

**Instituto Federal de Educação, Ciência e
Tecnologia do Espírito Santo**

Alexandre Jastrow da Cruz

Maikysuel Simões Braga

**Relatório do Trabalho 3º de
Fundamentos de Sistemas Multimídia**

Vídeo

Serra, Espírito Santo
2020

Alexandre Jastrow da Cruz

Maikysuel Simões Braga

Relatório do Trabalho 3º de Fundamentos de Sistemas Multimídia – Vídeo

Trabalho sobre vídeo apresentado ao curso de Sistemas de Informação para avaliação parcial da disciplina de Fundamentos de Sistemas Multimídia.

Serra, 03 de setembro de 2020

Professor: *Flávio Giraldele Bianca*

Sumário

1. INTRODUÇÃO	4
2. OBJETIVO	4
3. DESENVOLVIMENTO	4
3.1. Atividade 1: Investigando formatos e características de arquivos de vídeo	4
3.1.1. 1A - Investigando os arquivos de vídeo fornecidos.	4
3.1.2. 1B - Investigando vídeos originados de HDTV Europeu	6
3.1.3. 1C - Investigando os formatos usados pelo Youtube	6
3.1.4. 1D - Investigando a câmera do seu smartphone e os transcoders feitos pelo Whatsapp e Telegram	10
3.2. Atividade 2: Comparando diferentes subsamplings	12
3.3. Atividade 3: Analisando as características dinâmicas do vídeo	14
3.3.1. 3A - Analisando um vídeo em slow motion	14
3.3.2. 3B - Analisando uma cena “aleatória” de Matrix (1999)	15
3.3.3. 3C - Analisando um episódio inteiro de Game of Thrones	16
3.4. Atividade 4: Conversão de vídeo	17
3.4.1. 4A - Comprimindo para H.264/AVC	17
3.4.2. 4B - Comprimindo para H.265/HEVC	18
3.4.3. 4C - Comprimindo para VP9	18
3.4.4. 4D - Avaliação geral do desempenho dos formatos	19
4. Link dos arquivos de vídeo gerados	20
5. CONCLUSÃO	20
REFERÊNCIAS BIBLIOGRÁFICAS	20

Lista de Tabelas

Tabela 1 – Formatos de vídeo do Youtube	7
Tabela 2 - Formato AV1 no Youtube.....	10
Tabela 3 - Vídeos gravado com o celular.....	10
Tabela 4 - Vídeos enviados pelo Telegram.....	11
Tabela 5 - Vídeos enviados pelo Whatsapp.....	12
Tabela 6 - Compressão lossless com WinRAR e Winzip	13
Tabela 7 - Compressão H.264/AVC.....	17
Tabela 8 - Compressão H.265/HEIC.....	18
Tabela 9 - Compressão VP9.....	18
Tabela 10 - Desempenho entre vários formatos	19

1. INTRODUÇÃO

O que seria do mundo dos Sistemas de Informação sem os incríveis recursos de áudio, imagem e vídeo? Seria um campo chato e cheio de textos, tabelas e gráficos sem fim. Nesse cenário cada vez mais competitivo, aqueles que tiverem habilidades e conhecimentos em áudio, vídeo, comunicações e TI o suficiente para convergir tudo em algo de qualidade, serão os que terão mais chance de prosperar com grande diferencial. E de uma parte desses temas que este trabalho irá tratar, o vídeo.

2. OBJETIVO

Conhecer diversos softwares usados na área de multimídia, em especial no campo de vídeo digital. Comparar diferentes formatos de compressão lossy (com perdas) aplicados a vídeo. Investigar as características de diferentes formatos de vídeo em diferentes situações. E julgar a qualidade subjetiva de diferentes formatos de vídeo lossy.

3. DESENVOLVIMENTO

3.1. Atividade 1: Investigando formatos e características de arquivos de vídeo

3.1.1. 1A - Investigando os arquivos de vídeo fornecidos.

- **Coldplay - The Scientist [Glastonbury Festival 2011]:** Este é um arquivo do tipo container de extensão, mkv (matroska), que encapsula dois formatos de mídia, sendo o primeiro um vídeo do formato MPEG versão 2, que é um padrão antigo e pouco eficiente do grupo MPEG, comparado aos padrões atuais. Provavelmente utilizado por ser padrão de transmissão das emissoras de TV da região. Possui uma resolução Full HD (16:9), e um bitrate de 45.2 Mb/s, que é consideravelmente alto, necessitando de uma rede robusta para transmissão em tempo real. Tem 25 frames por segundo, adotando o padrão europeu de exibição na tv. A segunda mídia se trata de um áudio, no formato MP2, sendo antigo e de pior qualidade em comparação ao seu sucessor MP3, mas como ela possui uma

taxa de amostragem boa de 48 khz e um bitrate de 256 kb/s, pode garantir a transparência do áudio de 2 canais.

