja, as médias já incluindo todos SRCs) e só depois foram verificadas as diferenças.

Tabela 4.2: Diferenças entre os padrões de instabilidade para cada SRC.

Configuração			**SRCs**								
			1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**Média**
T	1-2	$p4$-$p0$	0,021	0,017	0,001	0,080	0,085	0,023	-0,033	0,077	0,034
		$p8$-$p4$	0,011	0,051	-0,031	0,001	-0,012	0,031	0,022	0,014	0,011
	2-3	$p4$-$p0$	0,011	0,008	-0,061	0,145	0,020	0,022	0,005	0,000	0,019
		$p8$-$p4$	0,029	0,020	0,029	-0,016	0,028	-0,019	0,025	0,035	0,016
E	1-2	$p4$-$p0$	0,039	0,025	-0,001	0,021	0,028	-0,005	0,107	0,001	0,027
		$p8$-$p4$	0,006	-0,027	0,084	-0,018	0,033	0,014	0,054	0,045	0,024
	2-3	$p4$-$p0$	0,050	-0,016	-0,028	0,048	-0,098	0,008	0,060	0,034	0,007
		$p8$-$p4$	-0,042	-0,014	0,085	-0,004	0,059	-0,013	-0,050	0,030	0,007
Q	1-2	$p4$-$p0$	0,060	0,062	-0,005	0,096	0,041	0,079	0,026	0,039	0,050
		$p8$-$p4$	-0,018	-0,013	0,068	-0,022	-0,081	0,037	-0,027	0,017	-0,005
	2-3	$p4$-$p0$	-0,036	-0,021	-0,096	0,059	-0,004	-0,014	-0,018	-0,015	-0,018
		$p8$-$p4$	0,041	0,011	0,066	-0,005	0,023	0,000	-0,039	-0,002	0,012

Como se pode perceber nos três gráficos, a diferença do MOS entre os três padrões de variação é muito pequena, onde, na maioria dos casos, os padrões instáveis ($p4$ e $p8$) apresentam valores um pouco superiores ao estável ($p0$). Esta superioridade, porém, é muito pequena (em média 0,02). Isso mostra que a instabilidade, da forma modelada neste trabalho, não degradou a qualidade do vídeo, porém também não acrescentou nenhuma qualidade. Comparando os dois padrões instáveis também não existe grande diferença, sendo que o padrão $p8$ normalmente tem MOS levemente maior que o padrão $p4$ (em média, é 0,01 maior). Estas afirmações são válidas tanto para as camadas 1-2 quanto para as camadas 2-3.

Apesar das diferenças serem pequenas, os padrões instáveis ($p4$ e $p8$) em geral apresentaram melhor qualidade do que o padrão estável ($p0$), como já foi comentado. Este resultado não era esperado, pois a instabilidade é vista como um problema para a transmissão e variações no vídeo ao longo de sua exibição são normalmente vistas como prejudiciais. Um dos motivos que podem explicar esses resultados é que a troca de camadas na simulação de instabilidade não implica em nenhuma perda de qualidade adicional além daquela existente entre uma camada e outra (devido à perda de pacotes, por exemplo), como já comentado na seção 3.2.2. Outra interpretação possível é que, devido ao curto tempo de reprodução dos vídeos (14 segundos), as variações de camadas podem ter sido entendidas pelos avaliadores como um vídeo de boa qualidade que apresentou momentos com qualidade reduzida, e não como um vídeo de qualidade reduzida que apresentou momentos de melhor qualidade, como é o cenário proposto. Uma possibilidade seria expandir o tempo de exibição dos vídeos, como será comentado na seção 5.1 como um dos trabalhos futuros.

Novamente, apesar de as diferenças entre os padrões de instabilidade terem sido pequenas, os maiores valores se encontram na comparação dos HRCs aplicados nas camadas inferiores (1-2). Como exemplo, podem ser vistas na tabela 4.2 as linhas: "T | 1-2 | p4-p0" (1ª linha), "E | 1-2 | p4-p0" (5ª), "E | 1-2 | p8-p4" (6ª) e "Q | 1-2 | p4-p0" (9ª), que apresentam médias 0,034, 0,027, 0,024 e 0,050, respectivamente. Este é um indício de que variações em camadas inferiores (resolução espacial, temporal e/ou PSNR inferiores) podem ser mais facilmente percebidas do que variações em camadas superiores, e, em geral, indicam que essas variações representam um pequeno ganho de qualidade.

4.5 Análise dos resultados em relação às taxas de codificação

Embora não seja o propósito principal deste trabalho, a análise da qualidade em relação à taxa de codificação de cada vídeo também é muito importante. A análise das taxas utiliza somente os HRCs que utilizam o padrão estável de variação das camadas ($p0$), pois os outros padrões, devido à instabilidade, geram vídeos com taxas variáveis ao longo do tempo. É possível calcular uma taxa média para os casos instáveis, mas os votos atribuídos a essas PVSs estão relacionados às variações de camadas que elas apresentam, e não à simples codificação de um vídeo à uma taxa fixa, que é o que está sendo avaliado nesta seção.

A figura 4.7 mostra a relação entre o MOS obtido e a taxa de codificação dos vídeos. O primeiro gráfico (a) mostra a média dos resultados para todos SRCs, ou seja, tanto os valores MOS quanto as taxas exibidas no gráfico foram calculadas a partir da média dos valores de todos SRCs. Estes valores são apenas para o padrão estável de variação de camadas ($p0$), e cada linha do gráfico representa uma das configurações de codificação (Temporal, Espacial e Qualidade). Cada linha apresenta 2 pontos, sendo que o primeiro

ponto representa os resultados para a primeira camada (HRC “p0 | 1-2”, conforme legendas anteriores) e o segundo ponto representa os resultados para a segunda camada (HRC “p0 | 2-3”).

O segundo gráfico (b) da figura 4.7 mostra os mesmos dados do gráfico (a), mas agora juntamente com o MOS e a taxa para os padrões instáveis ($p4$ e $p8$). Como indicado na linha que representa a configuração Temporal, os 6 pontos de cada linha são, respectivamente: “p0 | 1-2”, “p4 | 1-2”, “p8 | 1-2”, “p0 | 2-3”, “p4 | 2-3” e “p8 | 2-3”. As taxas dos padrões instáveis foram calculadas a partir da taxa das camadas utilizadas e do tempo em que cada uma delas foi utilizada na simulação de instabilidade. Por exemplo, para calcular a taxa do segundo ponto das linhas, foram utilizadas as taxas das camadas 1 e 2 e o tempo total utilizado para cada camada foi 10 segundos para a camada 1 e 4 segundos para a camada 2, como é definido pelo padrão de instabilidade $p4$ (ver seção 3.3.5). Nesse gráfico pode ser visto que a instabilidade normalmente aumenta levemente a qualidade subjetiva (como já visto anteriormente), mas aumenta significantemente a taxa de codificação (mais precisamente a banda de rede utilizada, já que a instabilidade ocorre na transmissão e não a codificação), especialmente quando a instabilidade é entre as camadas 2 e 3. A tabela 4.3 mostra os dados para esses dois gráficos, onde as linhas da tabela representam os valores do MOS ou das taxas na mesma ordem em que são apresentados os pontos do gráfico (b) da figura 4.7.

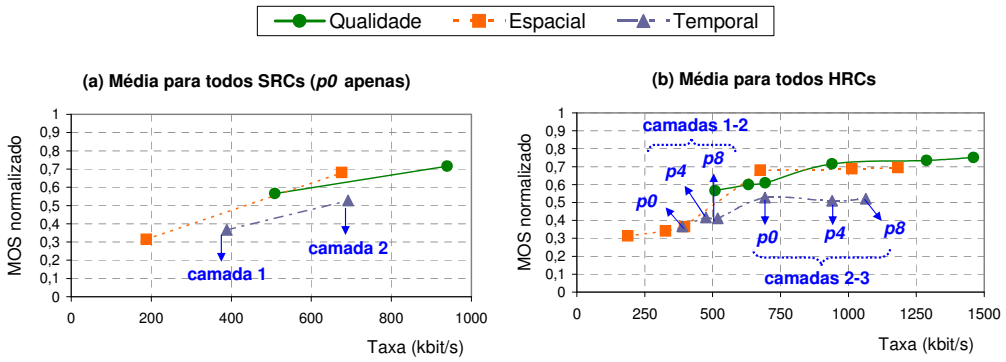


Figura 4.7: Análise da qualidade em relação à taxa de codificação dos vídeos.

Tabela 4.3: Comparação entre a variação do MOS e da taxa de codificação dos vídeos.

	Camadas:	**1-2**			**2-3**		
	Padrão:	$p0$	$p4$	$p8$	$p0$	$p4$	$p8$
T	**MOS:**	0,367	0,417	0,412	0,528	0,510	0,521
	Taxa (kbit/s):	388,65	475,37	518,74	692,20	940,12	1064,08
E	**MOS:**	0,314	0,341	0,364	0,681	0,688	0,694
	Taxa (kbit/s):	187,62	327,06	396,79	675,69	1013,61	1182,57
Q	**MOS:**	0,565	0,599	0,610	0,716	0,734	0,751
	Taxa (kbit/s):	508,96	632,08	693,64	939,88	1287,41	1461,18

A figura 4.8 apresenta gráficos criados da mesma forma que os gráficos da figura 4.7 (a), ou seja, utilizando apenas o padrão estável ($p0$). Porém, eles foram criados utilizando

apenas um SRC, e não a média dos votos de todos SRCs. Os valores para as taxas de codificação de cada SRC já foram apresentados na tabela 3.10. O SRC "rushfieldcuts" (5) foi o que apresentou maior taxa entre todos, enquanto o SRC "ed1" (1) foi o que apresentou menor taxa. Estes gráficos mostram que a relação entre MOS e taxa de codificação pode ser bastante variada dependendo do conteúdo dos vídeos, porém, na maioria dos casos a relação é semelhante. Os SRCs que apresentaram comportamento mais diferenciado dos outros foram os SRCs "ed1" e "speedbag", que também foram os SRCs que apresentaram menores taxas e que apresentaram menor correlação na comparação dos valores MOS_h entre os SRCs (ver tabela 4.1).

Os dados apresentados em relação à taxa dos vídeos serão utilizados na próxima seção (4.6) para comparação entre os métodos de escalabilidade.

4.6 Comparação entre os métodos de escalabilidade

A figura 4.9 mostra a relação entre as três configurações de codificação utilizadas, ou seja, a relação entre as escalabilidades temporal, espacial e de qualidade. Os valores são exibidos para cada um dos padrões de instabilidade isoladamente nos gráficos (a), (b) e (c), e uma média para os três padrões juntos em (d). Estes valores são os mesmos vistos na figura 4.6, mas organizados para isolar os padrões de instabilidade. Cada linha dos gráficos representa uma configuração de codificação, e cada um dos dois pontos das linhas representa a aplicação dos HRCs nas camadas 1-2 ou nas camadas 2-3. O primeiro ponto da linha sólida (em verde) no gráfico (a), por exemplo, corresponde ao HRC "Q | p0 | 1-2", e o segundo ponto corresponde ao HRC "Q | p0 | 2-3".

Como mencionado anteriormente, a diferença do MOS entre os padrões de instabilidade é bastante pequena, e isso pode ser visto comparando os 3 primeiros gráficos da figura 4.9 e notando que a média (d) também é muito similar a eles. Analisando o MOS médio através do gráfico (d), pode ser visto que a configuração Temporal tem MOS menores do que as outras, com 0,398 para as camadas 1-2 e 0,519 para as camadas 2-3. Para a configuração Espacial, as camadas 1-2 apresentaram MOS 0,339, valor um pouco inferior do MOS das camadas 1-2 para a configuração Temporal. Entretanto, para as camadas 2-3, a configuração Espacial teve um grande crescimento de qualidade, chegando ao valor de MOS 0,687. Já a configuração Qualidade apresentou valores MOS maiores do que as outras, com 0,591 nas camadas 1-2 e 0,733 nas camadas 2-3. Estes resultados mostram que, com ou sem instabilidade, a escalabilidade de qualidade obteve resultados melhores, enquanto a escalabilidade temporal foi a que apresentou pior desempenho. Para a escalabilidade espacial, as camadas superiores (CIF e 4CIF) tiveram resultado semelhante à escalabilidade de qualidade, mas, nas camadas inferiores, o uso de uma resolução muito reduzida em relação à resolução usada para exibição dos vídeos (4CIF) provocou redução considerável na qualidade.

A relação entre MOS e taxa de codificação para as configurações de codificação pode ser analisada utilizando as figuras 4.7 e 4.9 e os dados da tabela 4.3. Comparando as médias das configurações Espacial e Qualidade, pode ser visto que em ambos os pontos do gráfico (a) da figura 4.7 a configuração Qualidade possui taxa maior do que a configuração Espacial. Na primeira camada (primeiro ponto das linhas), a diferença das taxas é de 321,34 kbit/s e a diferença de MOS é 0,224. Já para a segunda camada, um aumento 264,19 kbit/s é visto na taxa, mas a diferença de MOS é bastante pequena, com valor 0,043. Esses valores certamente são diferentes se cada SRC for analisada individualmente, mas a maioria dos vídeos apresentou comportamento semelhante à este, como

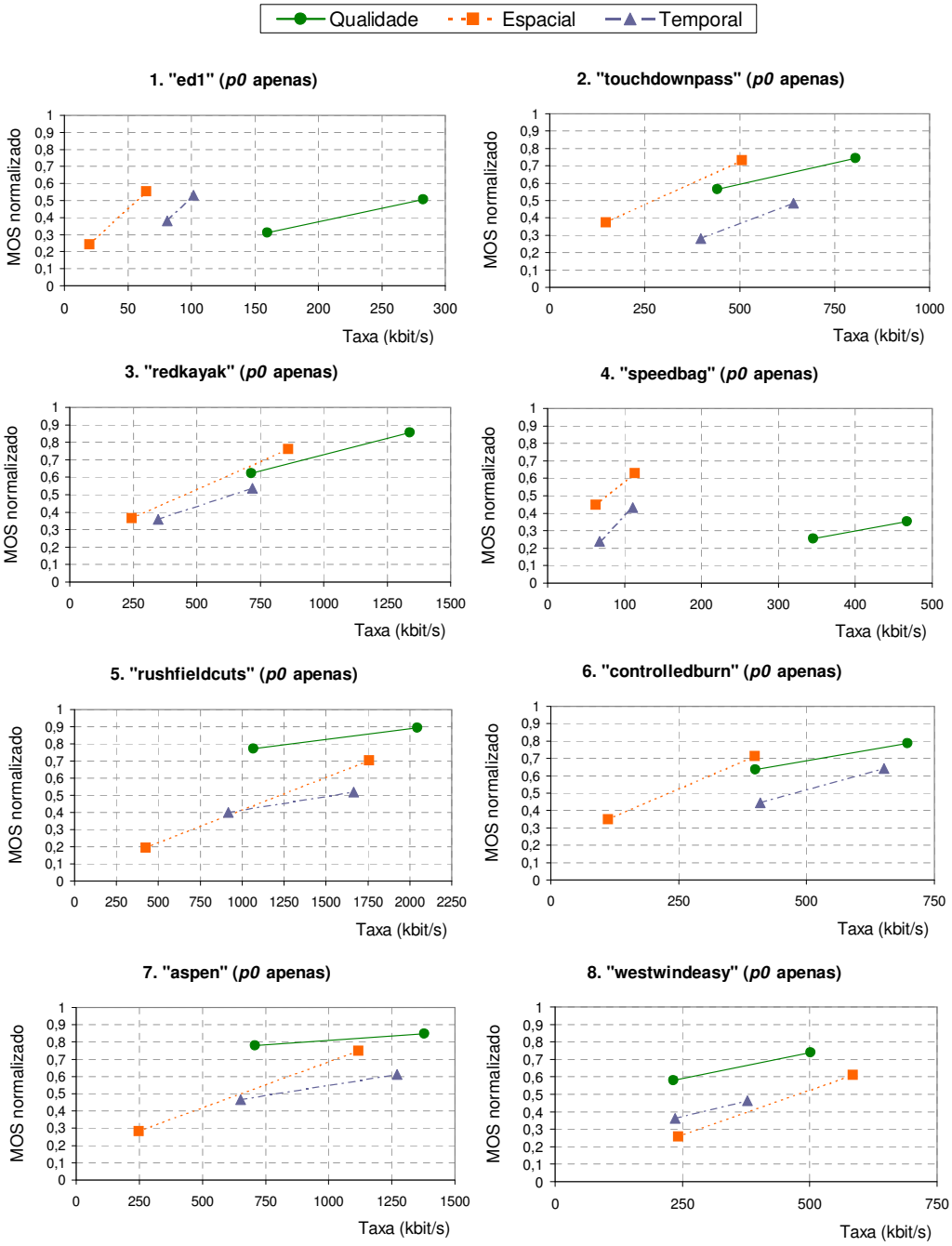


Figura 4.8: Análise da qualidade em relação à taxa de codificação para cada SRC individualmente.

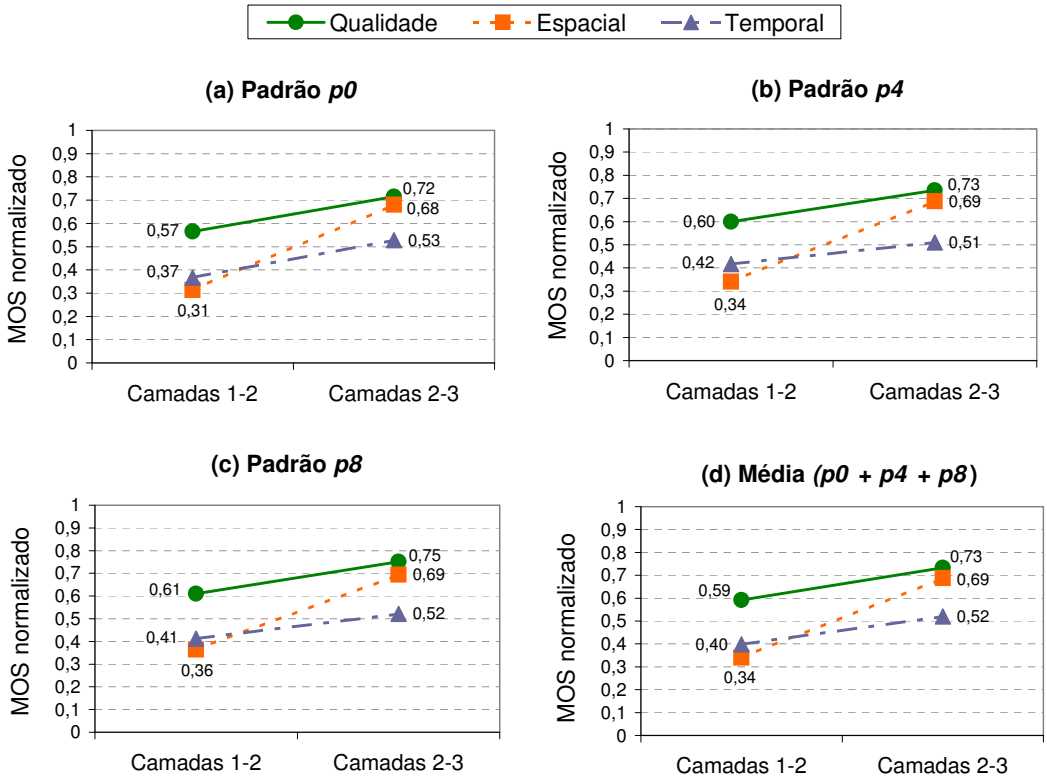


Figura 4.9: Relação da qualidade entre os métodos de escalabilidade.

pode ser visto na figura 4.8.

Comparada à configuração Temporal, a configuração Qualidade também apresentou MOS e taxas superiores. Na primeira camada, a configuração Qualidade apresenta 120,31 kbit/s a mais na taxa e apresenta MOS em média 0,244 superior à configuração Temporal. Na segunda camada as diferenças são parecidas, sendo 247,67 kbit/s na taxa e 0,22 pontos no MOS.

Por fim, comparando as configurações Temporal e Espacial, pode ser visto que a Espacial apresenta menores taxas na maioria dos casos e também possui MOS superior na maioria deles. Para a primeira camada, a configuração Espacial apresenta a taxa média 201,03 kbit/s inferior e tem valor de MOS apenas 0,053 pontos inferior, enquanto para a segunda camada, a taxa é apenas 16,51 kbit/s inferior mas o MOS passa a ser 0,176 pontos superior.