- **Game of Thrones S04E10:** Este arquivo como o anterior também se trata de um container mkv, com dois arquivos de mídia em seu interior, sendo um vídeo e um áudio. Este segundo possui um bitrate bastante alto para o formato AC-3, de 384 kbps e 6 canais de áudio, possivelmente o áudio foi projetado dessa forma para a exibição em home theaters. Já o arquivo de vídeo se trata de uma mídia H.264/AVC com 24 frames por segundo e resolução HD(1280x720), possivelmente essa mídia originalmente foi projetada para gravação em um DVD, necessitando que o aparelho seja capaz de reproduzir mídia de no mínimo level 4.1 no padrão MPEG H.264.
- **Slow Motion:** Este é um tipo de arquivo utilizado pela Apple para codificação de mídias para sua plataforma, que também podem ser vistos por outros decoders que sigam o padrão MPEG-4, pois o conteúdo interno se trata de um vídeo H.264/AVC de level 3.1 com 2 frames de referência. Essa mídia possui um bitrate de 4.33 Mb/s e resolução HD com taxa de quadros de 25 fps. Com essas informações nos parece ser um modelo exibido na europa por dispositivos da Apple. O áudio é um AAC com bitrate de 128 kb/s, que é o padrão para este formato.
- **The Matrix (1999) Versão 1:** Este arquivo é um container mkv com 2 formatos de mídias internas, o primeiro se trata de um vídeo no formato H.264/AVC de level 4.1 com 5 frames de referência. Parece que esta mídia preza pela qualidade, tendo uma resolução de 1920x800(2.40:1) e 24 FPS, gastando uma boa quantidade de espaço em disco. A segunda mídia é um áudio de um bitrate de 1510 kbps e 6 canais no formato DTS. A junção dessas mídias parece necessário para reprodução em locais com uma qualidade de reprodução diferenciada.
- **The Matrix (1999) Versão 2:** Como o anterior, também se trata de um container mkv com duas mídias, porém essa mídia se difere bastante sendo uma versão mais moderna do grupo MPEG no formato H.265/HEVC e resolução de 3840x1600(2.40:1). Parece ser um formato para exibição em projetores, utilizando uma taxa de quadros de 24 fps, essa mídia usa do HDR para obter cores mais realistas e contrastes mais profundos, isso porém tem um custo no

bitrate que é de 40.8Mb/s. O arquivo de áudio de 6 canais e um bitrate de 640kbps no formato AC-3, fortalece a teoria de que esse arquivo como um todo, seja usado em salas de cinema particular onde se tem uma qualidade um pouco maior do que a do usuário comum.

3.1.2. 1B - Investigando vídeos originados de HDTV Europeu

Assistindo o vídeo em velocidade normal no MPC-HC, tudo que foi possível notar foi a granulação da imagem, principalmente em partes onde se tinha pouca luz. Já com o AVIDemux, fica nítido que esse vídeo tem o entrelaçamento da imagem, isso se percebe com as várias linhas horizontais que aparecem durante todo o vídeo, além dos fantasmas que ela deixa quando ocorre uma movimentação mais acentuada. Ainda em alguns pontos do vídeo é possível notar que algumas dessas linhas não casa com as outras ficando uma espécie de risco na imagem.

Fazendo a análise das informações desse vídeo, é sim possível fazer a transmissão ao vivo, porém é necessária uma internet com conexão consideravelmente boa. Para fins de comparação, uma transmissão em 4k está em torno de 35 a 45 Mbps, e o vídeo está nesse range, só que estamos falando de um vídeo com taxa de quadros de 25 fps em resolução fullHD, bem diferente dos 4k dado como exemplo. Então concluindo, é possível, mas não recomendado do jeito que o arquivo se encontra.

3.1.3. 1C - Investigando os formatos usados pelo Youtube

Video - Best of 8K HDR 60p (FUHD)					
Resolução	Frame Rate	Codec	Container	Bitrate (kb/s \cong)	Tamanho (MB \cong)
7680x4320	60	VP9	MKV	20432	646
7680x4320	60	VP9	WebM	20400	646
3840x2160	30	VP9 LQ	MKV	16128	510
3840x2160	30	VP9 LQ	WebM	16129	509
3840x2160	60	VP9 HDR	MKV	29000	917
3840x2160	60	VP9 HDR	WebM	29000	917
3840x2160	60	VP9	MKV	24256	767
3840x2160	60	VP9	WebM	24224	766
2560x1440	30	VP9	MKV	7144	226
2560x1440	30	VP9	WebM	7112	225
2560x1440	60	VP9 HDR	MKV	16320	516
2560x1440	60	VP9 HDR	WebM	16320	516
2560x1440	60	VP9	MKV	10752	340

2560x1440	60	VP9	WebM	10720	339
1920x1080	30	H.264	MP4	3064	97
1920x1080	30	VP9	MKV	2336	74
1920x1080	30	VP9	WebM	2304	73
1920x1080	60	H.264	MP4	4960	154
1920x1080	60	VP9 HDR	MKV	6768	214
1920x1080	60	VP9 HDR	WebM	6736	213
1920x1080	60	VP9	MKV	3600	114
1920x1080	60	VP9	WebM	3600	114
1280x720	30	H264	MP4	880	28
1280x720	30	VP9	MKV	1328	42
1280x720	30	VP9	WebM	1296	41
1280x720	60	H.264	MP4	2752	87
1280x720	60	VP9 HDR	MKV	557	141
1280x720	60	VP9 HDR	WebM	4456	140
1280x720	60	VP9	MKV	2088	66
1280x720	60	VP9	WebM	2056	65
854x480	30	H264	MP4	600	19
854x480	30	VP9	MKV	728	23
854x480	30	VP9	WebM	896	22
854x480	60	VP9 HDR	MKV	1960	62
854x480	60	VP9 HDR	WebM	1928	61
640x360	30	H.264	MP4	408	13
640x360	30	VP9	MKV	440	14
640x360	30	VP9	WebM	440	14
640x360	60	VP9 HDR	MKV	1072	34
640x360	60	VP9 HDR	WebM	1040	33
426x240	30	H.264	MP4	248	8
426x240	30	VP9	MKV	280	9
426x240	30	VP9	WebM	248	8
426x240	60	VP9 HDR	MKV	536	17
426x240	60	VP9 HDR	WebM	536	17
256x144	30	H.264	MP4	192	6
256x144	30	VP9	MKV	192	6
256x144	30	VP9	WebM	192	6
256x144	60	VP9 HDR	MKV	312	10
256x144	60	VP9 HDR	WebM	280	9

Tabela 1 – Formatos de vídeo do Youtube

Nota-se a importante ausência do MOV (QuickTime), utilizado pela Apple, acreditamos que o Youtube não faz uso deles por usar o formato H264 por baixo. E para não haver uma espécie de duplicidade de arquivo desnecessária o Youtube adotou como padrão o MP4. Como se vê na tabela acima, já são muitas versões do mesmo arquivo, e adicionar mais algumas acarretaria em uma despesa considerável de armazenamento.

Também podemos notar que quando se tem a mesma resolução, o bitrate é parecido, com algumas pequenas diferenças que podem beneficiar um ou outro formato a depender da eficiência de compressão do arquivo. O que se destaca mesmo nesse quesito são os formatos com HDR, os 2 bits a mais por canal de cor acarretam em um aumento do tamanho do arquivo que por consequência gera um bitrate de quase o dobro de um formato de mesma resolução.

Agora segue uma lista do que achamos que seja recomendado para diferentes dispositivos:

- **Smart TV Full HD:** acreditamos que o mais indicado para este caso seja um arquivo de resolução Full HD, com container MP4, e formato H.264, a taxa de quadros pode variar entre 30 e 60, mas acreditamos que o indicado seja 60 e caso a internet tenha alguma instabilidade, envia-se para o dispositivo a uma taxa de 30 quadros por segundo;
- **Smart TV 4K Nova:** Neste caso específico se não tivermos problema de conexão, achamos o mais indicado a resolução 4K VP9 HDR no container MKV e 60 FPS, considerando que o aparelho é uma televisão moderna que tem suporte a essas tecnologias;
- **Smartphone antigo:** O formato ideal nesse caso seria uma resolução de 720p ou 480p com boa e média conexão, e 360p ou menos em casos de internet mais lenta ou dependente de dados móveis de operadora, ambos em uma taxa de 30 FPS, e no formato H.264 por ter uma ampla compatibilidade nativa com diversos dispositivos;
- **Smartphone Novo:** Este é um caso interessante, mesmo esses dispositivos tendo telas com grande resolução, acima de Full HD, acreditamos que o indicado para esses dispositivos seja uma resolução Full HD de vídeo a 60 FPS e suporte a HDR, utilizando em casos de conexões mais fracas resoluções abaixo de HD e sem HDR;
- **Desktops e Notebooks:** Esse é um caso complicado, pois existem inúmeras configurações de desktop, com diversos tipos, resoluções de monitor, então acreditamos que com boa conexão, deve-se enviar um vídeo em resolução Full HD a 60 FPS no formato VP9, aí caso a pessoa queira uma resolução diferente ela pode mudar manualmente, em conexões mais fracas, podemos baixar essa

resolução para um HD ou 480p, em casos mais extremos baixar um pouco mais. Agora se analisarmos que o Youtube tem uma inteligência que decida coisas do tipo, “esse *usuário está ouvindo música*”, ou coisa do tipo em background, talvez o ideal seja uma menor resolução, porém com uma alta qualidade de áudio;

- **Tablets:** Tendo uma boa conexão se deve enviar o vídeo de acordo com a resolução da tela do dispositivo, caso contrário envie em resoluções menores como 480p e 360p a 30 FPS.