Resumindo a comparação entre todas as configurações e analisando os gráficos individuais de cada SRC (figura 4.8), temos:

Espacial e Qualidade: Primeira camada da configuração Qualidade normalmente tem qualidade bastante superior, mas acaba apresentando maiores taxas de codificação para isso (apesar de a taxa não aumentar em todos os casos). Segunda camada da configuração Qualidade apresenta taxa sempre superior, como na maioria dos casos da primeira camada. Mas, para a segunda camada, a configuração Qualidade apresenta superioridade de MOS pequena, menor do que na primeira camada.

Temporal e Qualidade: Na maioria dos casos, na primeira camada a configuração Qualidade apresenta taxas bastante semelhantes à configuração Temporal, e ainda assim apresenta MOS superior. Nos outros casos o MOS também é superior, mas a taxa

da configuração Qualidade é consideravelmente maior. Já para a segunda camada, a superioridade do MOS se mantém semelhante, mas a configuração Qualidade passa a apresentar sempre taxas maiores.

Temporal e Espacial: Em praticamente todos os casos, a configuração Espacial apresenta menores taxas para a primeira camada, mas a diferença de MOS varia: em alguns casos é maior para a Espacial, em alguns casos é bastante parecida e em alguns casos é maior para a Temporal. Já na segunda camada, a configuração Espacial apresenta maior MOS na grande maioria dos casos, mas a diferença de taxas que varia neste caso: algumas vezes é maior para a configuração Temporal ou Espacial e outras vezes é bastante semelhante.

Em relação à comparação dos métodos de escalabilidade, algumas observações devem ser feitas. A configuração Qualidade foi a que apresentou melhor qualidade subjetiva, mas também apresentou maiores taxas de codificação na maioria dos casos. Já a configuração Espacial obteve pior qualidade subjetiva, mas muitas vezes utilizando menores taxas. Esta análise das taxas é importante, mas é apenas complementar neste trabalho, pois as HRCs não foram criadas para este objetivo. Com isso, as verificações das taxas acabam sendo feitas em apenas 2 pontos para cada configuração (apenas HRCs que utilizam o padrão $p0$ nas camadas 1 ou 2, como comentado no início da seção 4.5), não possibilitando uma verificação mais detalhada da influência destas taxas de codificação.

Outra observação é em relação ao uso de camadas com 3,75 fps, 7,5 fps e 30 fps para a configuração Temporal, eliminando a camada intermediária com 15 fps (ver seção 3.2.1). Com uma redução de 30 fps para 15 fps e não de 30 fps para 7,5 fps, a qualidade da configuração Temporal teria tendência a ser maior do que a obtida, mas provavelmente também apresentaria maiores taxas. Já a configuração Espacial poderia obter melhores resultados caso fosse utilizada alguma técnica de interpolação para ampliação das resoluções QCIF e CIF para 4CIF.

5 CONCLUSÕES

O trabalho descrito nesta dissertação está incluso na área de avaliação de qualidade de vídeo, também utilizando outras duas áreas de pesquisa da computação que são a codificação de vídeo e a transmissão de dados. Apesar de representarem diferentes áreas do conhecimento, estas áreas estão interligadas, e o estudo de uma geralmente envolve conhecimentos sobre as outras. Esta relação é especialmente vista neste trabalho, onde o objetivo principal é a avaliação de qualidade de vídeo, que envolve tanto a codificação quanto a transmissão desses dados.

O processo de avaliação de qualidade foi feito de forma subjetiva, ou seja, os resultados das avaliações foram obtidos através da opinião de pessoas sobre a qualidade dos vídeos visualizados. A avaliação subjetiva é geralmente organizada em diversas etapas e requer um detalhamento especial durante a especificação inicial dos objetivos e na criação do plano de avaliação. Todas as etapas realizadas foram detalhadas no capítulo 3 desta dissertação, e os resultados obtidos após as avaliações foram apresentados no capítulo 4.

O objetivo principal das avaliações subjetivas realizadas foi verificar os efeitos que a instabilidade na transmissão tem sobre a qualidade dos vídeos, principalmente em sistemas que utilizam transmissão em camadas. Foi utilizado o padrão H.264 SVC para codificação escalável dos vídeos e a metodologia de avaliação subjetiva ACR-HRR para aplicação das avaliações. Foram aplicadas 18 alterações (HRCs) sobre os 8 vídeos originais (SRCs), gerando assim 152 PVSs que foram avaliadas por um grupo de 22 avaliadores. O conjunto de alterações aplicadas sobre os vídeos permitiu analisar os efeitos da instabilidade, conforme o objetivo principal do trabalho, e também alguns objetivos secundários, como a comparação entre os três métodos de escalabilidade utilizados.

As principais contribuições deste trabalho estão nos resultados apresentados no capítulo 4, além da metodologia utilizada durante o desenvolvimento do trabalho (definição do plano de avaliação, uso das ferramentas como o JSVM, seleção do material de teste, execução das avaliações, entre outros), das aplicações desenvolvidas (ver apêndice B), da definição de alguns trabalhos futuros, que serão comentados ao longo desta seção, e possíveis objetivos que foram definidos para avaliações de qualidade (ver apêndice A).

A análise dos resultados mostrou que os vídeos instáveis têm qualidade subjetiva muito semelhante à dos vídeos estáveis, tanto para o padrão com pouca variação de camadas quanto para o padrão com bastante variação. Esse resultado foi muito semelhante para os três métodos de escalabilidade utilizados, mostrando que a instabilidade é percebida de maneira semelhante em todos os métodos, com exceção de que, na escalabilidade temporal, a percepção da qualidade tende a ser mais divergente entre os avaliadores. Como os padrões instáveis são semelhantes ao padrão estável, eles também são semelhantes entre si, o que indica que o aumento no número de variações nas camadas (comparando os padrões de instabilidade $p4$ com $p8$) não resultou em uma redução ou aumento significante

na qualidade dos vídeos. Além disso, estes resultados indicam que o nível de instabilidade pode não ser tão importante quanto a presença da instabilidade, ou seja, a diferença entre os padrões instáveis (diferença com valor n) é menor do que a diferença entre qualquer padrão instável e o padrão estável (diferença com valor $n' > n$, apesar de a diferença entre n e n' ser pequena). Além desta análise de instabilidade, seria interessante investigar os efeitos que as perdas ocasionadas por esta instabilidade têm sobre a qualidade subjetiva dos vídeos.

Comparando a qualidade média obtida para cada método de escalabilidade, a escalabilidade temporal apresentou qualidade sempre inferior aos outros métodos, enquanto a escalabilidade de qualidade superou as outras na maioria dos casos. A escalabilidade espacial apresentou qualidade subjetiva baixa em camadas inferiores (QCIF, neste caso), mas em camadas superiores (CIF, 4CIF) se aproximou dos resultados apresentados pela escalabilidade de qualidade.

Sendo as degradações introduzidas pelas escalabilidades de qualidade e espacial semelhantes (blocagem ou borramento dos quadros), o pior desempenho da escalabilidade espacial mostra que esta introduz artifícios mais desagradáveis ao olho humano. Porém, o uso de alguma técnica de interpolação para ampliação das resoluções menores poderia melhorar os resultados obtidos. Já o pior desempenho da escalabilidade temporal mostra que alterações no fluxo do vídeo tendem a ser mais indesejáveis do que alterações na qualidade dos quadros.

Como objetivo secundário, a análise da qualidade em relação à taxa de codificação dos vídeos foi feita com base nos HRCs que utilizavam o padrão estável ($p0$), onde pôde ser visto que há um equilíbrio entre as escalabilidade espacial e de qualidade. A escalabilidade de qualidade apresentou qualidade subjetiva maior, mas às custas de uma taxa também um pouco maior. Já a escalabilidade temporal mostrou qualidade subjetiva pior do que outras e, na maioria dos casos, ainda apresentou maiores taxas.

De acordo com esses resultados, a escalabilidade temporal deveria ser utilizada somente quando realmente necessário (limitações de dispositivos, por exemplo). Caso contrário, é interessante utilizar escalabilidade espacial, mas limitando as camadas inferiores para uma resolução não muito baixa (não baixa como QCIF, por exemplo) e a escalabilidade de qualidade. Um trabalho futuro é a verificação de maiores detalhes desta relação entre os métodos de escalabilidade, assim como a utilização de mais de um método em conjunto, para complementar os dados aqui apresentados.

Em relação ao questionário entregue aos avaliadores ao final das avaliações, a grande maioria achou a duração dos vídeos (14 segundos) boa o suficiente para percepção dos detalhes, mas a duração geral da avaliação (1 hora mais intervalo) foi um pouco cansativa. Em relação à qualidade geral dos vídeos, a maioria dos avaliadores opinou que havia bastante variação na qualidade e que a dificuldade de se perceber as variações ao longo de um mesmo vídeo era média (41% acharam média e 41% acharam fácil). Os conteúdos dos vídeos não foram dados como inadequados por nenhum avaliador, onde a maioria achou-os bom ou razoável. As respostas de cada avaliador para todas as questões são apresentadas no apêndice D.4.

Com a finalização das avaliações, foram observadas algumas dificuldades apresentadas durante o processo, assim como algumas melhorias em determinadas etapas. Uma possível melhoria é em relação à duração geral das avaliações, que, como já comentado, foi entendida como muito longa pela maioria dos avaliadores. A realização de avaliações com tempo de duração menor (30 minutos, por exemplo) e a redução no número de variações (HRCs) e/ou vídeos (SRcs) pode resolver este problema, porém seria necessário

um número maior de avaliações (e, portanto, mais avaliadores) para verificar o mesmo conjunto de PVSs utilizadas neste trabalho.

Outra melhoria possível é em relação à duração das PVSs. Apesar da utilização de 14 segundos já ser mais longa que os 8-10 segundos normalmente utilizados, as variações de camadas que ocorrem neste período podem não ser interpretadas da maneira correta. Como já comentado na seção 4.4, as variações de camadas podem ter sido interpretadas pelos avaliadores como um vídeo de boa qualidade que apresentou momentos com qualidade reduzida, e não como um vídeo de qualidade reduzida que apresentou momentos de melhor qualidade (o que realmente aconteceria em uma transmissão de acordo com o modelo simulado). A melhoria em relação à esta questão seria utilizar vídeos com duração bastante maior (1 minuto, por exemplo), e com conteúdo atrativo para o avaliador, simulando um ambiente onde o avaliador esteja visualizando um vídeo de seu interesse e onde variações de qualidade ocorrem durante a visualização.

Em relação ao ambiente, dois fatores principais foram observados: a iluminação e o isolamento da sala. Em relação à iluminação, foi seguida a recomendação da norma BT.500, que recomenda um ambiente com iluminação menor ou igual a 20 lux (ver seção 3.4). Apesar de a sala utilizada ser reservada apenas para a avaliação, ela não era completamente isolada dos efeitos da luz do dia (ou da falta de luz à noite). Esta iluminação fraca neste ambiente dificultava a concentração após certo período de avaliação, como foi comentado por alguns avaliadores.

Outro aspecto importante está na quantidade e na escolha dos avaliadores. Apesar de os 22 avaliadores que participaram das avaliações estarem em faixas etárias variadas e nunca terem participado de avaliações de qualidade antes, grande parte deles trabalha na área da computação e, inclusive, com codificação de vídeo. Estes avaliadores têm mais facilidade de avaliar vídeos e, possivelmente, opiniões diferentes das que teriam um público geral. Porém, principalmente devido à questão das variações de qualidade ao longo de um vídeo, é interessante que os avaliadores já tenham certo conhecimento sobre o assunto para que possam perceber e avaliar melhor estas variações.

5.1 Trabalhos futuros

A realização de avaliações de qualidade seguindo o objetivo II ("Avaliação dos métodos de escalabilidade" — ver seção 3.1 e apêndice A) é a primeira possibilidade de trabalho futuro, que complementaria os resultados apresentados neste trabalho. Esta avaliação inclui a geração de um novo conjunto de vídeos com o objetivo de analisar os métodos de escalabilidade, dando atenção especial para a relação entre qualidade e taxa de codificação. Diversas etapas necessárias já foram realizadas, como a definição do plano de avaliação e geração do material de teste (PVSs). Já foi, inclusive, definido o uso da metodologia SAMVIQ (já descrita na seção 2.4.1.2) que, assim como a ACR, é suportada pelo aplicativo desenvolvido para execução das avaliações (a ferramenta *wxSVQ*). Os resultados de avaliações com estes objetivos seriam muito importantes para complementar os resultados deste trabalho.

Outra possibilidade é utilizar os resultados obtidos nestas avaliações para comparação com resultados de avaliações objetivas. Entre os métodos objetivos propostos para uso nestas comparações estão a ferramenta VQM do ITS (comentada na seção 2.4.2), que obteve ótimos resultados em avaliações do VQEG, e outras duas propostas que consideram a questão da escalabilidade dos vídeos, ou seja, consideram a resolução espacial e a resolução temporal dos vídeos além da qualidade dos quadros em si. Estas duas propos-

tas aparecem nos trabalhos de Monteiro e Nunes (MONTEIRO; NUNES, 2007) e Kim *et al.* (KIM et al., 2008). Apesar das extensas avaliações realizadas pelo VQEG envolvendo validação de métodos objetivos, elas não são voltadas para codificação escalável e também não envolvem vídeos onde há variações durante sua exibição, como acontece nos momentos de instabilidade simulados neste trabalho. A comparação de métodos objetivos com os resultados das avaliações aqui apresentadas pode ser importante para verificar se estes métodos podem ser utilizados para avaliação de vídeos escaláveis e com variação durante a exibição e, assim, passar a utilizar esse(s) método(s) para realizar avaliações semelhantes às aqui apresentadas e/ou expandir os resultados para mais variações (HRCs) e vídeos (SRCs).

Como apresentado no início deste capítulo, uma das melhorias possíveis para as avaliações seria expandir o tempo de duração dos vídeos para facilitar a percepção das variações e simular com mais eficácia um ambiente real. Também com estes objetivos, um importante trabalho futuro é utilizar um sistema real de transmissão em camadas para geração dos vídeos avaliados, ou seja, eliminar a etapa de simulação de instabilidade e utilizar um sistema real para obter os vídeos instáveis. Como foi comentado na seção 2.3, a maior dificuldade para isso é a falta de um sistema de transmissão em camadas real disponível para uso, porém, o ALMTF está sendo implementado e validado em ambientes reais e alguns resultados já foram obtidos (KROB et al., 2007).

Outras possibilidades de trabalhos futuros são a aplicação dos outros objetivos apresentados no apêndice A, seja de forma subjetiva ou de forma objetiva.

REFERÊNCIAS

AARON, A. et al. Wyner-Ziv coding of motion video. In: ASILOMAR CONFERENCE ON SIGNALS, SYSTEMS AND COMPUTERS, 36., 2002, Pacific Grove, California. **Conference Record...** Piscataway: IEEE, 2002. v.1, p.240–244. doi:10.1109/ACSSC.2002.1197184.

AL-SHAYKH, O. K. et al. Video compression using matching pursuits. **IEEE Transactions on Circuits and Systems for Video Technology**, [S.l.], v.9, n.1, p.123–143, Feb. 1999. doi:10.1109/76.744280.

ALLNATT, J. **Transmitted-Picture Assessment**. [S.l.]: John Wiley & Sons, 1983.

ARNOLD, J. F. et al. Efficient drift-free signal-to-noise ratio scalability. **IEEE Transactions on Circuits and Systems for Video Technology**, [S.l.], v.10, n.1, p.70–82, Feb. 2000. doi:10.1109/76.825862.

BARONCINI, V. New Tendencies in Subjective Video Quality Evaluation. **IEICE Transactions on Fundamentals of Electronics**, [S.l.], v.E89-A, n.11, p.2933–2937, Nov. 2006.

BARZILAY, M. A. J.; TAAL, J. R.; LAGENDIJK, R. L. Subjective Quality Analysis of Bit Rate Exchange Between Temporal and SNR Scalability in the MPEG4 SVC Extension. In: IEEE INTERNATIONAL CONFERENCE ON IMAGE PROCESSING, ICIP, 2007, San Antonio, Texas. **Proceedings...** Piscataway: IEEE, 2007. n.2, p.285–288.

BROTHERTON, M. D. et al. Subjective Multimedia Quality Assessment. **IEICE Transactions on Fundamentals of Electronics**, [S.l.], v.E89-A, n.11, p.2920–2932, November 2006. doi:10.1093/ietfec/e89-a.11.2920.

BRUN, P.; HAUSKE, G.; STOCKHAMMER, T. Subjective assessment of H.264-AVC video for low-bitrate multimedia messaging services. In: INTERNATIONAL CONFERENCE ON IMAGE PROCESSING, ICIP, 2004. **Anais...** [S.l.: s.n.], 2004. v.2, p.1145–1148. doi:10.1109/ICIP.2004.1419506.

BRUNO, G. **VEBIT**: um novo algoritmo para codificação de vídeo com escalabilidade. 2003. Dissertação (Mestrado em Ciência da Computação) — Instituto de Informática, Universidade Federal do Rio Grande do Sul, Porto Alegre, RS.

BURT, P.; ADELSON, E. The Laplacian Pyramid as a Compact Image Code. **IEEE Transactions on Communications**, New York, v.31, n.4, p.532–540, April 1983.

CONKLIN, G. J.; HEMAMI, S. S. A Comparison of Temporal Scalability Techniques. **IEEE Transactions on Circuits and Systems for Video Technology**, New York, v.9, n.6, p.909–919, Sept. 1999. doi:10.1109/76.785728.

CORRIVEAU, P. et al. All subjective scales are not created equal: the effects of context on different scales. **Signal Processing**, [S.l.], v.77, n.1, p.1–9, August 1999. doi:10.1016/j.physletb.2003.10.071.

DOMANSKI, M. et al. Spatio-temporal scalability for MPEG video coding. **IEEE Transactions on Circuits and Systems for Video Technology**, New York, v.10, n.7, p.1088–1093, Oct. 2000. doi:10.1109/76.875513.

EBU. **SAMVIQ - Subjective Assessment Methodology for Video Quality**. [S.l.], 2003. (BPN 056).

FARIAS, M. C. et al. Digital Television Broadcasting in Brazil. **IEEE Multimedia**, [S.l.], v.15, n.2, p.64–70, Apr.-June 2008.

FURHT, B. A Survey of Multimedia Compression Techniques and Standards. Part I: jpeg standard. **IEEE Transactions on Circuits and Systems for Video Technology**, New York, v.1, n.1, p.49–67, April 1995. doi:10.1006/rtim.1995.1005.

GIROD, B. What's wrong with mean-squared error? **In: Watson, A.B. Digital images and human vision**, Cambridge, MA, USA: MIT Press. p.207–220, 1993.

GIROD, B. et al. Distributed Video Coding. **Proceedings of the IEEE**, New York, v.93, n.1, p.71–83, Jan. 2005.

GONZALEZ, R. C.; WOODS, R. E. **Digital Image Processing**. 2nd ed. Reading, Massachusetts: Addison-Wesley, 1987.

GRAPS, A. An Introduction to Wavelets. **IEEE Computational Science and Engineering**, [S.l.], v.2, n.2, p.50–61, 1995. doi:10.1109/99.388960.

GUO BAO-LONG, D. G. guang. Improvement to Progressive Fine Granularity Scalable Video Coding. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL INTELLIGENCE AND MULTIMEDIA APPLICATIONS, ICCIMA, 2003, Xian, China. **Proceedings...** Los Alamitos: CA: IEEE, 2003. p.249–253. doi:10.1109/ISCAS.2005.1466025.

HSU, C.-H.; HEFEEDA, M. Structuring Multi-Layer Scalable Streams to Maximize Cient-Perceived Quality. In: IEEE INTERNATIONAL WORKSHOP ON QUALITY OF SERVICE, IWQOS, 15., 2007, Evanston, Illinois. **Proceedings...** Piscataway: IEEE, 2007. p.182–187.

HUYNH-THU, Q.; GHANBARI, M. A Comparison of Subjective Video Quality Assessment Methods for Low-Bit Rate and Low-Resolution Video. In: IASTED INTERNATIONAL CONFERENCE ON SIGNAL AND IMAGE PROCESSING, 2005, Honolulu, Hawaii, USA. **Proceedings...** [S.l.: s.n.], 2005. p.70–76.

HUYNH-THU, Q.; GHANBARI, M. Scope of validity of PSNR in image/video quality assessment. **Electronics Letters**, [S.l.], v.44, n.3, p.800–801, June 2008. doi:10.1049/el:20080522.

ITS. **NTIA/ITS Digital Video Quality Metric Takes Top Performing Spot**. Press Release. Disponível em: <http://www.its.bldrdoc.gov/press_releases/>. Acesso em: fev. 2009.

ITU-R. **Recommendation BT.500**: methodology for the subjective assessment of the quality of television pictures. [S.l.: s.n.], 2002.

ITU-R. **Recommendation BT.1683**: objective perceptual video quality measurement techniques for standard definition digital broadcast television in the presence of a full reference. [S.l.: s.n.], 2004.

ITU-T. **Recommentation P.910**: subjective video quality assessment methods for multimedia applications. [S.l.: s.n.], 1999.

ITU-T. **Recommendation J.144**: objective perceptual video quality measurement techniques for digital cable television in the presence of a full reference. [S.l.: s.n.], 2004.

JACK, K. **Video Desmystified**: a handbook for the digital engineer. 4th ed. [S.l.]: Elsevier, 2005.

JSVM. **Software Manual**. [S.l.: s.n.], 2008.

KIM, B.-J. et al. Low bit-rate scalable video coding with 3-D set partitioning in hierarchical trees (3-D SPIHT). **IEEE Transactions on Circuits and Systems for Video Technology**, [S.l.], v.10, n.8, p.1374–1387, Dec. 2000. doi:10.1109/76.889025.

KIM, C. S. et al. Measuring Video Quality on Full Scalability of H.264/AVC Scalable Video Coding. **IEICE Transactions on Communications**, [S.l.], v.E91-B, n.5, p.1269–1278, May 2008. doi:10.1093/ietcom/e91-b.5.1269.

KOZAMERNIK, F. et al. SAMVIQ - A New EBU Methodology for Video Quality Evaluations in Multimedia. **SMPTE Motion Imaging Journal**, [S.l.], v.114, n.4, p.152–160, April 2005.

K.R. RAO, P. Y. **Discrete Cosine Transform–Algorithms, Advantages, Applications**. San Diego, CA, USA: Academic Press, 1990.

KROB, A. et al. ALMTF: adaptive layered multicast tcp-friendly. In: WEBMEDIA, 2007. **Proceedings...** [S.l.: s.n.], 2007. p.9–16.

KWON, G.-I.; BYERS, J. W. Smooth Multirate Multicast Congestion Control. In: ANNUAL JOINT CONFERENCE OF THE IEEE COMPUTER AND COMMUNICATIONS SOCIETIES, IEEE INFOCOM, 22., 2003, San Francisco, CA. **Proceedings...** Piscataway: IEEE, 2003. v.2, n.1, p.1022–1032.

LEGOUT, A.; BIERSACK, E. PLM: fast convergence for cumulative layered multicast transmission schemes. In: ACM SIGMETRICS INTERNATION CONFERENCE ON MEASUREMENT AND MODELING OF COMPUTER SYSTEMS, SIGMETRICS, 2000, Santa Clara, CA. **Proceedings...** New York: ACM, 2000. p.13–22.

LI, B.; LIU, J. Multirate Video Multicast over the Internet: an overview. **IEEE Network**, New York, USA, v.17, n.1, p.24–29, 2003.

LI, J. et al. Generalized multicast congestion control. **Elsevier Computer Networks**, [S.l.], v.51, n.6, p.1421–1443, April 2007.

LI, W. Overview of Fine Granularity Scalability in MPEG-4 Video Standard. **IEEE Transactions on Circuits and Systems for Video Technology**, [S.l.], v.11, n.3, p.301–317, March 2001.

LI, W. et al. Fine Granularity Scalability in MPEG-4 for Streaming Video. In: IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, ISCAS, 2000, Geneva, Switzerland. **Proceedings...** Piscataway: IEEE, 2000. v.1, p.299–302.

LIU, J. et al. Adaptive Video Multicast over the Internet. **IEEE Multimedia**, [S.l.], v.10, n.1, p.22–33, March 2003. doi:10.1109/MMUL.2003.1167919.

MACCANNE, S. et al. Receiver driven layered multicast. In: ACM SIGCOMM, 1996, Stanford, California, USA. **Proceedings...** New York: ACM, 1996. p.117–130.

MARTUCCI, S. A. et al. A zerotree wavelet video coder. **IEEE Transactions on Circuits and Systems for Video Technology**, New York, v.7, n.1, p.109–118, Feb. 1997. doi:10.1109/76.554422.

MCCANNE, S.; VETTERLI, M.; JACOBSON, V. Low-complexity video coding for receiver-driven layered multicast. **IEEE Journal on Selected Areas in Communications**, [S.l.], v.15, n.6, p.983–1001, August 1997. doi:10.1109/49.611154.

MCCARTHY, J. D.; SASSE, M. A.; MIRAS, D. Sharp or smooth? comparing the effects of quantization vs. frame rate for streamed video. In: CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS, SIGCHI, 2004, Vienna, Austria. **Anais...** New York: ACM, 2004. p.535–542. doi:10.1145/985692.985760.

MONTEIRO, J. M.; NUNES, M. S. A Subjective Quality Estimation Tool for the Evaluation of Video Communication Systems. In: IEEE SYMPOSIUM ON COMPUTERS AND COMMUNICATIONS, ISCC, 2007, Santiago, Portugal. **Proceedings...** Piscataway: IEEE, 2007. p.MW–75–MW–80.

NEMETHOVA, O. et al. Subjective Evaluation of Video Quality for H.264 Encoded Sequences. In: SYMPOSIUM ON TRENDS IN COMMUNICATIONS, SYMPOTIC, 2004, Bratislava, Slovakia. **Proceedings...** [S.l.]: IEEE, 2004. p.191–194.

OHM, J.-R. Three-Dimensional Subband Coding With Motion Compensation. **IEEE Transactions on Image Processing**, [S.l.], v.3, n.5, p.559–571, Sept. 1994. doi:10.1109/83.334985.

OHM, J.-R. Advances in Scalable Video Coding. **Proceedings of the IEEE**, New York, USA, v.93, n.1, p.42–56, Jan. 2005.

OSTERMANN, J. et al. Video coding with H.264/AVC: tools, performance, and complexity. **IEEE Circuits and Systems Magazine**, [S.l.], v.4, n.1, p.7–28, First quarter of 2004. doi:10.1109/MCAS.2004.1286980.

OUARET, M.; DUFAUX, F.; EBRAHIMI, T. Codec-Independent Scalable Distributed Video Coding. In: IEEE INTERNATIONAL CONFERENCE ON IMAGE PROCESSING, ICIP, 2007. **Proceedings...** [S.l.: s.n.], 2007. v.3, p.III–9–III–12. doi:10.1109/ICIP.2007.4379233.

PéCHARD, S. et al. Suitable Methodology in Subjective Video Quality Assessment: a resolution dependent paradigm. In: INTERNATIONAL WORKSHOP ON IMAGE MEDIA QUALITY AND ITS APPLICATIONS, IMQA, 2008, Kyoto, Japan. **Proceedings...** [S.l.: s.n.], 2008.

PEREIRA, F.; ALPERT, T. MPEG-4 video subjective test procedures and results. **IEEE Transactions on Circuits and Systems for Video Technology**, New York, v.7, n.1, p.32–51, Feb. 1997. doi:10.1109/76.554416.

PINSON, M.; WOLF, S. Comparing subjective video quality testing methodologies. In: SPIE VIDEO COMMUNICATIONS AND IMAGE PROCESSING CONFERENCE, 2003. **Proceedings...** [S.l.: s.n.], 2003. v.5150, p.573–582. doi:10.1117/12.509908.

PODILCHUK, C. I.; JAYANT, N. S.; FARVARDIN, N. Three-Dimensional Subband Coding of Video. **IEEE Transactions on Image Processing**, [S.l.], v.4, n.2, p.125–139, Feb. 1995. doi:10.1109/83.342187.

PRESTO. **D5.4 - High Quality Compression for Film and Video**. Disponível em: <http://presto.joanneum.ac.at/Public/D5_4.pdf>. Acesso em: fev. 2009.

RIZZO, L. PGMCC: a tcp-friendly single-rate multicast congestion control scheme. In: APPLICATIONS, TECHNOLOGIES, ARCHITECTURES, AND PROTOCOLS FOR COMPUTER COMMUNICATION, 2000. **Proceedings...** New York: ACM, 2000. p.17–28. doi:10.1145/347059.347390.

ROESLER, V. **SAM**: um sistema adaptativo para transmissão e recepção de sinais multimídia em redes de computadores. 2003. Tese (Doutorado em Ciência da Computação) — Instituto de Informática, Universidade Federal do Rio Grande do Sul, Porto Alegre.

SCHäFER, R. et al. MCTF and Scalability Extension of H.264/AVC and its Application to Video Transmission, Storage, and Surveillance. **Proceedings of the SPIE**, Bellingham, Washington, v.5960, p.343–354, July 2005. doi:10.1117/12.631425.

SCHUSTER, B. **Fine granular scalability with wavelets coding**. [S.l.]: ISO/IEC, 1998. (JTC1/SC29/WG11, MPEG98/M4021).

SCHWARZ, H. et al. Analysis of Hierarchical B Pictures and MCTF. In: IEEE INTERNATIONAL CONFERENCE ON MULTIMEDIA AND EXPO, ICME, 2006, Toronto, Canada. **Proceedings...** Piscataway: IEEE, 2006. p.1929–1932.

SCHWARZ, H. et al. Overview of the Scalable Video Coding Extension of the H.264-AVC Standard. **IEEE Transactions on Circuits and Systems for Video Technology**, [S.l.], v.17, n.9, p.1103–1120, Sept 2007.

SCHWARZ, H.; WIEGAND, T. **Implementation and performance of FGS, MGS, and CGS**. [S.l.]: Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, 2007. (Doc. JVT-V126).

SLEPIAN, D.; WOLF, J. Noiseless coding of correlated information sources. **IEEE Transactions on Information Theory**, [S.l.], v.19, n.4, p.471–480, July 1973.

SULLIVAN, G. J.; WIEGAND, T. Video Compression - From Concepts to the H.264/AVC Standard. **Proceedings of the IEEE**, New York, v.93, n.1, p.18–31, January 2005.

TIAN, D.; GABBOUJ, M.; HANNUKSELA, M. M. Sub-sequence video coding for improved temporal scalability. In: IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, ISCAS, 2005. **Proceedings...** Piscataway: IEEE, 2005. v.6, p.6074–6077. doi:10.1109/ISCAS.2005.1466025.

VETRO, A.; SUN, H. Media Conversions to Support Mobile Users. In: CANADIAN CONFERENCE ON ELECTRICAL AND COMPUTER ENGINEERING, CCECE, 2001, Toronto, Canada. **Proceedings...** Piscataway: IEEE, 2001. v.1, p.607–612. doi:10.1109/CCECE.2001.933753.

VICISANO, L. et al. TCP-like congestion control for layered multicast data transfer. In: IEEE INFOCOM, 1998, San Francisco, California, USA. **Proceedings...** New York: IEEE, 1998.

VQEG. **Phase I Subjective Test Plan Version 3**. Disponível em: <ftp://ftp.crc.ca/crc/vqeg/phase1-docs/>. Acesso em: fev. 2009.

VQEG. **Final Report From The Video Quality Experts Group On The Validation Of Objective Models Of Video Quality Assessment**. Disponível em: <ftp://ftp.its.bldrdoc.gov/dist/ituvidq/phase1_final_report/COM-80E.pdf>. Acesso em: fev. 2009.

VQEG. **Final Report From the Video Quality Experts Group on the Validation of Objective Models of Video Quality Assessment, Phase II**. Disponível em: <ftp://ftp.its.bldrdoc.gov/dist/ituvidq/frtv2_final_report/VQEGII_Final_Report.pdf>. Acesso em: fev. 2009.

VQEG. **Multimedia Group Test Plan, Draft Version 1.21**. Disponível em: <ftp://vqeg.its.bldrdoc.gov/Documents/VQEG_Kyoto_Mar08/MeetingFiles/>. Acesso em: fev. 2009.

VQEG. **RRNR-TV Group Test Plan Version 2.1**. Disponível em: <http://www.its.bldrdoc.gov/vqeg/projects/rrnr-tv/>. Acesso em: fev. 2009.

WALLACE, G. K. The JPEG Still Picture Compression Standard. **Commun. ACM**, New York, NY, USA, v.34, n.4, p.30–44, 1991. doi:10.1145/103085.103089.

WANG, Z.; BOVIK, A.; LU, L. Why is image quality assessment so difficult? In: IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, ICASSP, 2002. **Proceedings...** [S.l.: s.n.], 2002. v.4, p.IV–3313– IV–3316. doi:10.1109/ICASSP.2002.1004620.

WANG, Z.; SHEIKH, H. R.; ALAN C. BOVIK. Objective Video Quality Assessment. In: FURHT B.; MARQUES, O. E. **Handbook of Video Databases**: design and applications. Boca Raton, Florida, USA: CRC Press, 2003. p.1041–1078.

WIDMER, J. et al. A Survey on TCP-Friendly Congestion Control. **IEEE Network**, [S.l.], v.15, n.3, p.28–37, May 2001.

WIDMER, J.; HANDLEY, M. Extending equation-based congestion control to multicast applications. In: CONFERENCE ON APPLICATIONS, TECHNOLOGIES, ARCHITECTURES, AND PROTOCOLS FOR COMPUTER COMMUNICATIONS, 2001. **Proceedings...** New York: ACM, 2001. p.275–285. doi:10.1145/383059.383081.

WIEGAND, T.; SULLIVAN, G. J.; BJøNTEGAARD, G.; LUTHRA, A. Overview of the H.264/AVC Video Coding Standard. **IEEE Transactions on Circuits and Systems for Video Technology**, New York, v.13, n.7, p.560–576, July 2003.

WIEN, M.; SCHWARZ, H.; OELBAUM, T. Performance Analysis of SVC. **IEEE Transactions on Circuits and Systems for Video Technology**, New York, v.17, n.9, p.1194–1203, September 2007. doi:10.1109/TCSVT.2007.905530.

WU, F. et al. A framework for efficient progressive fine granularity scalablevideo coding. **IEEE Transactions on Circuits and Systems for Video Technology**, New York, v.11, n.3, p.332–344, Mar. 2001. doi:10.1109/76.911159.

WYNER, A.; ZIV, J. The rate-distortion function for source coding with side information at the decoder. **IEEE Transactions on Information Theory**, [S.l.], v.22, n.1, p.1–10, January 1976.

XIAO, F. **DCT-based Video Quality Evaluation – Final Project for EE392J**. Disponível em: <http://compression.ru/video/quality_measure/vqm.pdf>. Acesso em: fev. 2009.

XU, Q.; XIONG, Z. Layered Wyner-Ziv Video Coding. **IEEE Transactions on Image Processing**, [S.l.], v.15, n.12, p.3791–3803, Dec. 2006.

APÊNDICE A OBJETIVOS PROPOSTOS PARA AVALIAÇÕES SUBJETIVA DE VÍDEO ESCALÁVEL

Este apêndice contém a descrição dos objetivos propostos para realização das avaliações subjetivas de qualidade. Esses objetivos já foram exibidos na seção 3.1, através da tabela 3.1. Como o objetivo escolhido "I: Estabilidade vs. Instabilidade" já foi descrito, aqui serão descritos apenas os outros 4 objetivos.

A.1 II - Avaliação dos métodos de escalabilidade

Resumo: Para determinada banda disponível na rede, qual método (ou combinação de métodos) atinge maior qualidade subjetiva? O aumento na taxa de codificação *sempre* representa aumento na qualidade percebida? O comportamento verificado é o mesmo para todos os métodos de escalabilidade?

Descrição: A figura A.1 mostra um exemplo claro do que se pretende atingir com este objetivo (os números ao lado das linhas indicam o número de quadros por segundo de cada ponto). Esta imagem foi adaptada do trabalho de Monteiro e Nunes (MONTEIRO; NUNES, 2007), onde o gráfico foi gerado para uma sequência de vídeo a partir de fórmulas criadas para simular os resultados que uma avaliação subjetiva de qualidade apresentaria, portanto representa uma técnica objetiva.

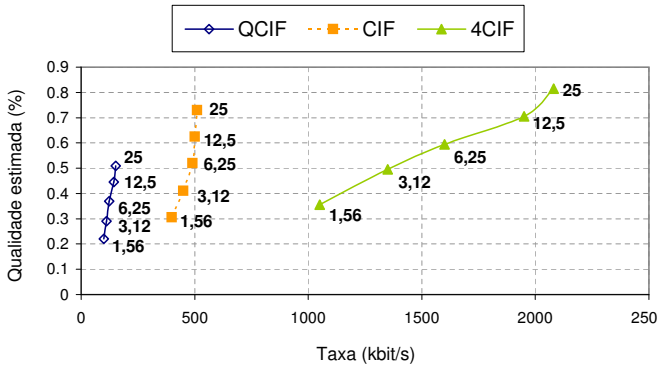


Figura A.1: Gráfico de exemplo dos resultados esperados com o objetivo II.

A codificação escalável possui três métodos principais de criação de vídeos escaláveis e possibilita a combinação destes métodos, gerando assim diversas configurações possíveis para as camadas de vídeo. O aumento no nível dos parâmetros de cada método (aumento da resolução espacial de QCIF para CIF, por exemplo)

normalmente resulta em aumento da taxa de codificação do vídeo, porém, nem sempre o aumento na taxa utilizada representa aumento na qualidade percebida pelos usuários.

Neste objetivo, o pretende-se identificar qual configuração das camadas resulta na melhor qualidade subjetiva para diversas taxas de codificação. Além disso, outras conclusões podem ser buscadas, como:

- Analisar os resultados para cada uma das sequências de vídeo para verificar se as configurações "ótimas" são válidas apenas para sequências específicas, para determinados grupos de sequências ou se são válidas de maneira geral;
- Verificar qual método de escalabilidade tem maior impacto sobre a qualidade do vídeo.

Metodologia proposta: Os vídeos das avaliações são codificados utilizando as escalabilidades espacial, temporal e, possivelmente, de qualidade. A tabela A.1 mostra os valores dos parâmetros utilizados em cada método de escalabilidade.

Tabela A.1: Parâmetros propostos para a codificação no objetivo II.

Método	**Valores**
Espacial	SQCIF (128x96), QCIF (176x144), CIF (352x288), 4CIF (704x576)
Temporal	30, 15, 7.5, 3.25, 1.125 fps ou 25, 12.5, 6.25, 3.125, 1.5625 fps
Qualidade	2 pontos definidos para cada uma das combinações de escalabilidade temporal e espacial: Q1: taxa igual a taxa de codificação total Q2: 3/4 da taxa de codificação total

A escalabilidade de qualidade é diretamente relacionada à medida sinal-ruído das imagens e, no SVC, as camadas de qualidade são definidas através do valor da taxa de codificação na qual devem ser codificadas. Se a taxa especificada for menor do que a necessária para codificar toda a camada, os coeficientes menos importantes passam a ser descartados até que se atinja a taxa especificada. Quanto maior a taxa disponível para determinada camada, mais dados serão codificados e maior será o SNR das imagens. Portanto, é esperado que, com o aumento da taxa de cada camada e sem modificações nas dimensões temporais e espaciais, o SNR aumentará e, consequentemente, a qualidade subjetiva também.

Seguindo este raciocínio, o aumento da qualidade na escalabilidade de qualidade tende a ser mais linear em relação ao aumento da taxa de codificação do que nas escalabilidades espacial e temporal. Portanto, são utilizadas principalmente as escalabilidades temporal e espacial, e, possivelmente, apenas dois pontos de escalabilidade de qualidade, exibidos na tabela A.1.

Cada vídeo é codificado com sua maior resolução temporal e espacial e, a partir deste vídeo, são extraídas todas as outras configurações. A figura A.2 mostra um

exemplo da decomposição de um vídeo na resolução 4CIF utilizando 30 quadros por segundo, a partir do qual são gerados 20 vídeos no total. Com o uso da escalabilidade de qualidade, cada um deles é separado em dois grupos, onde Q1 é o vídeo gerado na figura A.2, sem nenhuma posterior redução de qualidade, e Q2 é o mesmo vídeo com redução de qualidade até atingir 3/4 da taxa de codificação que ele apresentava.

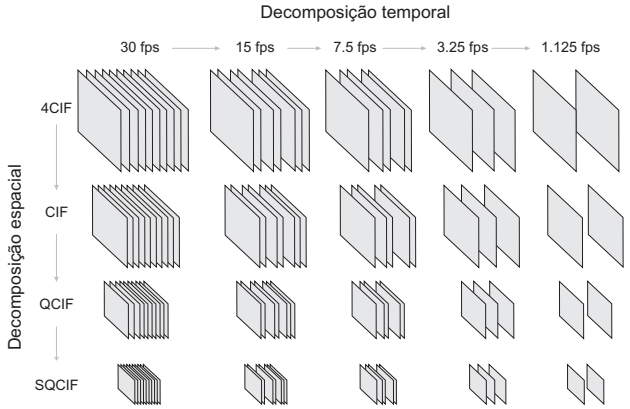


Figura A.2: Exemplo de decomposição temporal e espacial de um vídeo.