Não conseguimos notar diferença em questão de qualidade quando analisamos os vídeos em 1080p@60 H.264 e VP9, os dois são bem parecidos inclusive onde se notam os macroblocos, claro que ao darmos pausa nos vídeos no mesmo frame, os macroblocos notados estarão em lugares diferentes, mas em uma análise superficial não foi possível notar diferença de qualidade. Porém quando se fala dos demais quesitos, o VP9 tem por volta de 75% do tamanho do arquivo H.264, então em uma análise superficial não notamos qualquer diferença em questão de qualidade, mesmo com uma redução considerável do tamanho e bitrate.

Quando analisamos os dois arquivos H.264 podemos ver que o que interfere no tamanho, e por consequência no bitrate, é a taxa de quadros por segundo. O interessante notar porém é que apesar de um ter a metade da quantidade de quadros, isso não se reflete no tamanho, enquanto o H.264 com 30 FPS possui um tamanho de 97 MB o vídeo a 60 FPS tem um tamanho de 154 MB, ou seja, teve um aumento de 158% aproximadamente, isso provavelmente se dá por que existe uma maior redundância dos quadros por segundo, fazendo com que um frame completo seja suficiente para reconstruir vários outros com pouca diferença para ele, diferente do arquivo a 30 FPS onde a cada mudança de quadro existe um movimento considerável dos macroblocos, sendo necessário a captura de um frame completo com maior frequência.

Quanto a valer a pena, isso é subjetivo, valeu a pena para quem? Para o Youtube quanto menor o arquivo melhor para ele, enquanto para o usuário quanto maior a qualidade e a fluidez melhor para o usuário. No nosso caso foi possível notar a diferença de fluidez de um para o outro, então para nós é melhor o 1080p@60FPS, o que é ruim para o Youtube que precisa de mais armazenamento e conexão, ruim também para as operadoras de internet que precisam transmitir um vídeo de maior bitrate.

Vídeo - Japan in 8K					
Resolução	Frame Rate	Codec	Container	Bitrate (kb/s \cong)	Tamanho (MB \cong)
3840x2160	24	VP9	MKV	17255	297
3840x2160	24	AV1	MP4	12200	210
1920x1080	24	H264	MP4	3137	54
1920x1080	24	AV1	MP4	1917	33
1280x720	24	H264	MP4	1801	31
1280x720	24	AV1	MP4	1103	19

Tabela 2 - Formato AV1 no Youtube

O formato AV1 está presente em todas as resoluções, já o formato H.264 só não está presente na resolução 4K. O Youtube optou por fazer dessa forma por que este formato não é muito otimizado para trabalhar nesta resolução, assim ele escolheu pelo formato VP9 do próprio Google. Na questão fluidez todos deixam a desejar, 24 FPS consideramos pouco. No quesito compatibilidade já tivemos um problema, com o AV1 como é um formato recente, foi necessário procurar por codecs para sua reprodução. Já na qualidade não foi notado diferenças em altas resoluções, mas quando olhamos para a resolução HD podemos ver, que diferente da H.264, ela disfarça melhor a predição de movimento, além de ter um tamanho bem menor que os demais formatos.

Aparentemente o AV1 é bem promissor, com uma qualidade similar ou melhor que seus concorrentes e com uma taxa de compressão consideravelmente maior. O que nos deixou intrigados foi esse frame rate, porque 24? Então após discutirmos sobre, achamos que esse é um teste para ver o quão bom o AV1 é bom em fazer as predições de movimento em relação aos demais, para a reconstrução dos demais quadros.

3.1.4. 1D - Investigando a câmera do seu smartphone e os transcoders feitos pelo Whatsapp e Telegram

Vídeo Original						
Resolução	Frame Rate	Codec	Container	Tamanho (Segundos)	Bitrate (kb/s \cong)	Tamanho (MB \cong)
1920x1080	30	H.264	MP4	60	19660	144
1920x1080	60	H.264	MP4	59.919	19660	144
3840x2160	30	H.264	MP4	60	41369	303
3840x2160	60	H.264	MP4	59.730	41165	300

Tabela 3 - Vídeos gravado com o celular

Agora temos alguns resultados interessante, praticamente não existe diferença entre os arquivos de mesma resolução, exceto pelo fato de um estar a 30 fps e o outro a 60 fps, diferentemente dos arquivos baixados do Youtube, onde se teve um aumento substancial do tamanho do vídeo ao mudar a taxa de quadros, aqui não ocorreu isso.

Quando analisamos a qualidade da gravação é nítida a diferença de fluidez, a questão então agora é, por que não vemos uma diferença no tamanho parecida com a dos vídeos do Youtube? Após discutirmos sobre, acreditamos que o que ocorre é uma excelente predição de movimento. A 30 fps existem diferenças muito brusca de movimento, fazendo com que se tenha que guardar mais frames completos, acarretando no aumento do tamanho do vídeo, enquanto o vídeo a 60 fps, existe uma transição mais suave de quadro para quadro, fazendo com que um frame completo dure por mais tempo na predição de movimento.