Utilizando 4 resoluções espaciais e 5 resoluções temporais, com 8 vídeos de 9 segundos, a duração total de exibição de todos os vídeos seria em torno de 24 minutos. Para estas avaliações a metodologia utilizada seria a ACR ou SAMVIQ, portanto a duração total dos testes ficaria em torno de 35 minutos (possivelmente mais longa no SAMVIQ).

A.2 III - Avaliação dos métodos de escalabilidade com variação nas camadas

Resumo: Qual método apresenta melhor qualidade quando ocorrem variações durante a exibição do vídeo? Alterações na dimensão espacial, temporal ou na qualidade (PSNR) do vídeo tem maior influência na qualidade percebida pelos usuários?

Descrição: Este objetivo é semelhante aos objetivos I e II, porém, não será verificada qual combinação de métodos atinge melhor qualidade para determinada taxa de codificação, mas sim qual método de escalabilidade atinge melhor qualidade quando existem variações no número de camadas utilizadas ao longo da exibição do vídeo.

A variação no número de camadas que o receptor esta recebendo depende de diversos fatores, que são analisados pelos algoritmos de adaptabilidade e controle de congestionamento. Caso o receptor tenha limitações em relação à resolução espacial e/ou resolução temporal, ele obviamente só poderá receber as camadas que transportam vídeo com as resoluções que ele suporta. Porém, caso não existam limitações, o número de camadas que ele receberá depende da banda de rede que foi estimada, e esta banda pode sofrer alterações devido à, principalmente, os tráfegos concorrentes. Com alterações na banda estimada, o número de camadas utilizadas pelo receptor pode variar.

Em momentos em que existem alterações, a variação do número de camadas utilizadas pode afetar a qualidade do vídeo de diferentes maneiras conforme o método

de escalabilidade utilizado para criação destas camadas. A preferência dos usuários é a variação do número de quadros por segundo, da resolução espacial ou da medida SNR das imagens?

Metodologia proposta: São utilizadas 5 camadas, onde os vídeos são codificados seguindo 6 padrões de variação das camadas *para cada um* dos 3 métodos de escalabilidade. Para abranger diversas possibilidades de variação das camadas, são utilizadas as 6 configurações descritas abaixo e ilustradas na figura A.3:

- 2 constantes nas camadas 3 e 5;
- 1 com variação da primeira para a última camada (crescente) e 1 com variação da última para a primeira camada (decrescente);
- 2 com pouca variação, oscilando entre 2 ou 3 camadas.

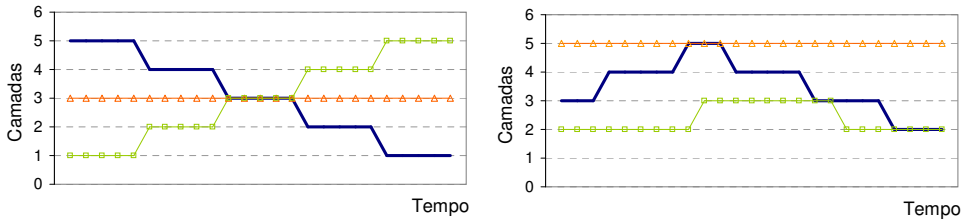


Figura A.3: Cada linha nos gráficos representa um dos 6 padrões de variação de camadas para o objetivo III.

Com as configurações citadas, é possível verificar a relação entre variações lentas e rápidas, fluxos constantes e fluxos com variação, e variações crescentes e decrescentes. Um número maior de configurações é inviável devido à utilização dos 3 métodos de escalabilidade, totalizando 18 configurações que vão ser utilizadas para codificar cada um dos vídeos.

Utilizando estas 18 configurações, com 8 vídeos de 10 segundos, a duração total de exibição de todos os vídeos seria em torno de 24 minutos. A metodologia utilizada seria a ACR, SSCQS ou SAMVIQ, portanto a duração total dos testes ficaria em torno de 35 minutos (possivelmente mais no SAMVIQ). Por apresentar variações nas camadas, é interessante o uso da SSCQS, que permite a alteração contínua do voto atribuído conforme o vídeo vai sendo exibido. Assim, é possível verificar o impacto observado pelo usuário na qualidade do vídeo quando as camadas são alteradas e comparar esse impacto entre os métodos de escalabilidade.

Para aumentar o número de configurações, é possível utilizar apenas os métodos de escalabilidade espacial e temporal. Assim, poderiam ser utilizadas 9 formas de variação das camadas para cada um dos métodos e seria mantido o tempo de execução dos testes citado anteriormente.

Para haver justiça nos testes, as camadas utilizadas para cada método devem ser equivalentes em relação à banda utilizada para codificação e transmissão. Inicialmente são escolhidas as resoluções para o método de escalabilidade espacial de acordo com as resoluções padrões utilizadas, provavelmente 4CIF, CIF, QCIF e SQCIF. Os parâmetros para os outros métodos de escalabilidade são definidos de acordo com a taxa de codificação dessas camadas.

A.3 IV - Quantidade de camadas

Resumo: Definição de um número de camadas que satisfaça os usuários. *Poucas* camadas geram variações bruscas, menor flutuação e pior aproveitamento de banda, enquanto *muitas* camadas podem gerar sobrecarga desnecessária e um grande número de variações entre as camadas, mas aproveitar melhor a banda disponível.

Descrição: Os métodos de escalabilidade do SVC permitem a adaptação da transmissão para uma grande quantidade de dispositivos através da variação na resolução espacial, temporal e qualidade SNR. Esta adaptação pode ser feita para um simples receptor ou múltiplos receptores, podendo também pode ser utilizada em sistemas de transmissão em camadas. Devido a esta granularidade fina que os métodos de escalabilidade permitem, é possível a criação de diversas camadas de vídeo, fazendo com que as transições entre elas sejam graduais e melhorem a experiência de visualização do usuário.

Apesar do uso de diversas camadas com granularidade fina aparentemente representar a melhor escolha, as flutuações na banda disponível para os receptores pode resultar em cenários que favoreçam o uso de um número menor de camadas em favor de uma maior estabilidade na transmissão.

Em sistemas de transmissão em camadas, a quantidade de camadas utilizadas está diretamente relacionada com a granularidade da adaptação, a justiça com outros tráfegos e a estabilidade da transmissão. Um número grande de camadas aumenta as possibilidades de adaptação aos receptores e possibilita um ajuste mais adequado da banda utilizada para aproveitar com maior eficiência a banda disponível. Por favorecer a adaptação, um número grande de camadas também auxilia os mecanismos a garantir a equidade de banda com tráfegos concorrentes (*fairness*) (LI; LIU, 2003). Além disso, a utilização de várias camadas permite transições graduais na qualidade do vídeo, reduzindo o impacto que a troca de camadas exerce sobre a visualização. Apesar de todas essas vantagens, a utilização de várias camadas aumenta a instabilidade do sistema e necessita diversas operações de *join* e *leave* se utilizadas em um ambiente multicast, que são consideradas custosas e necessitam certo tempo para terem efeito.

Por outro lado, a utilização de um número menor de camadas favorece a estabilidade do sistema e não requer tantas operações de *join* e *leave*, porém, reduz a adaptabilidade, dificulta a manutenção da equidade de banda com tráfegos concorrentes e resulta em alterações mais bruscas na qualidade do vídeo. Em relação à equidade de banda, utilizando de 3 a 5 camadas já é possível se ter uma boa adaptabilidade e garantir que exista equidade. Um número maior de camadas não necessariamente oferecerá melhores condições (LI; LIU, 2003).

A figura A.4 mostra quatro exemplos de cenários que podem ser considerados para a avaliação. A linha tracejada (vermelha) mostra uma simulação da variação da banda estimada pelos algoritmos que controlam a recepção dos dados. A linha escura mostra a variação das camadas em um sistema com 9 camadas (sistema A - escala da esquerda dos gráficos) e a linha clara mostra a variação em um sistema com apenas 3 camadas (sistema B - escala da direita). Para que ocorra uma mudança de camada no sistema B, a banda deve variar cerca de 4 vezes o necessário para que ocorra uma mudança no sistema A.

Os gráficos (a) e (c) da figura A.4 mostram variações lentas na banda, que provocam alterações nas camadas dos dois sistemas. As variações em A são lentas, enquanto em B são mais bruscas. figura A.4 (b) mostra um cenário onde as variações na banda provocam diversas alterações nas camadas do sistema A, mas não são suficientes para alterar as camadas do sistema B, que permanece estável. Já no gráfico (d), ocorrem alterações mínimas na banda, mas que provocam alterações em ambos os sistemas.

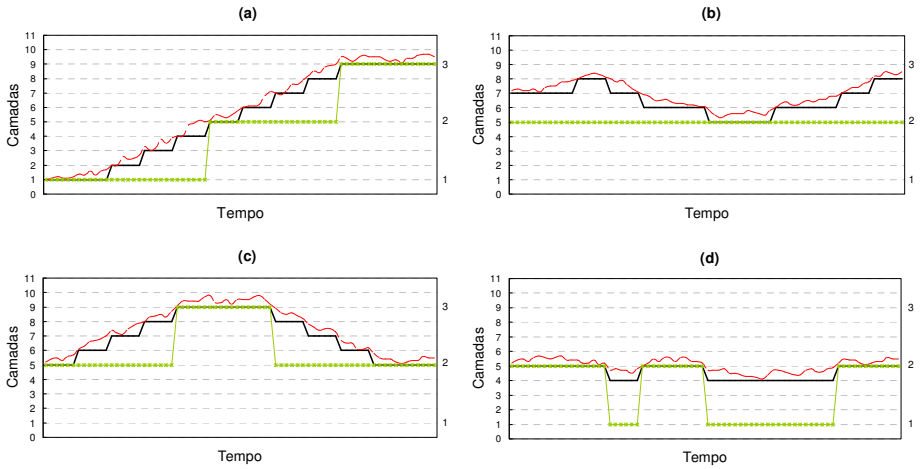


Figura A.4: Cenários para comparação de sistemas com diferente número de camadas no objetivo IV.

Metodologia proposta: São utilizados dois sistemas hipotéticos para geração dos vídeos analisados: o sistema A, com 9 camadas, e o sistema B, com 3 camadas, já exibidos na figura A.4. O número de camadas de B foi escolhido para ser o menor possível, mas que ainda possibilite certa variedade de cenários para adaptabilidade (um número de camadas entre 3 e 5 já permite boa adaptabilidade (LI; LIU, 2003)), enquanto o número de camadas de A foi escolhido de modo a ser bastante superior a B, mas pequeno o suficiente para que fosse possível realizar a variação de todas as camadas em um período de 10 segundos (provável duração dos vídeos nas avaliações).

Os cenários, ou configurações, utilizados para codificar os vídeos são separados em 3 categorias: pouca, média e bastante variação de banda. São utilizadas 8 configurações distribuídas dentro destas 3 categorias, sendo que em 4 configurações as camadas de B se mantêm constantes e em outras 4 B sofre variações. Essas configurações procuram abranger um grande número de cenários, mas sem favorecer nenhum dos sistemas. Abaixo são descritas as 8 configurações de acordo com suas categorias:

- **Pouca variação:** Variação de até 3 camadas no sistema A. Duas configurações utilizadas, uma com o sistema B estável na camada 1 e outra com o mesmo sistema estável na camada 2, seguindo padrões diferentes de alteração na banda e provocando alterações de 3 camadas no sistema A.
- **Variação média:** Variação de 4 a 7 camadas do sistema A. São utilizadas 4 configurações, 2 com variação de 4 camadas em A, com B estável, e outras 4 com variação de 6 camadas em A, com uma variação em B.

- **Bastante variação:** Variação de todas as camadas em ambos os sistemas. Duas configurações utilizadas, uma com a banda crescente e uma decrescente. Na crescente, a avaliação é feita da camada 1 até a camada 9 no sistema A e da camada 1 até a camada 3 no sistema B. Na decrescente, a variação ocorre da camada 9 até a camada 1 no sistema A e da camada 3 até a camada 1 no sistema B.

Utilizando as 8 configurações citadas (para cada um dos dois sistemas) e 10 vídeos com duração de 10 segundos cada, a duração aproximada de exibição de todos os vídeos fica em 22 minutos. As prováveis metodologias para uso nestas avaliações são ACR, SSCQS e SAMVIQ. Como a duração de exibição dos vídeos é relativamente pequena, é possível que seja utilizada a SAMVIQ sem que as avaliações sejam demasiadamente longas.

Para configuração das camadas, são utilizados os três conceitos de escalabilidade. Para existir justiça nas comparações entre os sistemas, são utilizadas camadas com configurações compatíveis, descritas na tabela A.2. A tabela exibe as dimensões temporal e espacial das camadas e, aquelas com o indicador MGS, são camadas criadas com o uso da técnica MGS para redução da taxa de codificação e, consequentemente, da qualidade.

Tabela A.2: Configuração das camadas para codificação no objetivo IV.

Camada em A	**Configuração**	**Camada em B**
1	QCIF 15 fps	1
2	QCIF 30 fps - MGS	
3	QCIF 30 fps	
4	CIF 15 fps	
5	CIF 30 fps - MGS	2
6	CIF 30 fps	
7	4CIF 15 fps	
8	4CIF 30 fps - MGS	
9	4CIF 30 fps	3

A.4 V - MGS vs. Escalabilidade espacial

Resumo: Mantendo-se a mesma dimensão espacial durante a exibição dos vídeos, os dois métodos apresentam variações apenas na qualidade das imagens. Qual apresenta melhor qualidade subjetiva?

Descrição: As técnicas de escalabilidade espacial têm como objetivo reduzir a dimensão espacial do vídeo para adaptar a transmissão a receptores com capacidades limitadas, como dispositivos móveis, por exemplo. Com a redução da dimensão, também é reduzida a banda necessária para transmissão do vídeo, portanto a escalabilidade

espacial também pode ser utilizada simplesmente para redução da banda e adaptação do vídeo para receptores com menor capacidade de banda.

O MGS (*Medium Grain Scalability*) do H.264 SVC é uma técnica de escalabilidade de qualidade que reduz o número de coeficientes utilizados nas imagens a fim de adaptar o vídeo à taxa de codificação especificada. O H.264 SVC também apresenta outra forma de escalabilidade de qualidade chamada CGS (*Coarse Grain Scalability*), que é realizada com a modificação no grau de quantização das imagens e tem o processo de codificação igual ao da escalabilidade espacial, porém sem as etapas de redução e ampliação da resolução espacial. O CGS permite a criação de um determinado número de camadas apenas durante a codificação, enquanto o MGS permite a adaptação de cada camada para diversas taxas de bits que possam ser necessárias, mesmo após a codificação.

Apesar das diferenças nos objetivos das duas técnicas, em um cenário onde não é necessária a adaptação da resolução espacial do vídeo, mas é necessária a escalabilidade para transmissão em camadas, tanto escalabilidade espacial quanto MGS podem ser utilizadas para redução da taxa de codificação e criação das camadas.

A figura A.5 mostra um exemplo de configurações que podem ser utilizadas durante os testes. A coluna C1 mostra um vídeo codificado em três camadas utilizando escalabilidade espacial, onde as camadas estão na resolução QCIF, CIF e 4CIF. Na coluna C2, o mesmo vídeo é codificado utilizando escalabilidade de qualidade MGS. O vídeo é inicialmente codificado na maior resolução que foi utilizada durante a codificação em C1 e então são extraídas as três camadas conforme a taxa de codificação utilizada para codificar cada uma das três camadas em C1. Para exibição dos vídeos, todos devem utilizar a mesma resolução, que é a maior utilizada durante a codificação (4CIF no exemplo). Os vídeos utilizando apenas as camadas inferiores de C1 obviamente precisariam ser redimensionados durante a exibição. Com as camadas codificadas apresentando a mesma taxa de codificação, é possível comparar os dois métodos em relação à qualidade exibida ao usuário.

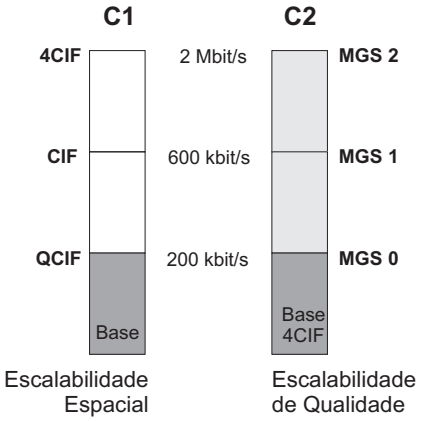


Figura A.5: Comparação entre camadas criadas com MGS e com escalabilidade espacial.

Metodologia proposta: O SVC possibilita a codificação das camadas espaciais em resoluções arbitrárias, com a única restrição de que nem a resolução horizontal nem a vertical podem ser reduzidas de uma camada para a próxima, enquanto o MGS possibilita a extração de diversas taxas de bits para as camadas.

Apesar da possibilidade de se utilizar diversas resoluções espaciais, é interessante utilizar resoluções padronizadas, que são utilizadas como base em inúmeros dispositivos, sistemas, avaliações, etc. Como exibido na figura A.5, a maior resolução utilizada seria 4CIF, sendo reduzida para CIF, QCIF e, possivelmente, para SQCIF.

Utilizando estas quatro resoluções espaciais, são necessárias 8 configurações para codificação dos vídeos, 4 para a escalabilidade espacial e 4 para a escalabilidade de qualidade. As taxas de codificação para a criação das camadas na escalabilidade de qualidade serão definidas conforme as taxas necessárias para a codificação das camadas de escalabilidade espacial. Com 8 configurações e utilizando 12 vídeos de 10 segundos, o tempo total de duração da exibição de todos os vídeos é exatamente 16 minutos.

Outra opção é não utilizar uma resolução tão baixa quanto SQCIF e utilizar apenas 4CIF, CIF e QCIF. Com uma camada base de maior resolução, é possível que o método de escalabilidade espacial apresente melhores resultados em relação à qualidade.

Utilizando as 8 configurações iniciais mais as 6 criadas com este segundo caso, são geradas 14 configurações. Utilizando 12 vídeos de 10 segundos, o tempo total de exibição fica em torno de 28 minutos. Com todas essas configurações, é possível comparar o MGS com a técnica de escalabilidade espacial e também verificar a influência da camada base em relação à codificação espacial.

APÊNDICE B APLICATIVOS DESENVOLVIDOS

Este apêndice contém a descrição dos aplicativos criados para utilização neste trabalho. Três aplicativos foram desenvolvidos para executar diversas tarefas e eles serão descritos juntamente com algumas informações sobre sua utilidade e a forma como devem ser utilizados.

Os três aplicativos foram desenvolvidos com a linguagem C++ para a plataforma Windows XP. Eles foram criados especificamente para as necessidades encontradas durante o desenvolvimento do trabalho, entre elas o cálculo das medidas TI e SI (ver seção 3.3.3), simulação da instabilidade (ver seção 3.3.5) e, principalmente, a execução das avaliações subjetivas (ver seção 3.4).

Os aplicativos, arquivos de exemplo para utilização deles e código fonte de alguns estão disponíveis em <http://www.inf.ufrgs.br/~lcdaronco/dissertacao/>.

B.1 TI & SI

Este aplicativo calcula as medidas TI (*Temporal Information*) e SI (*Spatial Information*) de um vídeo, de acordo com as fórmulas descritas na norma BT.500 (ITU-R, 2002) e já apresentadas na seção 3.3.3.

O aplicativo recebe um vídeo como entrada e mostra na saída padrão as informações sobre os valores de TI e SI de cada quadro do vídeo. Com a opção *minimal* habilitada, o aplicativo retorna apenas os valores máximo (valor normalmente utilizado), mínimo e médio do TI e SI do vídeo como um todo. Ele ainda possui a opção de indicar o número de pixels que devem ser removidos de cada lado (incluindo partes superior e inferior) dos quadros antes de realizar os cálculos (como é recomendado na BT.500 que sejam removidas colunas de 20 pixels das laterais e linhas de 20 pixels da parte superior e inferior).

O aplicativo trabalha com vídeos de entrada no formato YUV com amostragem 4:2:0, no mesmo formato utilizado pelos aplicativos do JSVM. Abaixo é descrita a maneira de funcionamento do aplicativo, chamado *tisi*:

```
tisi <WIDTH> <HEIGHT> <FILENAME> [-m] [-crop]
           <WIDTH>: Largura dos quadros
          <HEIGHT>: Altura dos quadros
        <FILENAME>: Nome do arquivo de vídeo de entrada
                -m: Minimal. Saída do programa:
                    <FILENAME> SI TI SI(min) TI(min) SI(média) TI(média)
             -crop: Reduzir quadros
                    <VALUE>: Número de pixels a serem cortados
```

Comando:

```
tisi 704 576 video.yuv -m 20
```

B.2 LYUV

Esta ferramenta foi criada, inicialmente, para processar vídeos em camadas (***L**ayered **YUV**)*, mas foi expandida para efetuar outras operações que tornaram-se necessárias. O aplicativo também trabalha com vídeos de entrada no formato YUV com amostragem 4:2:0, no mesmo formato utilizado pelos aplicativos do JSVM. Ele apresenta 7 operações, que serão descritas a partir das informações de uso do programa exibidas abaixo:

```
lyuv -<option> <params>
          Encode: -e <config> <output> [-n <width> <height> <fps>]
          Decode: -d <config> <input>
            Info: -i <file>
          Expand: -x <input> <output> <#frames> <width> <height>
             Cut: -t <input> <output> <#frames> <width> <height>
         Convert: -c <input> <width> <height> <fps> <output> <width>
                     <height> <fps>
         Get rate from log: -l <log> <output> <#frames cut> <tlayer>
                               <slayer> <qlayer> <fps>
```

Encode (-e): Esta operação é utilizada para unir diversos vídeos em um arquivo único. O objetivo principal é unir vídeos que representam diversas camadas de um mesmo vídeo fonte para simular a variação nas camadas ao longo da exibição do vídeo.

A operação deve receber como entrada o nome de um arquivo de saída `<output>` e o arquivo de configuração `<config>`, que segue o formato do exemplo abaixo:

```
vidA.yuv 704 576 15 30
vidB.yuv 352 288 30 90
vidA.yuv 704 576 15 15
vidC.yuv 352 288 30 60
```

Cada linha do arquivo representa um vídeo de entrada e as colunas são os campos: (nome do arquivo) (largura) (altura) (fps) (número de quadros). Seguindo o arquivo de exemplo, o programa pegará os primeiros 30 quadros do primeiro vídeo (“vidA.yuv”) e os colocará na saída. Esses 30 quadros representam 2 segundos de vídeo, pois o primeiro vídeo está armazenado à 15 quadros por segundo. Após isso, o programa pegará 90 quadros do segundo vídeo (“vidB.yuv”), começando na marca de 2 segundos, e os adicionará no arquivo de saída. O processo segue até que todos os vídeos do arquivo de configuração sejam utilizados.

A operação ainda apresenta a possibilidade de escolher entre a normalização dos vídeos ou a inclusão de um cabeçalho. Quando a opção `-n` é utilizada, os vídeos serão normalizados para a resolução e fps especificados pelos parâmetros `<width>`, `<height>` e `<fps>`. Com o arquivo de exemplo, uma possibilidade seria normalizar os vídeos para a resolução 704x576 com 30 fps. Para a normalização, é utilizada a operação “Convert”, que será explicada na sequência deste apêndice.

Quando a opção `-n` *não* é utilizada, os vídeos não são normalizados, ou seja, serão armazenados no arquivo de saída com a resolução espacial e temporal exatamente como especificada no arquivo de configuração. Para identificar as resoluções de cada vídeo e em que ponto eles iniciam, é incluído um cabeçalho no arquivo de saída, que segue o formato da figura B.1.

O cabeçalho é formado por 4 bytes com os caracteres "LYUV", seguidos por 2 bytes que identificam o número de vídeos utilizados para gerar o arquivo de saída (valor inteiro). Após estes campos, são utilizados 10 bytes para descrição de cada vídeo de

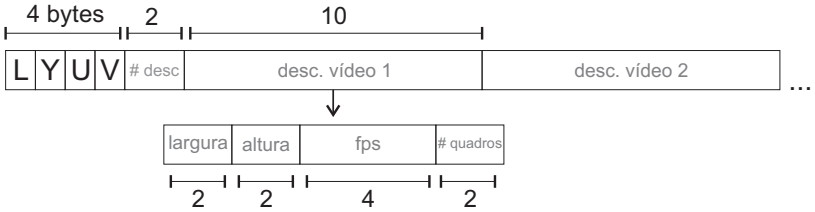


Figura B.1: Cabeçalho incluído pela operação “Encode”.

entrada. Estes 10 bytes são divididos em: 2 bytes para a largura do vídeo (inteiro), 2 bytes para a altura (inteiro), 4 bytes para o número de quadros por segundo (ponto flutuante) e 2 bytes para a quantidade de quadros (inteiro).

Decode (-d): Efetua a operação inversa da “Encode”, ou seja, recebe o vídeo de entrada e o separa em diversos arquivos de saída. Só pode ser utilizada com vídeos que contém o cabeçalho inserido pela operação “Encode”.

Info (-i): Mostra as informações sobre o vídeo de entrada `<file>` que foi previamente criado com a operação “Encode”. Só pode ser utilizada com vídeos que contém o cabeçalho inserido pela operação “Encode”.

Expand (-x): Inclui um determinado número de quadros no início e no fim do vídeo de entrada. Recebe como entradas um vídeo `<input>`, um vídeo de saída `<output>`, o número de quadros que devem ser inseridos no início e no fim `<#frames>`, a largura dos quadros `<width>` e a altura dos quadros `<height>`.

Se o número de quadros informado for 30, por exemplo, os primeiros 30 quadros do vídeo de entrada serão copiados, a ordem de exibição deles será invertida e eles serão inseridos no início do vídeo. O mesmo é feito para os 30 últimos quadros do vídeo de entrada, que são copiados, têm sua ordem invertida e então são inseridos no final do vídeo.

Cut (-t): É a operação inversa da “Expand”. Recebe os mesmos parâmetros de entrada e remove `<#frames>` quadros do início e do final do vídeo de entrada.

Convert (-c): Converte um vídeo de determinada resolução espacial e/ou temporal para outra resolução espacial e/ou temporal. Este programa só é utilizado para aumento das resoluções e não redução. Os métodos de ampliação da resolução espacial e temporal são bastante simples: replicação de pixels e replicação de frames, como já comentado na seção 3.3.5.

A operação já foi testada para ampliar vídeos de QCIF para CIF e de QCIF ou CIF para 4CIF. Em relação à ampliação temporal, qualquer vídeo com taxa de quadros por segundo f pode ser convertido para uma taxa $f * a$, contanto que a seja um valor inteiro. Ele já foi testado para conversão de taxas como 3,75 fps para 7,5, 15 e 30 fps, por exemplo.

Os parâmetros para utilização desta operação são, em ordem de uso: o vídeo de entrada `<input>`, largura do vídeo de entrada `<width>`, altura do vídeo de entrada `<height>`, fps do quadro de entrada `<fps>`, vídeo de saída `<output>`, largura do vídeo de saída `<width>`, altura do vídeo de saída `<height>` e fps do quadro de saída `<fps>`.

Get rate from log (-l): Esta operação calcula a taxa de codificação de um vídeo codificado a partir dos arquivos de registro (logs) gerados durante o processo de codificação com o JSVM. Ela foi criada pois era necessário calcular a taxa dos vídeos removendo os quadros iniciais e finais que eram incluídos antes da codificação. Como esses quadros só são removidos após a decodificação, não era possível calcular a taxa do vídeo codificado a partir da *bitstream* (como normalmente é feito) sem considerar esses quadros. Com esta operação, os quadros são removidos do cálculo através da análise dos logs de codificação.

Um arquivo de log do codificador do JSVM é semelhante ao trecho de arquivo abaixo:

```
AU      0: I    T0 L0 Q0   QP 23   Y 42.0312  U 43.7317  V 44.4531     24584 bit
        0: I    T0 L1 Q0   QP 22   Y 42.9563  U 45.3406  V 46.0686     72272 bit
        0: I    T0 L2 Q0   QP 22   Y 43.3267  U 47.1178  V 47.8441    151784 bit
AU     16: P    T0 L0 Q0   QP 23   Y 42.1224  U 43.9963  V 44.7373      6728 bit
       16: P    T0 L1 Q0   QP 22   Y 43.3104  U 45.9052  V 46.5784     28264 bit
```

O programa utiliza a última coluna de cada linha, que contém o número de bits utilizados por cada quadro, para calcular a taxa de codificação do vídeo. Os primeiros parâmetros passados ao programa são o nome do arquivo de log `<log>` e o nome do arquivo de saída `<output>`, onde será incluída uma linha contendo o valor da taxa calculada para o vídeo. O parâmetro `<#frames cut>` indica o número de quadros que devem ser ignorados do início e no fim do arquivo de log. Como o arquivo de log contém informações de todos os quadros de cada camada do vídeo que foi codificado, os parâmetros `<tlayer>`, `<slayer>` e `<qlayer>` são utilizados para especificar a camada temporal, espacial e de qualidade, respectivamente, que devem ser utilizadas para cálculo da taxa. Por fim, o parâmetro `<fps>` é utilizado para informar o número de quadros por segundo do vídeo.

B.3 wxSVQ

wxSVQ é o nome dado ao aplicativo que foi desenvolvido para execução das avaliações subjetivas. É este aplicativo que exibe os vídeos aos avaliadores e coleta os votos atribuídos. Ele foi desenvolvido com a linguagem C++ na plataforma Windows XP e utiliza a biblioteca SDL (*Simple DirectMedia Layer*) para exibição dos vídeos e a biblioteca wxWidgets para construção da interface gráfica.

O aplicativo suporta as metodologias ACR, que foi utilizada nas avaliações deste trabalho, e também a SAMVIQ. Para ambas as metodologias, o aplicativo recebe como entrada um arquivo de configuração que especifica os vídeos que devem ser utilizados, o nome do SRC e o nome do HRC deste vídeo. Os nomes do SRC e do HRC de cada vídeo são utilizados para posicioná-los de forma adequada ao longo da avaliação. No caso da metodologia ACR, os vídeos são simplesmente dispostos em ordem aleatória, evitando que dois ou mais vídeos do mesmo SRC sejam exibidos em sequência. Na metodologia SAMVIQ a ordenação é mais elaborada, seguindo os seguintes passos:

- os vídeos são divididos em sessões contendo 10 vídeos (no máximo), incluindo uma referência explícita e uma implícita (as referências são identificadas no arquivo de configuração por conterem o valor do HRC igual a “ref”);
- Cada sessão deve conter vídeos de apenas um SRC, ou seja, não pode ser utilizado mais de um SRC em uma mesma sessão;

- Se algum SRC conter mais do que 10 vídeos, eles serão divididos igualmente em um número maior de sessões conforme necessário. Essas sessões são exibidas em sequência durante a avaliação;

- As sessões criadas são dispostas em ordem aleatória, assim como os vídeos de dentro de cada sessão. Esta ordem aleatória é criada cada vez que o aplicativo é executado.

Entre as características da aplicação, ela permite a continuação de uma avaliação através dos arquivos de log que são gravados à cada etapa da avaliação, arquivos que também permitem a verificação dos momentos em que cada quadro de cada vídeo foi exibido, para permitir verificar se os vídeos foram exibidos com a taxa de quadros por segundo correta.

As figuras B.2 e B.3 mostram exemplos de telas da aplicação executando a metodologia ACR e a metodologia SAMVIQ, respectivamente.

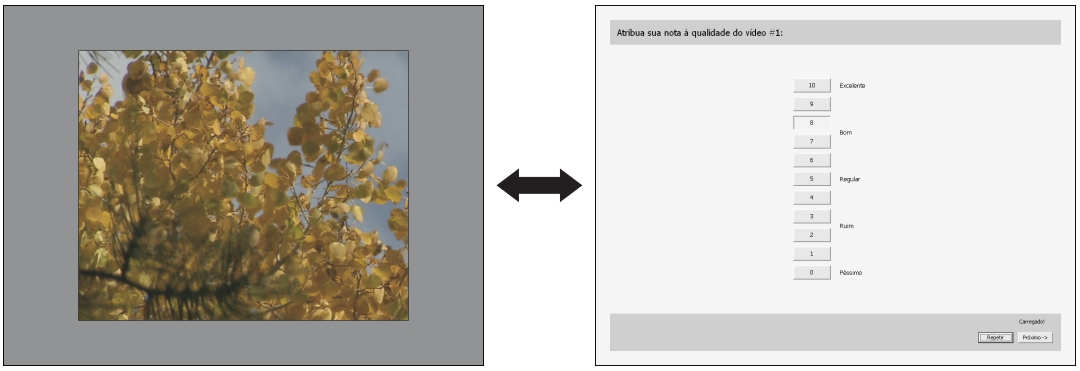


Figura B.2: Exemplo de telas do aplicativo *wxSVQ* com a metodologia ACR.

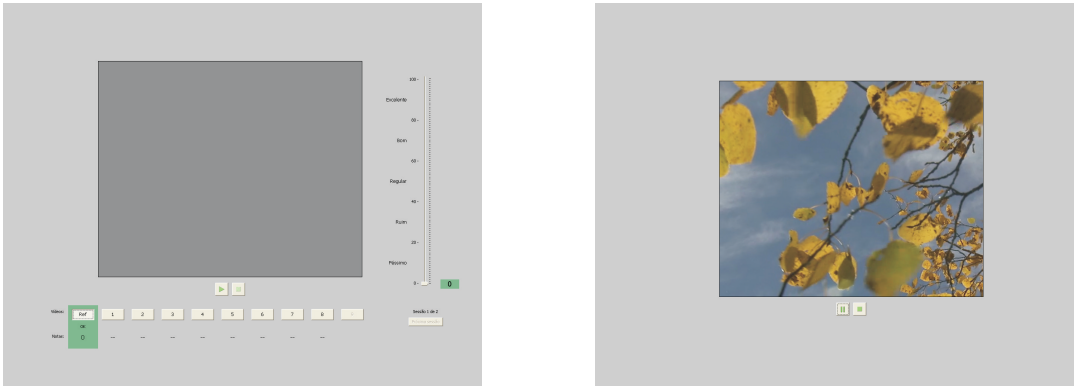


Figura B.3: Exemplo de telas do aplicativo *wxSVQ* com a metodologia SAMVIQ.

APÊNDICE C PROCESSAMENTO DOS VÍDEOS

C.1 Pré-processamento

O pré-processamento dos vídeos foi realizado com o uso de três ferramentas: AviSynth (versão 2.5), VirtualDub (versão 1.7.8) e FFmpeg (versão SVN-r11870, fevereiro de 2008). As operações realizadas em cada um deles são exibidas na tabela C.1 e o modo de utilização de cada ferramenta será descrito nas seções a seguir. Além do uso dessas ferramentas, também faz parte do pré-processamento dos vídeos a redução espacial da resolução 4CIF para CIF e QCIF, que foi realizada com as ferramentas do JSVM. Este processo já foi comentado na seção 3.3.4 e o funcionamento prático é descrito no apêndice C.2, juntamente com a descrição do uso das outras ferramentas do JSVM.

Os arquivos de configuração e exemplos aqui citados também podem ser encontrados na Internet em: <http://www.inf.ufrgs.br/~lcdaronco/dissertacao>

Tabela C.1: Ferramentas utilizadas e operações realizadas no pré-processamento dos vídeos.

Ferramenta	**Operações**
AviSynth	- Corte de áreas (*crop*) - Redimensionamento espacial - Redimensionamento temporal (redução para 30 fps) - Corte temporal (remoção de quadros) - Conversão de entrelaçado para progressivo
VirtualDub	- Processar o arquivo de configuração do AviSynth - Gravação em AVI no formato 4:2:0 YCbCr (YV12/I420)
FFmpeg	- Remoção dos cabeçalhos do arquivo AVI

C.1.1 AviSynth

O AviSynth é uma ferramenta que trabalha como um servidor de *frames* (um *frameserver*) e que é controlada através de arquivos de configuração, os *scripts*. O uso do AviSynth é bastante simples: é criado um *script* e este arquivo deve ser aberto por um programa que suporte arquivos AVI. Quando o *script* for aberto, o AviSynth fará o processamento especificado neste arquivo e alimentará a aplicação que abriu o *script* com o vídeo resultante deste processamento. Para a aplicação que executou o *script*, este vídeo

é visto como qualquer outro que esteja armazenado em disco, ou seja, o processamento feito pelo AviSynth é transparente para a aplicação que está utilizando seus *scripts*.

A utilização do AviSynth foi realizada de diferentes maneiras, conforme o formato dos vídeos de entrada. As funções utilizadas são listadas na tabela C.2, e na sequência são exibidos alguns exemplos de arquivos de configuração que foram utilizados e que fazem uso dessas funções.

Tabela C.2: Funções utilizadas no AviSynth.

Nome	**Função**
AviSource()	Abertura de arquivos AVI
RawSource()	Abertura de arquivos YUV
AssumeFPS()	Indicação do fps do vídeo de entrada
Trim()	Corte temporal (remoção de quadros)
LanczosResize()	Redimensionamento espacial
crop()	Corte de regiões
KernelDeint()	Conversão de entrelaçado para progressivo

Exemplo 1:

```
RawSource("src11_ref.yuv",720,576,"UYVY")
AssumeFPS(30)
KernelDeint(order=1)
crop(8,0,704,576)
```

Exemplo 2:

```
SetMemoryMax(100)
AviSource("RedKayak_8bit.avi",false)
AssumeFPS(30)
crop(300,0,880,720)
LanczosResize(704,576)
```

Exemplo 3:

```
SetMemoryMax(100)
RawSource("2_ParkJoy_1280x720.yuv",1280,720,"UYVY")
AssumeFPS(60)
crop(200,0,880,720)
LanczosResize(704,576)
ConvertFPS(30,zone=80)
```

C.1.2 VirtualDub

O VirtualDub é uma ferramenta utilizada para captura e processamento de vídeo que trabalha com arquivos AVI. Neste trabalho, o VirtualDub foi utilizado para execução dos arquivos de configuração do AviSynth e gravação dos resultados (os vídeos processados) em disco.

O processo de execução do VirtualDub foi feito através de sua versão de linha de comando (também possui uma versão com interface gráfica), com o uso de arquivos de configuração similares aos arquivos criados para o AviSynth, que especificam as operações que devem ser executadas. Os arquivos criados para execução do VirtualDub que processam cada um dos 3 exemplos dados para o uso do AviSynth são mostrados abaixo:

Exemplo 1:

```
VirtualDub.video.SetInputFormat(0);
VirtualDub.Open(U"src11_to_4CIF.avs");
VirtualDub.video.SetOutputFormat(15);
VirtualDub.video.SetMode(3);
VirtualDub.SaveAVI(U"src11_704x576_YUV420.avi");
```

Exemplo 2:

```
VirtualDub.video.SetInputFormat(0);
VirtualDub.Open(U"RedKayak_to_4CIF.avs");
VirtualDub.video.SetOutputFormat(15);
VirtualDub.video.SetMode(3);
VirtualDub.SaveAVI(U"RedKayak_704x576_YUV420.avi");
```

Exemplo 3:

```
VirtualDub.video.SetInputFormat(0);
VirtualDub.Open(U"2_ParkJoy_to_4CIF.avs");
VirtualDub.video.SetOutputFormat(15);
VirtualDub.video.SetMode(3);
VirtualDub.SaveAVI(U"2_ParkJoy_704x576_YUV420.avi");
```

O processamento executado pelos arquivos exemplificados acima é, inicialmente, a identificação do formato de entrada dos dados (o número 0 indica a detecção automática do formato) e a leitura dos arquivos `.avs`, ou seja, o *script* criado para o AviSynth. Posteriormente, é identificado o formato de saída (o número 15 indica o formato 4:2:0 YCbCr, YV12/I420) e o modo de processamento que deve ser utilizado pelo programa (o número 3 indica *full processing mode*). Por fim, é chamada a função que grava o arquivo processado em disco. A linha de comando para execução destes exemplos é bastante simples, como exibida abaixo, onde "exemplo.vds" é um arquivo que contém os comandos do VirtualDub (como qualquer um dos 3 exemplos acima).