Vídeo Telegram				
Resolução original	Qualidade envio	Resolução Telegram	Bitrate (kb/s \cong)	Tamanho (MB \cong)
1080p@30	Alta	1280x720	2600	20
	Média	854x480	994	19.8
	Baixa	480x270	747	6
1080p@60	Alta	1280x720	2614	19.4
	Média	854x480	994	7.8
	Baixa	480x270	745	6
4k@30	Alta	1280x720	2618	20
	Média	854x480	997	8
	Baixa	480x270	748	6
4k@60	Alta	1280x720	2618	19.4
	Média	854x480	995	7.8
	Baixa	480x270	746	6

Tabela 4 - Vídeos enviados pelo Telegram

Olhando para a tabela dá pra notar um padrão de envio pelo Telegram, com alta resolução ele diminui para 720p em boa qualidade, 480p com a qualidade no médio e quando se escolhe a opção de baixa qualidade ele manda em uma resolução de 270p. Outra coisa a se notar, é que mesmo em sua melhor qualidade o vídeo fica cheio de artefatos e granulação, bem diferente da versão original, em questão de tamanho praticamente não existe diferença de 30 para 60 fps. Outro ponto é que o vídeo original ficou salvo em H264/AVC - Level 5.1, para o video em 4k@30, no 4k@60 o Level é

modificado para 5.2, já no 1080p@30 e 1080p@60 foram salvos no Level 4 e 4.1 respectivamente. Quando esses vídeos são enviados pelo Telegram e há o transcoding, o Level utilizado é do 3.1 para baixo, fazendo com que se tenha um range maior de dispositivos que podem executar esse vídeo.

Vídeo WhatsApp			
Resolução original	Resolução WhatsApp	Bitrate (kb/s \cong)	Tamanho (MB \cong)
4k@30	640x352	1265	9.8
4k@60	640x352	1269	9.7
1080@30	640x352	1266	9.7
1080@60	640x352	1267	9.7

Tabela 5 - Vídeos enviados pelo Whatsapp

Diferentemente do Telegram, o WhatsApp não dá opções de qualidade no envio, assim eles simplesmente fazem o que querem, como podemos ver nesta tabela, praticamente não há diferença de 4k para FullHD, e a resolução que de fato é enviada, é bastante baixa, isso reflete bastante na qualidade do vídeo enviado, que é bastante sujo, cheio de artefatos, e com pouca margem para dar zoom.

]O interessante notar é que se pegarmos a resolução do Telegram 480p e compararmos com a resolução 352p do WhatsApp, percebe-se que o telegram consegue comprimir com maior qualidade em termos visuais e de espaço em disco.

3.2. Atividade 2: Comparando diferentes subsamplings

O nome dos arquivos possui três informações distintas. As partes 420, 422 e 444 são respectivamente 4:2:0, 4:2:2 e 4:4:4, que se referem ao *Chroma Subsampling*, um esquema de codificação de imagens e vídeos através da implementação de componentes para alteração nas cores e luminância, aproveitando-se assim do sistema de visão humana que tem menor capacidade para diferenciar cores do que luminosidades. Esse esquema de codificação faz com que imagens e vídeos tenham um tamanho menor em bits.

A segunda parte, 720p, diz respeito à resolução 1280 x 720 pixels, também conhecida como resolução HD. Enquanto a terceira parte, 50, informa a quantidade de frames por segundo que há no vídeo.

Cálculo do tamanho dos arquivos:

1. Arquivo *in_to_tree_420_720p50*:
Pixels por frame = $1280 \times 720 = 921600$ pixels
YUV420 = 1,5 bytes por pixel
Bytes por frame = 1382400 bytes
Frames = 500
Tamanho total = $1382400 \times 500 = 691.200.000$ bytes = **675.000 KB**
2. Arquivo *in_to_tree_422_720p50*:
Pixels por frame = $1280 \times 720 = 921600$ pixels
YUV422 = 2 bytes por pixel
Bytes por frame = 1843200 bytes
Frames = 500
Tamanho total = $1843200 \times 500 = 921.600.000$ bytes = **900.000 KB**
3. Arquivo *in_to_tree_444_720p50*:
Pixels por frame = $1280 \times 720 = 921600$ pixels
YUV444 = 3 bytes por pixel
Bytes por frame = 2764800 bytes
Frames = 500
Tamanho total = $2764800 \times 500 = 1382400000$ bytes = **1.350.000 KB**

Como os arquivos não possuem metadados, eles ficaram com exatamente os mesmos valores mostrados no Windows.

Testando a compressão em Lossless utilizando o WinRAR e WinZip.