Comando:

```
vdub /s exemplo.vds
```

C.1.3 FFmpeg

O FFmpeg é outro programa que executa diversas operações sobre áudio e vídeo. Ele foi utilizado neste trabalho apenas para a remoção dos cabeçalhos dos arquivos AVI criados após a execução do VirtualDub. A remoção dos cabeçalhos é feita de forma simples, onde o arquivo AVI é informado como entrada e o FFmpeg é instruído a salvá-lo no formato YUV 4:2:0 (YV12/I420), como utilizado pelas ferramentas do JSVM. A linha de comando abaixo exemplifica o uso do FFmpeg:

Comando:

```
ffmpeg -i src5_704x576.avi -pix_fmt yuv420p -vcodec rawvideo src5_704x576.yuv
```

C.2 Codificação escalável

Como já comentado na seção 3.3.4, o processo de codificação escalável foi todo realizado com o uso das ferramentas do JSVM. Entre as ferramentas disponibilizadas pelo JSVM, seis foram utilizadas: codificação, codificação iterativa, decodificação, extração, redimensionamento e verificação de taxa de codificação e PSNR. Além do uso dessas ferramentas, outras etapas também foram realizadas com o uso da aplicação *lyuv* (ver apêndice B.2): a remoção de quadros, a simulação da instabilidade e o cálculo da taxa de codificação através dos arquivos de log.

Abaixo, as etapas da codificação (e outras etapas menores também necessárias) serão descritas na ordem em que foram executadas: (I) redimensionamento espacial, (II)

inclusão de quadros adicionais, (III) codificação iterativa, (IV) codificação, (V) extração de camadas, (VI) decodificação, (VII) remoção dos quadros adicionais, (VIII) análise da taxa e PSNR e, por fim, (IX) simulação da instabilidade. Os pontos principais da descrição dessas etapas são a apresentação dos arquivos de configuração utilizados na codificação e as linhas de comando para execução de cada ferramenta. Todas essas ferramentas do JSVM trabalham com vídeos no formato YUV 4:2:0 (YV12/I420).

Os exemplos que serão exibidos são apenas para alguns arquivos de configuração ou comandos utilizados, mas diversos outros foram necessários na prática. Os demais arquivos podem ser encontrados em <http://www.inf.ufrgs.br/~lcdaronco/dissertacao>

(I) Redimensionamento espacial: O redimensionamento espacial desta etapa difere-se do redimensionamento feito durante o pré-processamento. Como comentado na seção 3.3.4, é necessário redimensionar os vídeos para as resoluções QCIF e CIF (primeira e segunda camadas) antes de codificá-los. Esse redimensionamento é feito com a ferramenta do JSVM chamada *DownConvertStatic*.

Esta ferramenta possibilita a redução da resolução espacial através de dois métodos: um método não-normativo (JVT-R006) e um método diádico (filtro de redução espacial do MPEG-4). No manual de utilização do JSVM (JSVM, 2008), é recomendado o uso do primeiro método, o não-normativo, que foi o método utilizado neste trabalho. Os comandos abaixo exemplificam a redução de vídeos com resolução 4CIF para CIF e QCIF, respectivamente.

Comandos:

```
DownConvertStatic 704 576 city_704x576.yuv 352 288 city_352x288.yuv 0
DownConvertStatic 704 576 ice_704x576.yuv 176 144 ice_352x288.yuv 0
```

(II) Inclusão de quadros adicionais: Esta etapa consiste na inclusão dos dois segundos adicionais (mais precisamente, 64 quadros) no início e no fim de cada vídeo, como descrito na seção 3.3.4. O processo foi feito com o uso a operação "Expand" do programa *lyuv*, que é descrita no apêndice B.2. Abaixo são exibidos 3 exemplos de comandos utilizados para execução desta operação:

Comandos:

```
lyuv -x 3_aspen_704x576.yuv 3_aspen_704x576-exp.yuv 64 704 576
lyuv -x 3_snowmnt_352x288.yuv 3_snowmnt_352x288-exp.yuv 64 352 288
lyuv -x 3_redkayak_176x144.yuv 3_redkayak_176x144-exp.yuv 64 176 144
```

(III) Codificação iterativa: A codificação iterativa é o processo de realização de diversas codificações de um mesmo vídeo, com objetivo de encontrar os parâmetros de quantização que possam ser utilizados para codificar este vídeo na taxa ou qualidade determinadas. No JSVM, o processo é feito com o uso da ferramenta *FixedQPEncoderStatic*.

Para o uso desta ferramenta, são especificados os arquivos de configuração normais utilizados na codificação (serão exibidos no item IV) e outro arquivo adicional contendo, entre outros, os parâmetros de quantização iniciais. A partir desses parâmetros iniciais, a ferramenta utiliza o codificador diversas vezes, aumentando ou reduzindo o nível de quantização a cada iteração, com intuito de atingir a taxa (*bitrate*) ou qualidade (PSNR) determinados. No caso da codificação em mais de uma camada, o processo é realizado uma vez para cada camada: inicialmente são encontrados os parâmetros apenas para a primeira camada, depois são encontrados os parâmetros para a segunda camada, e assim por diante.

No arquivo de configuração do *FixedQPEncoderStatic* também é especificado o erro máximo (positivo e negativo) que pode ser tolerado no ajuste à taxa ou qualidade determinadas. Abaixo é exibido um arquivo de configuração do *FixedQPEncoderStatic*, que foi utilizado para codificação do vídeo "aspen" com a configuração de codificação Espacial:

```
---------- GENERAL ----------
L_aspen_S                     # Label
H264AVCEncoderLibTestStatic   # Encoder binary
PSNRStatic                    # PSNR binary
aspen_s_main.cfg              # Configuration file
str/aspen_s.264               # Results bitstream file
mot                           # Motion information folder
548                           # Number of frames to be encoded
16                            # GOP Size
32                            # Intra period (-1 for only 1 I pic)
30.0                          # Frames per second
3                             # Number of layers
0                             # Constrained intra for base layer
10                            # Number of Iterations
1                             # Mode (0:Rate, 1:PSNR)


---------- LAYER 0 ----------
176                       # Input width
144                       # Input height
3_aspen_176x144-exp.yuv   # Input file
rec/aspen_s_layer0.yuv    # Reconstructed file
38.00                     # Bit rate [kbit/s]
0.50                      # Maximum negative mismatch [%]
0.50                      # Maximum positive mismatch [%]
29.00                     # StartBaseQpResidual
29.00                     # StartQpModeDecision
1                         # Entropy (0:CAVLC, 1:CABAC)
0                         # Inter-layer prediction (0:no, 1:always, 2:MB adaptive)
-1                        # Base layer ID

---------- LAYER 1 ----------
352                       # Input width
288                       # Input height
3_aspen_352x288-exp.yuv   # Input file
rec/aspen_s_layer1.yuv    # Reconstructed file
38.00                     # Bit rate [kbit/s]
0.50                      # Maximum negative mismatch [%]
0.50                      # Maximum positive mismatch [%]
27.50                     # StartBaseQpResidual
27.50                     # StartQpModeDecision
1                         # Entropy (0:CAVLC, 1:CABAC)
2                         # Inter-layer prediction (0:no, 1:always, 2:MB adaptive)
0                         # Base layer ID

---------- LAYER 2 ----------
704                       # Input width
576                       # Input height
3_aspen_704x576-exp.yuv   # Input file
rec/aspen_s_layer2.yuv    # Reconstructed file
38.00                     # Bit rate [kbit/s]
0.50                      # Maximum negative mismatch [%]
0.50                      # Maximum positive mismatch [%]
28.00                     # StartBaseQpResidual
28.00                     # StartQpModeDecision
1                         # Entropy (0:CAVLC, 1:CABAC)
2                         # Inter-layer prediction (0:no, 1:always, 2:MB adaptive)
1                         # Base layer ID
```

Ao final da execução, a ferramenta mostra a taxa e PSNR obtidos para cada camada, o erro desses valores em relação ao valor alvo e o parâmetro de quantização utilizado para alcançar estes valores, como no exemplo abaixo:

```
L0:   QP = 31.000000    MQP = 31.000000    RATE =  111.8291 PSNR =   37.8845 [ -0.304 ] (3 iterations)
L1:   QP = 30.000000    MQP = 30.000000    RATE =  404.4591 PSNR =   38.1524 [  0.401 ] (2 iterations)
L2:   QP = 31.000000    MQP = 31.000000    RATE = 1082.3015 PSNR =   38.0981 [  0.258 ] (3 iterations)
```

(IV) Codificação: Para codificação escalável, o JSVM disponibiliza uma ferramenta denominada *H264AVCEncoderLibTestStatic*, a mesma utilizada durante a codificação iterativa. Este codificador é capaz de codificar vídeos em H.264 escalável ou não escalável.

Os diversos parâmetros necessários para codificação podem ser informados por linha de comando na chamada da ferramenta ou através de arquivos de configuração. Neste trabalho, utilizamos diversos arquivos de configuração, que serão exemplificados abaixo. É especificado um arquivo de configuração para cada camada de vídeo e mais um arquivo principal, que contém as configurações que se aplicam a todas as camadas. Os exemplos abaixo mostram os parâmetros que foram utilizados para codificação do SRC "aspen" com a configuração de codificação Espacial.

Arquivo de configuração principal:

```
#====================================================
# GENERAL
#====================================================
AVCMode               0                 # 0:Scalable, 1:AVC
OutputFile            str/aspen_q.264   # Bitstream file
FrameRate             30.0              # Maximum frame rate [Hz]
FramesToBeEncoded     548               # Number of frames (at input frame rate)


#====================================================
# CODING STRUCTURE
#====================================================
GOPSize               16                # GOP Size (at maximum frame rate)
IntraPeriod           32                # Multiple of GOPSize. Intra Period
BaseLayerMode         2                 # Base layer mode
                                        #  0: AVC compatible w larger DPB
                                        #  1: AVC compatible (no extraction of temporal layers)
                                        #  2: AVC w subsequence SEI to support temporal scal.

#====================================================
# LAYER DEFINITION
#====================================================
NumLayers             3                 # Number of layers. Spatial or CGS layers (1..8)
LayerCfg              aspen_q_layer0.cfg  # Layer configuration file
LayerCfg              aspen_q_layer1.cfg  # Layer configuration file
LayerCfg              aspen_q_layer2.cfg  # Layer configuration file


#====================================================
# MOTION SEARCH
#====================================================
SearchMode            4                 # Search mode (0:BlockSearch, 4:FastSearch)
SearchFuncFullPel     3                 # Distortion measure used on integer-sample positions
                                        # (0:SAD-Y, 1:SSE-Y, 2:HADAMARD-Y, 3:SAD-YUV)
SearchFuncSubPel      0                 # Distortion measure used on sub-sample positions
                                        # (0:SAD-Y, 1:SSE-Y, 2:HADAMARD-Y)
SearchRange           96                # Maximum search range for motion
```

Configuração camada 1:

```
#====================================================
# INPUT/OUTPUT
#====================================================
SourceWidth           176               # Input frame width - shall be multiple of 16
SourceHeight          144               # Input frame height - shall be multiple of 16
FrameRateIn           30                # Input  frame rate [Hz]
FrameRateOut          30                # Output frame rate [Hz]
InputFile             aspen_176x144.yuv  # Input file
ReconFile             rec/aspen_s_l0.yuv # Reconstructed file
SymbolMode            1                 # Entropy coding mode (0:CALVC, 1:CABAC)


#====================================================
# CODING
#====================================================
QP                    29.0              # Base QP for the layer


#====================================================
# INTER-LAYER PRED
#====================================================
InterLayerPred        0                 # Inter-layer pred (0:no, 1:always, 2:yes, MB adaptive way)
                                        # Shall ALWAYS be 0 for base layer. Best efficiency = 2
```

Configuração camada 2:

```
#==========================================================
# INPUT/OUTPUT
#==========================================================
SourceWidth           352               # Input frame width - shall be multiple of 16
SourceHeight          288               # Input frame height - shall be multiple of 16
FrameRateIn           30                # Input  frame rate [Hz]
FrameRateOut          30                # Output frame rate [Hz]
InputFile             aspen_352x288.yuv  # Input file
ReconFile             rec/aspen_s_l1.yuv # Reconstructed file
SymbolMode            1                 # Entropy coding mode (0:CALVC, 1:CABAC)


#==========================================================
# CODING
#==========================================================
QP                    27.5              # Base QP for the layer


#==========================================================
# INTER-LAYER PRED
#==========================================================
InterLayerPred        2                 # Inter-layer pred (0:no, 1:always, 2:yes, MB adaptive way)
                                        # Shall ALWAYS be 0 for base layer. Best efficiency = 2
```

Configuração camada 3:

```
#==========================================================
# INPUT/OUTPUT
#==========================================================
SourceWidth           704               # Input frame width - shall be multiple of 16
SourceHeight          576               # Input frame height - shall be multiple of 16
FrameRateIn           30                # Input  frame rate [Hz]
FrameRateOut          30                # Output frame rate [Hz]
InputFile             aspen_704x576.yuv  # Input file
ReconFile             rec/aspen_s_l2.yuv # Reconstructed file
SymbolMode            1                 # Entropy coding mode (0:CALVC, 1:CABAC)


#==========================================================
# CODING
#==========================================================
QP                    28.0              # Base QP for the layer


#==========================================================
# INTER-LAYER PRED
#==========================================================
InterLayerPred        2                 # Inter-layer pred (0:no, 1:always, 2:yes, MB adaptive way)
                                        # Shall ALWAYS be 0 for base layer. Best efficiency = 2
```

Como pode ser visto nos exemplos acima, no arquivo de configuração principal são especificados parâmetros gerais da codificação, como o tamanho do GOP, o modo de codificação da camada base, as configurações do processo de estimativa de movimento e são indicados os arquivos de configuração de cada camada. Nos arquivos de configuração de cada camada são especificados os parâmetros que podem ser específicos para cada camada, como a resolução espacial, o número de quadros por segundo, a forma de codificação entrópica, o parâmetro de quantização (QP), entre outros.

Estes arquivos apenas exemplificam o uso do codificador, mas a lista de parâmetros que podem ser utilizados é muito mais extensa. Os parâmetros utilizados também diferem-se para as outras configurações de codificação (Temporal e Qualidade). Não é o objetivo desta seção mostrar todos os parâmetros e descrevê-los, mas sim deixar registrado um exemplo de como foram utilizados. Maiores informações podem ser encontradas no manual de utilização do JSVM (JSVM, 2008) e o restante dos arquivos utilizados está disponível no endereço comentado no início desta seção.

A execução do codificador após a construção dos arquivos de configuração é bastante simples, basta executá-lo por linha de comando informando o arquivo de configuração como parâmetro. Abaixo é exibido um exemplo de como é este comando para os arquivos do vídeo “aspen” mostrados anteriormente.

Comando para execução do codificador:

```
H264AVCEncoderLibTestStatic -pf aspen_s_main.cfg
```

Como já comentado na seção 3.3.4, é interessante observar o longo tempo de duração do processo de codificação. Em média, para um vídeo de 14 segundos, o tempo de codificação foi: 1 hora para a codificação Temporal, 4 horas para a codificação Espacial e 10 horas para a codificação Qualidade. Este é o tempo médio de duração, que varia conforme o SRC que está sendo codificado. O processamento foi feito em computadores com configurações variadas (mas equivalentes): algumas máquinas com processadores Pentium 4 3.0 GHz e 1GB de memória e outras com configurações similares.

(V) Extração de camadas: Após a codificação, é gerado um arquivo que contém o vídeo codificado com todas as suas camadas. Para separação dos vídeos de cada camada, o JSVM dispõe da ferramenta *BitStreamExtractorStatic*.

A extração das camadas é feita com os parâmetros `-l`, `-t` e `-q`, que indicam a camada espacial, temporal e de qualidade que devem ser extraídas, respectivamente. No caso da escalabilidade de qualidade por CGS (utilizada neste trabalho), as camadas são indicadas pelo parâmetro `-l` e não pelo parâmetro `-q`, pois, como comentado na seção 2.2.5, a escalabilidade de qualidade com CGS é considerada um caso especial da escalabilidade espacial.

O processo de extração é bastante simples, basta executar a ferramenta *BitStreamExtractorStatic* para cada camada que deve ser extraída informando os parâmetros corretos. Abaixo são exibidos os comandos que foram utilizados para extração das 3 camadas de vídeo para cada uma das configurações de codificação: Temporal, Espacial e Qualidade.

Comandos configuração Temporal[1]:

```
BitStreamExtractorStatic str/aspen_t.264 str/aspen_t_l0.264 -l 0 -t 1
BitStreamExtractorStatic str/aspen_t.264 str/aspen_t_l2.264 -l 0 -t 2
BitStreamExtractorStatic str/aspen_t.264 str/aspen_t_l3.264 -l 0 -t 4
```

Comandos configuração Espacial:

```
BitStreamExtractorStatic str/aspen_s.264 str/aspen_s_l0.264 -l 0 -t 4
BitStreamExtractorStatic str/aspen_s.264 str/aspen_s_l2.264 -l 1 -t 4
BitStreamExtractorStatic str/aspen_s.264 str/aspen_s_l3.264 -l 2 -t 4
```

Comandos configuração Qualidade:

```
BitStreamExtractorStatic str/aspen_q.264 str/aspen_q_l0.264 -l 0 -t 4
BitStreamExtractorStatic str/aspen_q.264 str/aspen_q_l2.264 -l 1 -t 4
BitStreamExtractorStatic str/aspen_q.264 str/aspen_q_l3.264 -l 2 -t 4
```

(VI) Decodificação: O processo de decodificação é bastante simples, assim como a extração das camadas. Como cada camada de vídeo já foi extraída para um arquivo (*bitstream*), basta acionar o decodificador *H264AVCDecoderLibTestStatic* para cada um desses arquivos. Abaixo são exibidas as linhas de comando para decodificar as 3 camadas da configuração Qualidade do vídeo "aspen".

Comandos:

```
H264AVCDecoderLibTestStatic str/aspen_q_l0.264 dec/aspen_q_l0-exp.yuv
H264AVCDecoderLibTestStatic str/aspen_q_l1.264 dec/aspen_q_l1-exp.yuv
H264AVCDecoderLibTestStatic str/aspen_q_l2.264 dec/aspen_q_l2-exp.yuv
```

[1]Note que os valores para o parâmetro `-t` são 1, 2 e 4. O valor 3 indicaria a extração de uma camada com 15 fps, que não foi utilizada (ver seção 3.2.1).

(VII) Remoção de quadros adicionais: Como foi feita a inclusão de quadros adicionais antes da codificação, após a decodificação de cada camada os vídeos ainda conterão esses quadros adicionais. Assim como a inclusão dos quadros, a remoção é feita com o uso do aplicativo *lyuv*, mas agora com a operação "Cut", já descrita no apêndice B.2.

Abaixo são exibidos exemplos de comandos utilizados para remoção dos quadros das camadas do vídeo "aspen" nas configurações Espacial e Temporal. É importante observar que, para a configuração Temporal, são extraídos menos quadros do que nas outras configurações, o que ocorre devido à redução temporal que é realizada nas camadas inferiores durante a codificação.

Comandos configuração Temporal:

```
lyuv -t dec/aspen_t_l0-exp.yuv dec/aspen_t_l0.yuv 8  704 576
lyuv -t dec/aspen_t_l1-exp.yuv dec/aspen_t_l1.yuv 16 704 576
lyuv -t dec/aspen_t_l2-exp.yuv dec/aspen_t_l2.yuv 64 704 576
```

Comandos configuração Espacial:

```
lyuv -t dec/aspen_s_l0-exp.yuv dec/aspen_s_l0.yuv 64 176 144
lyuv -t dec/aspen_s_l0-exp.yuv dec/aspen_s_l0.yuv 64 352 288
lyuv -t dec/aspen_s_l0-exp.yuv dec/aspen_s_l0.yuv 64 704 576
```

(VIII) Análise da taxa de codificação e PSNR: Os valores da taxa e PSNR dos vídeos codificados já são exibidos durante a codificação e durante a extração dos dados, portanto esta é uma etapa que normalmente não precisa ser adicionada ao processo. Porém, como os vídeos codificados neste trabalho incluem quadros adicionais que são removidos após a decodificação, os valores informados durante a codificação e extração são os valores para o vídeo expandido, e não para o vídeo que será realmente utilizado nas avaliações. Portanto, após a remoção dos quadros adicionais, todos os vídeos são analisados novamente utilizando a ferramenta do JSVM chamada *PSNRStatic*.

A *PSNRStatic* é uma ferramenta utilizada para cálculo do PSNR entre dois vídeos e que também pode ser utilizada para verificação da taxa de codificação de um vídeo. Para o cálculo do PSNR, são informados dois vídeos não compactados (formato YUV), sendo um deles o vídeo originalmente utilizado na codificação e o outro um dos vídeos gerados após a extração e decodificação (o vídeo de uma das camadas). Para cálculo da taxa, também deve ser informado o arquivo de vídeo codificado, ou seja, a *bitstream*. Abaixo são exibidos alguns exemplos do uso da ferramenta para cálculo do PSNR de vídeos codificados com cada uma das três configurações de codificação.

Comandos configuração Temporal:

```
PSNRStatic 704 576 aspen_704x576.yuv aspen_t_l0.yuv 3
PSNRStatic 704 576 aspen_704x576.yuv aspen_t_l1.yuv 2
PSNRStatic 704 576 aspen_704x576.yuv aspen_t_l2.yuv 0
```

Comandos configuração Espacial:

```
PSNRStatic 176 144 aspen_176x144.yuv aspen_s_l0.yuv
PSNRStatic 352 288 aspen_352x288.yuv aspen_s_l1.yuv
PSNRStatic 704 576 aspen_704x576.yuv aspen_s_l2.yuv
```

Comandos configuração Qualidade:

```
PSNRStatic 704 576 aspen_704x576.yuv aspen_q_l0.yuv
PSNRStatic 704 576 aspen_704x576.yuv aspen_q_l1.yuv
PSNRStatic 704 576 aspen_704x576.yuv aspen_q_l2.yuv
```

Nos exemplos, os parâmetros indicam, respectivamente: largura do vídeo, altura do vídeo, vídeo original, vídeo degradado e, opcionalmente, nível de decomposição temporal. No caso da configuração Espacial, o vídeo original ao qual cada camada deve ser comparada já é o vídeo após a redução feita na etapa (I). Para a configuração Temporal, a diferença é que deve ser informado ao aplicativo qual o nível de decomposição temporal, o último parâmetro nos exemplos acima. A análise de PSNR na configuração Qualidade não tem nenhuma grande complicação, basta utilizar comandos similares aos utilizados na configuração Espacial, mas sempre com a resolução espacial 704x576.

Como a taxa de codificação é calculado com base no arquivo codificado e o arquivo codificado ainda contém os quadros adicionais, esse valor de taxa não é referente aos vídeos que seriam utilizados nas avaliações. Portanto, a ferramenta *PSNRStatic* não pôde ser utilizada para este cálculo.

Para cálculo da taxa de forma precisa, foi utilizada a operação "Get rate from log" do aplicativo *lyuv*, descrita no apêndice B.2. Com ela, a taxa é calculada a partir dos resultados da codificação (encontrados nos logs), mas removendo os quadros adicionais. Abaixo são exibidos exemplos para as três configurações de codificação.

Comandos configuração Temporal:

```
lyuv -l logs\aspen_t_enc.log logs\aspen_t_PSNR-rate.log 64 1 0 0 30
lyuv -l logs\aspen_t_enc.log logs\aspen_t_PSNR-rate.log 64 2 0 0 30
lyuv -l logs\aspen_t_enc.log logs\aspen_t_PSNR-rate.log 64 4 0 0 30
```

Comandos configuração Espacial:

```
lyuv -l logs\aspen_q_enc.log logs\aspen_q_PSNR-rate.log 64 4 0 0 30
lyuv -l logs\aspen_q_enc.log logs\aspen_q_PSNR-rate.log 64 4 1 0 30
lyuv -l logs\aspen_q_enc.log logs\aspen_q_PSNR-rate.log 64 4 2 0 30
```

Comandos configuração Qualidade:

```
lyuv -l logs\aspen_q_enc.log logs\aspen_q_PSNR-rate.log 64 4 0 0 30
lyuv -l logs\aspen_q_enc.log logs\aspen_q_PSNR-rate.log 64 4 1 0 30
lyuv -l logs\aspen_q_enc.log logs\aspen_q_PSNR-rate.log 64 4 2 0 30
```

Nos exemplos acima, a ordem dos parâmetros é: arquivo de log gerado na codificação, arquivo de saída (onde será adicionado o valor da taxa de codificação), número de quadros que devem ser ignorados no início e fim do vídeo, camada temporal, camada espacial, camada de qualidade e número de quadros por segundo (fps). Para as configurações Espacial e Qualidade, a camada temporal é sempre a 4ª, ou seja, a última (30 fps). Já a camada espacial deve variar para cálculo da taxa de cada camada (como já comentado, a configuração Qualidade utiliza MGS e, portanto, as camadas de qualidade podem ser consideradas como camadas espaciais). E, na configuração Temporal, basta variar os parâmetros que indicam qual camada temporal será calculada.

(IX) Simulação da instabilidade: A simulação da instabilidade, já comentada nas seções 3.2.2 e 3.3.5, foi feita com o uso da operação "Encode" do aplicativo *lyuv*. Para isso, inicialmente os arquivos de configuração criados para uso do *lyuv* foram organizados seguindo a nomenclatura exibida na tabela C.3.

Os arquivos de configuração são utilizados para informar ao *lyuv* como deve ser feita a integração das camadas no vídeo final, ou seja, eles contêm a descrição do padrão de instabilidade usado e quais os vídeos que formarão o vídeo final. Abaixo serão exibidos exemplos para a codificação do HRC 3 de cada configuração de

Tabela C.3: Relação dos arquivos de configuração com os HRCs.

Nome	HRC	Nome	HRC	Nome	HRC
t-hrc1	T | p0 | 1	s-hrc1	E | p0 | 1	q-hrc1	Q | p0 | 1
t-hrc2	T | p0 | 2	s-hrc2	E | p0 | 2	q-hrc2	Q | p0 | 2
t-hrc3	T | p4 | 1-2	s-hrc3	E | p4 | 1-2	q-hrc3	Q | p4 | 1-2
t-hrc4	T | p4 | 2-3	s-hrc4	E | p4 | 2-3	q-hrc4	Q | p4 | 2-3
t-hrc5	T | p8 | 1-2	s-hrc5	E | p8 | 1-2	q-hrc5	Q | p8 | 1-2
t-hrc6	T | p8 | 2-3	s-hrc6	E | p8 | 2-3	q-hrc6	Q | p8 | 2-3

codificação, ou seja, os HRCs aqui chamados de “t-hrc3”, “s-hrc3” e “q-hrc3”, cuja relação com a nomenclatura utilizada para os HRCs no restante deste trabalho é exibida na tabela C.3. Estes HRCs utilizam o padrão de instabilidade $p4$ aplicado entre as camadas 1 e 2.

Arquivo de configuração para o HRC “t-hrc3” do SRC “aspen”:

```
aspen_t_l0.yuv 704 576 3.75 11
aspen_t_l1.yuv 704 576 7.5 16
aspen_t_l0.yuv 704 576 3.75 11
aspen_t_l1.yuv 704 576 7.5 15
aspen_t_l0.yuv 704 576 3.75 15
```

Arquivo de configuração para o HRC “s-hrc3” do SRC “aspen”:

```
aspen_s_l0.yuv 176 144 30 90
aspen_s_l1.yuv 352 288 30 60
aspen_s_l0.yuv 176 144 30 90
aspen_s_l1.yuv 352 288 30 60
aspen_s_l0.yuv 176 144 30 120
```

Arquivo de configuração para o HRC “q-hrc3” do SRC “aspen”:

```
aspen_q_l0.yuv 704 576 30 90
aspen_q_l1.yuv 704 576 30 60
aspen_q_l0.yuv 704 576 30 90
aspen_q_l1.yuv 704 576 30 60
aspen_q_l0.yuv 704 576 30 120
```

Como já descrito na seção B.2, cada linha dos arquivos de configuração representa um vídeo que será incluído no arquivo de saída. Junto ao nome do vídeo, é informada a resolução espacial, o número de quadros por segundo e o número de quadros que devem ser utilizados para este vídeo. Nos exemplos dados acima, alterna-se entre os vídeos das camadas 1 e 2 de forma a seguir o padrão de instabilidade $p4$.

Além da variação entre as camadas, os vídeos também foram normalizados durante a simulação da instabilidade, ou seja, foram todos convertidos para a resolução 4CIF utilizando 30 fps. O aplicativo *lyuv* permite que esta normalização seja realizada durante o processo de união dos vídeos, basta informar alguns parâmetros adicionais quando o aplicativo for executado.

A simulação da instabilidade para o padrão $p8$ e para os padrões $p4$ e $p8$ entre as camadas 2-3 é bastante semelhante aos exemplos dados. A maior diferença está nos HRCs que utilizam o padrão $p0$, que, por ser o padrão estável, não precisa desta etapa de simulação de instabilidade. No padrão $p0$, os vídeos das camadas 1 (no “hrc1”) ou 2 (“hrc2”) são apenas normalizados com a operação “Convert” do

aplicativo *lyuv* e copiados para um novo arquivo, gerando assim uma PVS que será utilizada nas avaliações.

Abaixo é exibido um arquivo *batch* (para processamento em lote na plataforma Windows) que foi utilizado para aplicar todos os padrões de instabilidade em um SRC, gerando assim as 18 PVSs que são criadas para cada SRC. Este arquivo mostra como deve ser executado o aplicativo *lyuv* e também como foram aplicados os HRCs que utilizam o padrão estável $p0$, já que estes não necessitam da simulação de instabilidade.

Comandos para simulação da instabilidade:

```
copy /B "%1_q_l0.yuv" /B "HRC/%1_q-hrc1.yuv"
copy /B "%1_q_l1.yuv" /B "HRC/%1_q-hrc2.yuv"
lyuv -e %1_q-hrc3.cfg HRC/%1_q-hrc3.yuv -n 704 576 30
lyuv -e %1_q-hrc4.cfg HRC/%1_q-hrc4.yuv -n 704 576 30
lyuv -e %1_q-hrc5.cfg HRC/%1_q-hrc5.yuv -n 704 576 30
lyuv -e %1_q-hrc6.cfg HRC/%1_q-hrc6.yuv -n 704 576 30

lyuv -c %1_t_l0.yuv 704 576 3.75 HRC/%1_t-hrc1.yuv 704 576 30
lyuv -c %1_t_l1.yuv 704 576 7.5 HRC/%1_t-hrc2.yuv 704 576 30
lyuv -e %1_t-hrc3.cfg HRC/%1_t-hrc3.yuv -n 704 576 30
lyuv -e %1_t-hrc4.cfg HRC/%1_t-hrc4.yuv -n 704 576 30
lyuv -e %1_t-hrc5.cfg HRC/%1_t-hrc5.yuv -n 704 576 30
lyuv -e %1_t-hrc6.cfg HRC/%1_t-hrc6.yuv -n 704 576 30

lyuv -c %1_s_l0.yuv 176 144 30 HRC/%1_s-hrc1.yuv 704 576 30
lyuv -c %1_s_l1.yuv 352 288 30 HRC/%1_s-hrc2.yuv 704 576 30
lyuv -e %1_s-hrc3.cfg HRC/%1_s-hrc3.yuv -n 704 576 30
lyuv -e %1_s-hrc4.cfg HRC/%1_s-hrc4.yuv -n 704 576 30
lyuv -e %1_s-hrc5.cfg HRC/%1_s-hrc5.yuv -n 704 576 30
lyuv -e %1_s-hrc6.cfg HRC/%1_s-hrc6.yuv -n 704 576 30
```

No exemplo acima, "%1" é um parâmetro que é passado ao arquivo *batch* com o nome do SRC que deve ser processado. O arquivo *batch* é então executado uma vez para cada um dos 11 SRCs utilizados na avaliação e treinamento para geração de todas as PVSs definidas.

APÊNDICE D DADOS DAS AVALIAÇÕES SUBJETIVAS

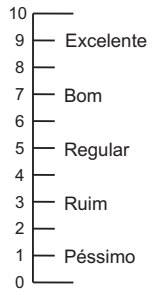
D.1 Instruções impressas

O texto abaixo consiste no documento que foi entregue a todos os avaliadores logo antes do início das avaliações, contendo as principais instruções que deviam estar claras a todos eles.

Instruções aos avaliadores

O processo de avaliação consiste na visualização de diversos vídeos de curta duração e atribuição de uma nota a cada um. A atribuição da nota é feita logo após a visualização de cada vídeo. Estes vídeos foram processados de diferentes maneiras, portanto eles podem parecer diferentes (ou não) para você.

A nota atribuída corresponde ao nível de qualidade que você interpretou após ver o vídeo. Não existe uma resposta certa, baseie-se em seu gosto e julgamento. Você pode escolher uma entre onze notas, que estão divididas em 5 grupos:



A nota 0 corresponde à pior qualidade possível, e representa um vídeo dificilmente distinguível e/ou muito desagradável/irritante de se assistir. A nota 10 corresponde à melhor qualidade, representando vídeos onde não se percebe nenhuma degradação.

Após a exibição do vídeo, pense por alguns segundos e então atribua a nota. Tente não levar mais que 10 segundos neste processo. Assim que você votar e que o próximo vídeo estiver carregado, você poderá prosseguir com a avaliação. Você também pode repetir a exibição do vídeo mais uma vez (e apenas uma), mas só repita se for realmente necessário.

Tente prestar atenção nos detalhes dos vídeos e não em seu conteúdo, percebendo as variações de qualidade que poderão ocorrer durante a exibição. Observe os vídeos e atribua uma nota correspondente à qualidade geral ao longo de toda sua exibição. Esta nota também pode ser interpretada como o seu nível de satisfação com o serviço que lhe foi disponibilizado.

Serão exibidos 152 vídeos com 14 segundos de duração cada. Após a exibição de 76 vídeos você poderá fazer uma pausa antes de prosseguir para a próxima etapa (a aplicação

irá lhe avisar no momento certo). A estimativa de duração total da avaliação é de 1 hora.

Antes do processo de avaliação haverá uma breve sessão de treinamento. Nesta sessão você se familiarizará com a aplicação e com a qualidade dos vídeos que estará avaliando. Os vídeos selecionados para treinamento tentam abranger todos os processamentos feitos sobre os vídeos, incluindo vídeos de alta e de baixa qualidade. Tente utilizar este conhecimento para auxiliar nas decisões durante o início da avaliação.

Por fim, os vídeos podem parecer diferentes dependendo da distância de observação, portanto é necessário que você permaneça na posição indicada, movendo-se o mínimo necessário para ficar mais confortável.

Obrigado pela participação!

D.2 Questionário

As questões abaixo formam o questionário que foi entregue para todos os avaliadores após a execução das avaliações. As perguntas foram elaboradas de maneira bastante informal e o preenchimento do questionário era opcional e feito de forma anônima. Os resultados são exibidos no apêndice D.4.

- Duração dos vídeos individualmente:
    1. Muito longos, poderiam ser menores
    2.
    3. Boa pra perceber os detalhes
    4.
    5. Muito curtos, difícil perceber tudo tão rápido
- Duração geral da avaliação:
    1. Muito longa, muito cansativa
    2. Longa e cansativa, mas não tanto
    3. Boa
    4. Não cansativa, e poderia durar um pouco mais
    5. Poderia durar mais tempo sem problemas
- A maioria dos vídeos apresentava qualidade muito:
    1. Alta
    2. Bastante variação, mas a maioria alta
    3. Variada
    4. Bastante variação, mas a maioria baixa
    5. Baixa
- Dificuldade de se perceber as alterações ao longo dos vídeos (em média):
    1. Muito difícil
    2.

3. Alguns fáceis, alguns difíceis
4.
5. Fácil de perceber

- Conteúdo dos vídeos (em relação à concentração):

1. Ruim: atraiam a atenção para o conteúdo, difícil prestar atenção nos detalhes
2.
3. Indiferente
4.
5. Bom: não chamavam atenção, era fácil se focar nos detalhes

D.3 Faixa etária e gênero dos avaliadores

Na tabela D.1 é exibido o gênero e a faixa etária de cada um dos 22 avaliadores que participaram das avaliações deste trabalho.

Tabela D.1: Faixa etária e gênero de todos avaliadores.

	Sexo		**Faixa etária**					
Avaliador	**M**	**F**	**18-20**	**21-23**	**24-26**	**27-35**	**36-45**	**46-60**
1	x		x					
2	x		x					
3		x				x		
4	x		x					
5	x		x					
6	x						x	
7		x			x			
8	x					x		
9		x			x			
10	x		x					
11	x							x
12	x			x				
13	x			x				
14	x					x		
15	x					x		
16	x					x		
17	x					x		
18	x				x			
19	x			x				
20	x			x				
21	x			x				
22	x			x				
Total:	19	3	5	6	3	6	1	1
Total (%):	86,36	13,64	22,73	27,27	13,64	27,27	4,54	4,54

D.4 Resultados dos questionário

Nesta seção são exibidas as respostas do questionário apresentado no apêndice D.2. Entre os 22 avaliadores que participaram das avaliações, 18 preencheram o questionário, e as respostas para cada uma das questões são exibidas nas tabelas D.2, D.3, D.4, D.5 e D.6. Os identificadores dados aos avaliadores nesta seção *não* correspondem aos mesmos identificadores dos dados apresentados no apêndice D.3. Porém, os identificadores são únicos entre as tabelas desta seção, ou seja, o avaliador de número n em uma tabela é o mesmo avaliador de número n nas outras tabelas. A coluna "-" nas opções de resposta é marcada caso um avaliador não tenha respondido a questão.

Tabela D.2: Questão 1: Duração dos vídeos individualmente.

	Opções					
Avaliador	**1**	**2**	**3**	**4**	**5**	**-**
1						x
2			x			
3	x					
4			x			
5			x			
6			x			
7			x			
8			x			
9			x			
10			x			
11			x			
12			x			
13			x			
14			x			
15			x			
16			x			
17			x			
18			x			
Total:	1	0	16	0	0	1
Total (%):	5,56	0,00	88,89	0,00	0,00	5,56

D.5 Relatório dos votos de todo avaliadores

As tabelas D.7 e D.8 contém um relatório dos votos atribuídos por todos avaliadores para cada uma das PVSs avaliadas. Os votos dessas tabelas são brutos, ou seja, exatamente os valores atribuídos por cada avaliador antes do processo de normalização descrito na seção 4.1. Os SCRs são nomeados de acordo com os identificadores utilizados no restante do trabalho, que podem ser vistos na tabela 3.9. Os HRCs também são identificados como no restante do trabalho (ver seção 4.1), sendo que o HRC "ref" representa o vídeo de referência.

Tabela D.3: Questão 2: Duração geral da avaliação.

	Opções					
Avaliador	**1**	**2**	**3**	**4**	**5**	**-**
1		x				
2			x			
3		x				
4			x			
5			x			
6		x				
7		x				
8			x			
9		x				
10		x				
11		x				
12		x				
13		x				
14		x				
15		x				
16		x				
17			x			
18		x				
Total:	0	13	5	0	0	0
Total (%):	0,00	72,22	27,78	0,00	0,00	0,00

Tabela D.4: Questão 3: A maioria dos vídeos apresentava qualidade muito...

	Opções					
Avaliador	**1**	**2**	**3**	**4**	**5**	**-**
1				x		
2				x		
3				x		
4			x			
5				x		
6				x		
7				x		
8			x			
9			x			
10			x			
11			x			
12			x			
13			x			
14				x		
15				x		
16			x			
17				x		
18			x			
Total:	0	0	9	9	0	0
Total (%):	0,00	0,00	50,00	50,00	0,00	0,00

Tabela D.5: Questão 4: Dificuldade de se perceber as alterações ao longo dos vídeos (em média).

	Opções					
Avaliador	**1**	**2**	**3**	**4**	**5**	**-**
1			x			
2			x			
3					x	
4					x	
5				x		
6			x			
7			x			
8					x	
9					x	
10					x	
11			x			
12				x		
13			x			
14			x			
15					x	
16					x	
17					x	
18					x	
Total:	0	0	7	2	9	0
Total (%):	0,00	0,00	38,89	11,11	50,00	0,00

Tabela D.6: Questão 5: Conteúdo dos vídeos (em relação à concentração).

	Opções					
Avaliador	**1**	**2**	**3**	**4**	**5**	**-**
1			x			
2			x			
3					x	
4					x	
5				x		
6					x	
7			x			
8			x			
9					x	
10			x			
11			x			
12				x		
13			x			
14					x	
15					x	
16					x	
17			x			
18					x	
Total:	0	0	8	2	8	0
Total (%):	0,00	0,00	44,44	11,11	44,44	0,00

Tabela D.7: Valores brutos dos votos de todos os avaliadores para todas PVSs.