Compressão Lossless		
Original	RAR	ZIP
675000	64%	66%
900000	63%	64%
1350000	60%	62%

Tabela 6 - Compressão lossless com WinRAR e Winzip

Como podemos ver na Tabela 6, a compressão ficou em média 63% do tamanho do arquivo original, que é um valor razoavelmente satisfatório, considerando que foi utilizado um compressor genérico.

Interessante observar que quanto mais bits por pixel, maior foi a compressão do arquivo. Podemos inferir que quanto maior a amostragem do *chroma subsampling*, mais bits redundantes temos.

Após assistir os três vídeos de forma natural, foi constatado que todos eles são idênticos

perante a visão humana, exatamente como ocorreu nas imagens em JPEG. Mesmo ao comparar frame a frame, nenhuma mudança foi percebida. A princípio suspeitamos que o formato 4:2:0 teria prejudicado o vídeo, pois as paredes do prédio possuíam um comportamento artificial, como se fosse um granulado se movimentando, todavia, mesmo em 4:4:4 essa característica se manteve.

Foram feitas todas as combinações possíveis nos canais Y, U e V e absolutamente nenhuma diferença foi percebida. Como estudado previamente em imagens e revisto em vídeos, o canal Y (da luminância) é o que possui a maior quantidade de informações, mantendo a imagem praticamente como a original, porém em tons de cinza. Enquanto os canais U e V adicionam as cores, entretanto isoladamente, estes possuem pouquíssima informação, tornando o vídeo quase incompreensível.

O único parâmetro que muda entre os três é a taxa de amostragem do *chroma subsampling*, logo se os três tivessem a mesma taxa, o tamanho deles seriam idênticos.

A diferença do tamanho do formato YUV444 para o YUV420 é de 50%, com um altíssimo nível de transparência, o que é bem satisfatório, sendo que ainda poderiam ser usadas outras técnicas de compressão para diminuir mais ainda o tamanho dos arquivos, como a predição temporal e predição espacial por exemplo.

3.3. Atividade 3: Analisando as características dinâmicas do vídeo

3.3.1. 3A - Analisando um vídeo em slow motion

Em primeira análise, verificamos apenas o bitrate de cada cena, analisando o tamanho de cada quadro. O gráfico de barras foi fundamental nessa análise. Assim como em nossos vários testes em áudio, aqui percebemos a entropia de cada cena. Cenas como o jogador acertando a bolinha com o taco ou a bolinha de pelos verde batendo no chão tem menores taxas de bitrate, as cenas possuem grandes áreas de uma mesma cor e estáticas, resultando em baixa complexidade espacial. Enquanto cenas como das cheerleaders ou dos vários jogadores com tacos têm quadros com tamanhos maiores, as cenas além de mais coloridas, possuem mais movimentos. Uma cena interessante é a do cuspidor de fogo que já começa com uma boa taxa de bitrate, que vai subindo gradativamente à medida que o fogo vai se expandindo, um crescimento nítido da entropia na cena.

Os vetores de movimento são um caso à parte, é muito interessante prestar atenção neles e pensar que eles estão prevendo para onde seu respectivo macrobloco vai.

No início as coisas parecem bastante confusas, mas logo ficam fáceis de perceber, como nos pompons, o fogo se expandindo ou na mais clara de todas, a bola sendo chutada.

O vídeo começa com um frame do tipo I obviamente, mas a surpresa é que o segundo frame já é do tipo B, ele já usa uma predição bidirecional, utilizando como referência o frame anterior, o I-frame e o frame seguinte, o P-frame. O I-frame sempre possui um tamanho muito maior, pois o frame inteiro é novidade, o P-frame possui um tamanho médio enquanto o B-frame é muitíssimo menor. O B-frame certamente é muito menor por utilizar diferenças entre o frame atual e frames futuro e passado.

Muito provavelmente pelo fato do vídeo estar em super câmera lenta e a diferença entre os frames serem muito pequenas, os frames B e P sempre se alternam, nunca aparecendo dois P-frames seguidos por exemplo. Em intervalos de 3 ou 4 segundos surge um I-frame, para fazer o reinício da referência das predições. Exceto quando a cena muda drasticamente, nesse caso o I-frame surge independente do tempo.

Ao clicar na barra de progresso em algum ponto do vídeo, a cena aparece sem os motion vectors, isso ocorre porque sempre que clicamos na barra de progresso, caímos em um I-frame, e como bem sabemos, esse tipo de frame não possui motion vector.

3.3.2. 3B - Analisando uma cena “aleatória” de Matrix (1999)

Nesse vídeo de cara já fica muito evidente os vetores de movimento muito mais agressivos, com vetores que chegam a cortar a tela de cima a baixo. E momentos que a cena quase some de tantos vetores que aparecem. Claro que a causa disso é mudança brusca da posição dos elementos na tela. Ao pular frame por frame é quase inacreditável que juntando tudo forme um vídeo com boa fluidez.