		Avaliadores																					
SRC	**HRC**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**
1	ref	8	10	9	10	10	10	10	10	7	10	6	9	9	9	10	9	8	4	9	9	10	10
1	Q | p0 | 1	2	0	2	4	4	3	1	3	4	3	4	2	5	4	4	1	1	3	1	2	4	1
1	Q | p0 | 2	5	0	6	5	4	4	4	3	8	6	4	2	5	5	4	5	3	4	5	1	5	5
1	Q | p4 | 1-2	2	0	6	4	5	5	2	3	3	2	6	1	3	2	3	3	0	2	2	3	3	4
1	Q | p4 | 2-3	4	2	6	6	7	6	3	3	7	4	5	2	6	5	4	3	2	5	4	4	4	5
1	Q | p8 | 1-2	3	0	3	3	6	5	2	2	3	5	4	2	3	2	3	2	1	5	2	2	2	4
1	Q | p8 | 2-3	3	0	6	6	5	7	4	3	7	5	7	3	6	5	6	3	2	5	5	3	4	5
1	E | p0 | 1	3	0	3	3	3	7	3	2	2	4	2	0	3	4	2	0	0	2	1	0	1	2
1	E | p0 | 2	3	0	5	5	6	8	7	4	7	5	8	2	6	5	5	3	3	4	3	5	4	5
1	E | p4 | 1-2	1	2	2	1	3	5	0	1	3	2	7	1	5	3	3	1	1	3	0	0	4	2
1	E | p4 | 2-3	4	2	5	6	5	8	7	3	6	5	7	3	6	4	5	3	4	6	5	4	6	6
1	E | p8 | 1-2	1	0	3	1	4	7	3	1	5	4	5	2	6	2	2	0	1	2	1	1	0	3
1	E | p8 | 2-3	5	1	5	4	7	7	3	4	6	5	6	4	6	5	5	3	4	4	6	7	3	6
1	T | p0 | 1	3	0	4	3	0	5	0	4	8	3	7	5	4	1	2	4	1	2	0	1	5	4
1	T | p0 | 2	4	4	3	5	5	4	4	5	8	4	9	5	4	3	2	3	2	4	4	4	6	5
1	T | p4 | 1-2	5	4	2	3	1	5	2	3	7	3	8	5	4	2	2	2	3	2	4	3	5	3
1	T | p4 | 2-3	5	3	3	4	3	5	1	5	5	4	6	4	5	3	3	3	2	5	4	4	6	6
1	T | p8 | 1-2	4	2	7	2	4	3	4	5	4	4	5	3	5	2	1	3	1	4	1	3	7	4
1	T | p8 | 2-3	3	3	4	2	1	5	3	5	6	4	8	4	6	3	3	5	2	7	3	5	4	5
2	ref	6	8	8	9	10	10	10	9	10	9	10	7	9	8	10	9	9	9	9	8	9	10
2	Q | p0 | 1	3	2	5	5	6	8	3	6	6	7	9	3	6	6	4	4	5	5	4	4	5	6
2	Q | p0 | 2	5	7	5	7	8	8	5	5	8	7	8	5	8	6	8	7	7	5	6	6	7	8
2	Q | p4 | 1-2	3	2	5	5	5	8	6	5	5	6	7	5	7	4	5	6	6	6	2	4	5	7
2	Q | p4 | 2-3	5	6	7	7	6	8	6	7	8	7	7	6	6	6	7	7	7	7	6	6	5	8
2	Q | p8 | 1-2	4	5	6	5	6	8	4	6	8	5	6	5	7	6	4	6	6	6	5	5	4	6
2	Q | p8 | 2-3	5	6	7	6	6	8	8	7	8	8	8	6	6	6	8	6	7	6	9	6	5	7
2	E | p0 | 1	3	1	2	4	6	7	3	4	4	4	6	2	3	5	2	1	5	4	1	3	2	4
2	E | p0 | 2	4	8	6	7	9	8	6	7	8	7	8	4	6	6	5	6	7	7	6	7	6	7
2	E | p4 | 1-2	3	0	3	4	4	7	3	5	1	3	5	3	4	4	3	4	3	5	1	6	1	6
2	E | p4 | 2-3	5	5	6	6	8	8	6	7	7	7	7	4	7	6	7	7	7	7	4	6	6	8
2	E | p8 | 1-2	3	1	3	3	5	6	4	3	4	4	6	2	4	5	3	3	2	4	2	1	1	6
2	E | p8 | 2-3	6	3	7	6	10	8	4	6	6	6	8	6	6	6	7	7	7	7	5	7	4	7
2	T | p0 | 1	2	0	2	1	2	5	0	4	3	4	7	2	3	1	1	5	1	3	0	1	5	3
2	T | p0 | 2	2	4	3	2	2	5	3	5	6	4	7	3	4	5	4	7	1	4	4	4	7	7
2	T | p4 | 1-2	3	3	2	1	0	3	0	4	2	3	6	2	6	2	1	4	2	4	1	5	5	4
2	T | p4 | 2-3	3	2	3	3	5	5	4	6	6	4	8	3	5	3	2	5	2	4	1	5	6	7
2	T | p8 | 1-2	3	3	0	1	1	3	2	2	3	4	5	3	3	1	1	5	3	3	2	3	7	4
2	T | p8 | 2-3	3	4	4	5	0	6	1	6	5	3	6	3	5	3	2	6	2	5	1	6	6	6
3	ref	7	10	8	10	10	10	9	9	10	8	9	7	7	9	9	8	9	9	10	9	10	9
3	Q | p0 | 1	3	5	5	7	7	8	6	6	5	5	8	4	7	6	4	6	6	6	6	5	2	5
3	Q | p0 | 2	5	9	8	8	9	9	8	7	10	8	9	6	7	6	8	7	8	7	8	7	7	7
3	Q | p4 | 1-2	3	5	5	5	6	7	6	6	5	6	8	4	7	5	5	5	6	6	5	6	5	5
3	Q | p4 | 2-3	4	9	8	7	7	8	9	8	8	7	9	5	6	5	7	7	7	5	8	7	6	8
3	Q | p8 | 1-2	3	7	4	6	7	7	5	6	5	6	7	3	7	5	4	4	5	4	7	5	4	6
3	Q | p8 | 2-3	4	6	6	8	9	8	8	7	9	8	9	6	8	6	5	6	10	7	8	8	8	8
3	E | p0 | 1	3	1	5	4	6	7	4	2	6	4	3	2	4	5	3	0	0	4	1	3	0	6
3	E | p0 | 2	3	3	8	8	8	8	8	7	8	7	10	6	7	6	7	6	7	7	4	6	6	8
3	E | p4 | 1-2	3	0	3	3	5	7	4	3	2	3	7	2	3	4	2	3	2	5	2	1	4	4
3	E | p4 | 2-3	4	4	8	7	8	8	6	7	9	6	9	6	6	5	7	6	8	6	5	5	6	7
3	E | p8 | 1-2	3	1	5	3	3	6	3	5	4	3	7	2	4	4	3	6	5	4	2	5	2	5
3	E | p8 | 2-3	5	8	9	7	8	9	9	7	8	7	9	7	8	5	8	7	7	5	5	5	7	8
3	T | p0 | 1	1	2	1	2	1	5	2	5	6	3	7	3	4	2	1	4	0	2	3	3	7	4
3	T | p0 | 2	3	4	6	4	2	5	7	5	8	5	9	4	4	2	1	6	2	3	4	5	6	6
3	T | p4 | 1-2	2	1	1	3	0	5	5	3	4	3	6	3	5	1	2	4	1	1	1	4	7	3
3	T | p4 | 2-3	3	4	4	1	6	5	0	4	5	4	8	4	5	2	2	6	2	5	2	5	5	4
3	T | p8 | 1-2	1	3	5	2	1	5	6	1	6	4	7	3	4	2	1	5	2	4	3	4	6	4
3	T | p8 | 2-3	4	5	4	4	1	5	2	4	5	4	7	4	4	4	3	6	3	6	3	5	6	5
4	ref	8	10	7	10	10	9	8	8	8	8	7	9	8	9	10	9	8	9	10	7	10	10
4	Q | p0 | 1	1	0	0	3	6	4	6	1	1	2	6	1	4	2	1	2	0	1	4	3	0	3
4	Q | p0 | 2	3	0	1	5	6	4	2	1	3	2	6	2	6	2	3	2	0	3	5	5	2	6
4	Q | p4 | 1-2	3	0	1	4	6	5	3	3	3	3	3	1	5	4	3	1	1	5	4	4	1	3
4	Q | p4 | 2-3	4	2	2	4	6	6	6	4	2	5	6	3	6	4	3	4	3	3	6	5	5	6
4	Q | p8 | 1-2	1	0	3	4	5	5	0	3	3	2	7	1	4	3	2	2	0	5	5	3	2	5
4	Q | p8 | 2-3	5	2	4	5	6	7	3	3	6	3	5	2	7	3	5	5	1	4	5	4	2	5
4	E | p0 | 1	3	0	3	4	6	8	4	4	5	5	7	3	3	5	2	2	2	5	3	4	3	4
4	E | p0 | 2	3	5	3	6	7	8	5	4	8	4	6	5	7	5	7	5	6	5	6	4	5	7
4	E | p4 | 1-2	3	2	3	5	6	7	4	1	5	5	6	4	5	4	3	3	2	5	4	4	5	4
4	E | p4 | 2-3	4	4	6	6	9	8	5	5	6	5	8	4	7	5	6	5	6	5	6	6	6	8
4	E | p8 | 1-2	3	5	5	4	4	6	2	4	6	4	6	3	3	4	4	3	2	6	2	1	3	5
4	E | p8 | 2-3	4	7	5	8	8	7	5	5	4	6	7	5	6	4	8	5	6	6	6	5	6	7
4	T | p0 | 1	2	0	0	0	1	4	1	3	4	3	5	2	3	0	1	4	0	0	1	3	6	0
4	T | p0 | 2	3	2	3	2	4	5	3	4	7	4	7	3	4	3	2	4	2	5	2	1	6	5
4	T | p4 | 1-2	5	0	2	2	0	5	4	5	5	2	4	3	2	1	2	2	0	3	5	2	6	0
4	T | p4 | 2-3	2	3	3	4	7	5	5	6	6	3	5	3	6	2	3	5	3	5	3	4	6	6
4	T | p8 | 1-2	1	0	3	1	3	4	5	3	1	2	5	2	4	0	2	4	3	4	2	1	6	3
4	T | p8 | 2-3	2	3	4	2	7	4	5	4	5	3	8	4	5	3	3	4	3	5	5	2	6	7

Tabela D.8: Valores brutos dos votos de todos os avaliadores para todas PVSs (continuação).

		Avaliadores																					
SRC	**HRC**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**
5	ref	7	10	9	10	10	10	7	9	9	9	9	7	8	8	10	9	9	9	6	8	9	9
5	Q | p0 | 1	4	8	7	5	8	9	8	6	8	7	9	6	7	6	6	7	7	6	4	7	5	7
5	Q | p0 | 2	5	8	7	8	10	9	8	7	9	8	8	7	8	7	8	8	5	6	9	8	8	8
5	Q | p4 | 1-2	5	6	8	8	7	8	7	7	9	7	9	6	7	6	7	7	8	8	8	8	6	8
5	Q | p4 | 2-3	5	6	8	9	9	9	9	8	10	8	10	5	8	7	8	8	8	5	9	8	7	8
5	Q | p8 | 1-2	6	6	7	8	9	9	7	6	8	8	10	6	8	6	7	8	8	7	5	8	7	7
5	Q | p8 | 2-3	6	9	8	8	7	8	8	9	9	8	10	5	8	6	8	7	9	7	10	8	8	9
5	E | p0 | 1	0	0	3	0	2	6	0	2	1	2	2	1	2	3	1	4	1	5	0	1	0	2
5	E | p0 | 2	3	4	5	6	7	8	6	4	7	6	8	5	6	6	7	7	4	6	8	7	6	6
5	E | p4 | 1-2	1	0	5	1	3	6	1	1	2	2	2	2	3	3	1	1	1	2	1	1	1	3
5	E | p4 | 2-3	3	5	4	5	6	7	4	3	5	7	8	4	6	5	4	6	6	6	5	3	6	7
5	E | p8 | 1-2	3	0	2	2	5	5	1	4	3	3	2	2	1	4	2	3	1	4	1	0	0	3
5	E | p8 | 2-3	5	3	5	8	6	7	4	4	7	5	10	3	7	6	6	6	6	5	4	6	6	7
5	T | p0 | 1	3	3	5	2	0	4	0	4	5	4	5	2	5	1	2	4	4	3	2	5	6	4
5	T | p0 | 2	3	3	4	4	2	5	8	4	6	4	6	3	4	3	2	7	4	3	4	5	6	4
5	T | p4 | 1-2	3	2	4	4	2	5	3	4	5	3	6	5	3	2	2	4	2	4	3	4	6	5
5	T | p4 | 2-3	5	5	4	3	4	5	4	3	6	4	7	3	5	3	3	6	2	4	4	4	7	5
5	T | p8 | 1-2	3	2	2	2	3	5	0	3	6	4	6	2	5	2	2	5	3	1	1	4	5	3
5	T | p8 | 2-3	5	6	4	5	1	5	2	4	6	4	8	3	4	3	3	6	3	6	4	6	5	5
6	ref	7	9	7	10	10	10	9	8	10	8	9	8	9	8	9	8	9	9	9	8	9	9
6	Q | p0 | 1	3	0	5	7	8	8	6	5	7	6	9	4	7	6	6	7	2	3	6	6	6	7
6	Q | p0 | 2	6	3	5	8	8	10	8	8	8	6	9	6	8	7	7	7	5	6	7	7	7	5
6	Q | p4 | 1-2	4	7	5	7	7	7	5	6	8	7	9	5	7	5	5	6	2	3	4	6	4	8
6	Q | p4 | 2-3	5	4	6	9	10	9	9	8	8	7	8	6	7	7	7	6	7	6	6	8	6	8
6	Q | p8 | 1-2	5	4	4	7	8	8	9	5	6	6	9	5	8	5	5	6	6	5	5	6	5	7
6	Q | p8 | 2-3	4	5	6	8	9	9	7	6	9	7	8	5	7	6	8	7	7	6	7	8	7	7
6	E | p0 | 1	1	3	5	4	5	7	4	5	3	3	2	3	4	4	1	2	0	3	2	3	2	2
6	E | p0 | 2	4	5	4	7	7	9	8	4	8	7	7	7	7	5	4	7	6	5	5	8	5	8
6	E | p4 | 1-2	2	0	3	1	6	6	3	4	3	5	6	2	3	4	2	3	1	5	3	2	0	4
6	E | p4 | 2-3	4	2	7	6	9	7	6	6	6	7	9	6	8	6	7	7	7	7	5	7	3	7
6	E | p8 | 1-2	3	0	3	2	4	6	4	4	7	3	5	3	6	4	2	3	1	2	2	2	0	3
6	E | p8 | 2-3	4	8	4	7	7	8	6	7	7	7	7	6	9	5	5	6	6	6	5	6	4	6
6	T | p0 | 1	3	3	6	3	5	5	4	5	4	3	6	3	5	1	1	5	3	4	2	4	7	3
6	T | p0 | 2	4	4	7	5	3	6	7	8	6	3	8	5	7	4	5	6	4	6	4	6	6	5
6	T | p4 | 1-2	2	0	7	5	6	4	5	8	6	4	7	5	6	2	2	6	1	4	3	6	6	5
6	T | p4 | 2-3	4	5	6	6	6	5	8	7	4	4	7	5	4	5	4	6	3	6	3	8	8	5
6	T | p8 | 1-2	3	3	5	4	1	5	8	5	8	4	9	5	6	3	2	5	1	4	6	5	7	4
6	T | p8 | 2-3	4	2	7	4	5	6	5	7	7	4	9	6	5	4	2	8	4	7	4	6	6	6
7	ref	6	8	8	10	10	10	7	9	10	10	10	9	9	8	8	9	10	7	9	10	9	9
7	Q | p0 | 1	4	5	7	8	10	8	9	6	8	5	9	5	8	6	7	8	8	7	4	8	5	8
7	Q | p0 | 2	4	8	6	8	10	10	7	6	8	8	10	6	7	7	8	7	8	7	8	9	7	8
7	Q | p4 | 1-2	5	5	7	7	8	7	5	6	8	7	8	6	7	6	7	6	8	8	5	8	5	6
7	Q | p4 | 2-3	4	9	8	9	8	8	6	6	7	7	8	6	7	9	9	6	9	7	8	8	7	9
7	Q | p8 | 1-2	3	7	8	7	7	7	6	6	8	7	9	4	9	6	8	7	8	6	4	8	7	7
7	Q | p8 | 2-3	6	6	8	9	8	8	8	7	8	8	7	6	7	7	9	8	10	7	7	9	7	9
7	E | p0 | 1	2	0	3	2	3	7	1	4	3	3	5	2	4	4	3	1	1	3	0	1	1	2
7	E | p0 | 2	4	7	5	10	8	9	7	5	6	7	9	6	5	7	6	6	6	6	7	7	7	6
7	E | p4 | 1-2	2	0	3	3	4	7	4	4	4	4	3	3	3	5	2	4	1	6	3	4	3	3
7	E | p4 | 2-3	4	6	6	7	9	9	9	7	9	7	7	7	8	6	8	6	9	5	7	7	7	8
7	E | p8 | 1-2	3	0	4	4	4	7	4	5	6	4	6	3	4	4	3	2	2	4	3	6	3	5
7	E | p8 | 2-3	4	6	7	8	7	8	5	6	6	7	8	6	7	5	8	6	8	7	6	8	5	9
7	T | p0 | 1	2	4	2	2	2	5	5	4	8	3	7	2	5	2	2	6	3	3	5	5	7	5
7	T | p0 | 2	4	4	6	6	6	6	6	5	8	3	8	5	6	5	4	6	1	4	5	6	7	7
7	T | p4 | 1-2	3	3	4	2	0	5	5	5	6	3	6	5	4	2	1	6	3	7	4	5	6	5
7	T | p4 | 2-3	5	2	5	4	2	5	8	6	7	3	9	6	4	5	3	7	4	4	6	5	5	5
7	T | p8 | 1-2	2	2	4	2	2	4	8	4	6	4	8	3	6	2	2	6	4	3	4	5	4	3
7	T | p8 | 2-3	4	2	4	4	5	5	8	6	5	4	9	3	6	4	2	6	1	5	5	6	6	6
8	ref	7	10	7	10	10	8	9	10	10	9	10	9	9	7	10	8	10	9	10	9	9	9
8	Q | p0 | 1	3	0	6	6	9	7	5	5	5	7	6	4	6	4	6	4	7	3	8	5	6	6
8	Q | p0 | 2	5	3	7	7	8	9	6	8	8	7	8	5	8	8	8	4	7	4	9	5	5	7
8	Q | p4 | 1-2	4	0	5	7	9	7	6	5	7	6	9	4	6	7	7	6	6	6	8	5	5	7
8	Q | p4 | 2-3	5	2	5	7	10	8	5	8	8	7	6	6	8	6	8	5	9	6	8	6	8	8
8	Q | p8 | 1-2	4	0	7	6	9	8	7	7	6	5	9	5	7	5	8	4	7	5	8	5	6	7
8	Q | p8 | 2-3	5	1	5	8	9	8	8	7	9	7	8	6	7	8	8	6	8	5	8	7	8	8
8	E | p0 | 1	2	0	6	4	4	7	0	1	3	3	3	2	3	3	2	1	0	3	0	1	0	2
8	E | p0 | 2	3	1	6	5	8	9	2	4	5	7	6	3	6	6	7	6	6	5	5	6	8	7
8	E | p4 | 1-2	1	0	0	0	4	7	0	4	2	3	3	2	4	4	2	2	2	4	3	1	0	4
8	E | p4 | 2-3	4	2	5	5	7	8	4	5	6	6	8	4	6	6	8	7	7	6	4	4	7	8
8	E | p8 | 1-2	2	2	2	1	3	5	0	2	2	3	6	1	5	3	3	4	0	4	1	3	3	4
8	E | p8 | 2-3	4	4	6	5	6	8	5	5	9	6	8	5	6	5	7	7	4	6	6	7	6	7
8	T | p0 | 1	2	0	3	4	3	5	0	10	7	4	5	3	4	3	1	4	3	2	1	1	3	3
8	T | p0 | 2	4	2	3	2	5	5	0	3	7	5	6	3	6	2	3	5	4	4	4	4	10	5
8	T | p4 | 1-2	3	0	7	3	2	5	0	3	8	3	8	3	5	2	1	4	1	5	4	1	4	4
8	T | p4 | 2-3	2	0	4	4	1	4	1	4	7	4	9	3	6	2	4	6	4	5	4	3	5	4
8	T | p8 | 1-2	2	2	5	1	4	5	0	9	5	4	7	3	5	2	2	5	0	5	2	2	8	4
8	T | p8 | 2-3	3	3	3	4	5	5	1	5	5	3	8	4	5	3	3	6	3	4	3	1	5	7