As mudanças são tão bruscas, no momento da cena que o Neo solta o casaco até golpear o segurança no meio do peito são apenas 5 ou 6 frames, e o mais curioso é que o braço e a mão dele nem aparecem, o nosso cérebro se encarrega de construir todo o restante da imagem. Toda essa diferença entre um frame e outro faz com que em vários momentos do vídeo, não apareça B-frames, surgindo vários P-frames em sequência.

3.3.3. 3C - Analisando um episódio inteiro de Game of Thrones

Esse vídeo possui um bitrate médio de 3 436 kb/s. Ele inicia com a logo da HBO que inicia com um ruído branco em toda a tela, resultando nas maiores taxas de bitrate de todo o vídeo, 10x maior que o bitrate geral, batendo um pico de 34462 kb/s. Como bem sabemos, ruído tem altíssima taxa de aleatoriedade, dificultando em muito a compressão.

Seguindo pela introdução, a primeira observação é em relação a complexidade temporal, quanto mais rápido a “câmera” passeia pelo mapa, maior é o bitrate, junte isso há uma mudança brusca de cena (complexidade espacial) e temos as maiores taxas durante a introdução.

Ao analisar os frames individualmente foi notada uma característica inédita até então, existem dois tipos de B-frame, um menor e outro que fica com tamanho entre o B-frame menor e o P-frame. Os I-frames aparecem em cada mudança de cena. Os P-frames possuem *motion vector* muito mais acentuados que os B-frames, sendo que os B-frames menores são os que possuem *motion vectors* mais “tímidos”, ficando então diretamente ligados ao tamanho do frame.

O episódio analisado inicia com um bitrate abaixo da média, a complexidade espacial é muito baixa, com cenas escuras, praticamente em tons monocromáticos e de pouca informação, que provavelmente só não tem um bitrate menor ainda pela grande quantidade de motion vectors.

As cenas de batalha no gelo com bastante contraste praticamente dobraram as taxas de bitrate. O episódio segue com uma boa coerência, com taxas de bitrate equivalente a complexidade temporal e espacial. A cena final, quase uma imagem estática, possui uma taxa alta de bitrate devido à complexidade espacial, que assim como estudado em imagens, possui muitas áreas de alta frequência, o mar está bem agitado, o navio faz muitas bordas abruptas com o cenário e o céu possui muitas nuvens.

3.4. Atividade 4: Conversão de vídeo

3.4.1. 4A - Comprimindo para H.264/AVC

Compressão H.264/AVC						
Vídeo	Qualidade RF	Tempo (Sec)	Bitrate (kb/s \cong)	Tamanho (MB \cong)	Ratio Original (%)	Ratio Original Lossless(%)
Alta Qualidade	25	321	2427	176	0.414	2.61
Baixa Qualidade	31	286	1241	90	0.211	1.33
Qualidade VCE - Alta	25	665	4633	336	0.791	4.98

Tabela 7 - Compressão H.264/AVC

Antes de qualquer coisa vimos uma coisa realmente estranha nesta tarefa, a aceleração por hardware da AMD/VCE foi mais lenta do que usar o codec x264, além de aumentar o tamanho consideravelmente, conferimos, comprimimos de outras formas, com encoder no fast, ainda assim foi mais lento, a única explicação que encontramos para isso é que a placa de vídeo usada, AMD RX580, ou não tem suporte, ou é mau otimizada para esse tipo de tarefa, os arquivos de log estarão junto com os arquivos de vídeo ao fim do trabalho.

Quando olhamos para o vídeo de qualidade alta, é possível notar que a compressão dela para a versão original e lossless é gigantesca, sem uma perda muito evidente de qualidade, na versão de baixa qualidade, porém ainda aceitável, conseguimos comprimir ainda mais, porém com algumas perdas consideráveis. Quando olhamos para componentes de baixa frequência como o céu, é visível os macroblocos se ajustando com o decorrer da animação, sendo assim consideramos que se esses vídeos forem usados para transmissão, via Youtube ou Twitch por exemplo, a menos que a conexão seja bastante limitada, o ideal é usar a versão de alta qualidade.

3.4.2. 4B - Comprimindo para H.265/HEVC

Compressão H.265/HEVC						
Vídeo	Qualidade RF	Tempo (Sec)	Bitrate (kb/s \cong)	Tamanho (MB \cong)	Ratio Original(%)	Ratio Original Lossless(%)
Alta Qualidade	28.5	424	1227	89	0.209	1.32
Baixa Qualidade	35	319	620	45	0.105	0.66
Qualidade VCE - Alta	28	955	2854	207	0.48	3.07

Tabela 8 - Compressão H.265/HEVC

Ao analisarmos o codec x265 vimos que ele cumpre o que promete, usando ele, temos uma diminuição do tamanho, e por consequência o bitrate, em termos gerais, pela metade e mantendo a qualidade, claro que quando analisamos qualidade de transparência, a qualidade aceitável é subjetiva, o vídeo com codec x265 pode estar melhor ou pior do que o vídeo de codec x264, mas entramos em um consenso que as diferenças existentes nas duas configurações são imperceptíveis.

Por desengargo de consciência, foi feito a compressão do vídeo no fator de qualidade 25 do HandBrake, igual ao x264 feito anteriormente, para ver do que o codec era capaz, após fazer isso tivemos um vídeo com um ratio de 76.7% em comparação as mesmas configurações, mas com codec x264 nota-se uma evolução considerável. Novamente ocorreu da compressão usando H265(AMD VEC), de ter um acréscimo no tempo e tamanho.

3.4.3. 4C - Comprimindo para VP9

Compressão VP9						
Vídeo	Tempo (Sec)	Bitrate (kb/s \cong)	Tamanho (MB \cong)	Ratio x265 (%)	Ratio Original (%)	Ratio Original Lossless (%)
Alta Qualidade	3039	1227	99	111.2	0.233	1.47
Baixa Qualidade	2610	620	55	122.2	0.011	0.81

Tabela 9 - Compressão VP9

Diferente dos anteriores, não foi usado o fator de qualidade em busca da transparência e qualidade aceitável, pois se mostrou ineficiente mesmo colocando a qualidade no mínimo, o arquivo resultante teve um tamanho de mais que o dobro do seu concorrente direto, H.265. Sendo assim o que foi feito nesta etapa foi a codificação do vídeo por bitrate médio com duas passadas de encoding, usando como valor o bitrate dos vídeos resultantes da codificação do x265, para que pudesse ser feita uma comparação direta entre os dois formatos. O problema mais evidente dessa abordagem foram os tempos de codificação do formato, as demais comparações serão feitas no próximo tópico.

3.4.4. 4D - Avaliação geral do desempenho dos formatos

Desempenho dos formatos						
Vídeo	Qualidade	Tempo (Sec)	Bitrate (kb/s \cong)	Tamanho (MB \cong)	Ratio Original (%)	Ratio Original Lossless (%)
Original			569935	41465	100	629
Lossless			92751	6748	15.8	100
x264 Alta	25	321	2427	176	0.414	2.61
x264 Baixa	31	286	1241	90	0.211	1.33
x264 Alta AMD ACE	25	665	4633	336	0.791	4.98
x265 Alta	28.5	424	1227	89	0.209	1.32
x265 Baixa	35	319	620	45	0.105	0.66
x265 Alta AMD ACE	28	955	2854	207	0.48	3.07
VP9 Alta		3039	1227	99	0.233	1.47
VP9 Baixa		2610	620	55	0.011	0.81

Tabela 10 - Desempenho entre vários formatos

Quando avaliamos os formatos H.264 e H.265, é possível notar que o desempenho prometido pelo grupo MPEG é cumprido, porém por diversos de regras complicadas de patente, faz com que sua adoção não seja amplamente adotada.

Comparando o VP9 com o H.265, tivemos um problema, para que houvesse uma comparação mais justa tivemos que comprimir o VP9 por bitrate médio, pegando como base os valores dos arquivos gerados pelo x265. Comparando os resultados vimos que o VP9 após comprimido tem uma boa qualidade de imagem em boa parte do video, mas quando ocorre uma transição de um pedaço do vídeo mais parado, com pouca energia, para um mais movimentado com muita energia, há uma perda de qualidade considerável por alguns instantes, é como se o vídeo tomasse um susto e demorasse um pouco para se recuperar.

O tempo para fazer essa codificação foi bastante elevado, chegando a ser quase dez vezes maior que seu concorrente direto, x265. Ainda teve um acréscimo no tamanho de 10Mb nas duas resoluções, fazendo com que o H.265 seja um formato melhor hoje para ser utilizado. O problema que existe mesmo são questões de direitos e patentes que fazem o formato ser menos adotado.

Para usuários como aqueles que fazem upload para o Youtube ou sites do gênero, o ideal seria uma codificação do x265, já que a questão de direitos fica por parte das empresas, elas que se virem por lá para converter para o formato que melhor lhe convém, o ideal é mandar o vídeo com a melhor qualidade possível e viável para a internet do usuário.

4. Link dos arquivos de vídeo gerados

4.1. Atividade 1D

- Arquivo original: [original](#)
- Arquivo Compressão do Telegram: [telegram.rar](#)
- Arquivo Compressão do Whatsapp: [Whatsapp.rar](#)

4.2. Atividade 4A

- [H264/AVC](#)

4.3. Atividade 4B

- [H265/HEIC](#)

4.4. Atividade 4C

- [VP9](#)

5. CONCLUSÃO

Conclusão feita em vídeo, disponibilizado no link abaixo:

REFERÊNCIAS BIBLIOGRÁFICAS

Slides professor Flávio Giraldeli – IFES – 2021 – SERR