Sumarização Automática de Vídeos Baseada em Características de Cor e Agrupamentos

Sandra Eliza Fontes de Avila
Universidade Federal de Minas Gerais
Departamento de Ciência da Computação
Belo Horizonte, Minas Gerais, Brasil
sandra@dcc.ufmg.br

Resumo

Neste trabalho é proposta uma abordagem simples e eficiente para a sumarização automática de vídeos. O método é baseado na extração das características de baixo-nível das imagens e no algoritmo de agrupamento k-means. Os testes foram realizados em vídeos extraídos do repositório The Open Video Project. Os resumos produzidos foram avaliados através de usuários e também foram comparados com os resumos da base de vídeos utilizada. Os resultados mostraram que o método proposto é uma solução alternativa para o problema da sumarização automática de vídeos.

1. Introdução

Os avanços em técnicas de compressão, a diminuição no custo de equipamentos para aquisição e armazenamento vídeo e, ainda, a disponibilidade de meios de transmissão de dados em alta velocidade, têm facilitado a forma como os vídeos são criados, armazenados e distribuídos. Como conseqüência, os vídeos passaram a ser utilizados em várias aplicações. Devido ao aumento na quantidade de vídeos distribuídos e utilizados em aplicações atuais, a pesquisa e o desenvolvimento de novas tecnologias são necessários para um gerenciamento mais eficiente destes dados. Entre as diversas áreas possíveis de pesquisa, a sumarização automática de vídeos é uma etapa essencial para inúmeras aplicações de vídeos, tais como indexação, navegação e recuperação por conteúdo [28].

Sumarização de vídeo é o processo de extração de um resumo do conteúdo original do vídeo, cujo objetivo é fornecer rapidamente a informação concisa do conteúdo do vídeo, preservando a mensagem do vídeo original [25]. Recentemente, o resumo automático de vídeos tem atraído o interesse dos pesquisadores devido ao seu potencial comercial em diversas aplicações. Como conseqüência, novos modelos e algoritmos têm sido propostos na literatura da área.

Segundo [19, 28], os tipos de resumos gerados, a partir das técnicas de sumarização automática de vídeos, podem ser classificados em duas categorias principais: *key-frames* ou *video skim*. A primeira categoria, também conhecida como *representative frames*, *still-image abstracts* ou *static storyboard*, consiste na extração de um conjunto de quadros-chave do vídeo original, resultando em resumos estáticos. Já a segunda categoria, também conhecida como *moving-image abstract*, *moving storyboard* ou *summary sequence*, coleta um conjunto de tomadas[1] através de análise da similaridade ou da relação temporal entre os quadros, resultando em resumos dinâmicos. Uma vantagem do uso de resumos dinâmicos é a possibilidade de incluir elementos de áudio e movimentos realçando assim, tanto a expressividade quanto a informação presente no vídeo. Além disso, segundo [19], geralmente é mais interessante para o usuário assistir a um resumo em vídeo, resumo dinâmico, do que ver um conjunto de imagens, resumo estático. Por outro lado, os resumos estáticos permitem ao usuário acessar o conteúdo do vídeo de forma não-linear, pois uma vez que os quadros-chave tenham sido extraídos, existem diversas maneiras de visualizá-los além da seqüência restrita observada nos resumos dinâmicos, como demonstrado em [2, 10, 11, 22]. Estas formas podem permitir ao usuário obter uma compreensão mais rápida do conteúdo do vídeo. Neste trabalho, o método proposto para a sumarização de vídeo está voltado para a produção de resumos estáticos.

Na literatura, diferentes técnicas para gerar resumos estáticos têm sido propostas [3, 7, 12, 13, 15, 24, 32], sendo que a maioria delas baseiam-se em técnicas de agrupamento (*clustering*). Para esta técnica, a idéia é produzir resumos através do agrupamento de quadros/tomadas simila-

1 Uma *tomada* pode ser definida como uma seqüência de imagens que apresenta uma ação contínua no tempo e no espaço que foi capturada por uma única câmera.

res e apresentar um número limitado de quadros por agrupamento (na maioria dos casos, é selecionado um quadro por agrupamento). Nesta abordagem, é importante selecionar o tipo das características que serão utilizadas para representar os quadros (por exemplo, distribuição de cores, vetores de movimento, textura, forma) e medir a similaridade entre eles.

Apesar das técnicas existentes produzirem resumos com qualidade aceitável, elas geralmente utilizam técnicas de agrupamento complicadas que são computacionalmente caras e requerem um alto consumo de tempo [8]. Por exemplo, em [24] o tempo necessário para a produção de um resumo leva cerca de 10 vezes a duração do vídeo. De fato, não é aceitável que um usuário tenha que esperar 20 minutos para ter uma representação concisa de um vídeo que ele poderia ter assistido em apenas dois minutos.

Neste trabalho é proposta uma abordagem simples e eficiente para a sumarização automática de vídeos. O método é baseado na extração das características de baixo-nível das imagens (utilizando o espaço de cor RGB) e no algoritmo de agrupamento *k-means* [23]. Os testes foram realizados em uma amostra de 20 vídeos extraídos do repositório The Open Video Project [1]. Os resumos produzidos foram avaliados através de usuários e também foram comparados com os resumos do Open Video. Os resultados mostraram que o método proposto é uma solução alternativa para o problema da sumarização automática de vídeos.

O artigo está organizado como se segue. Na Seção 2 são apresentados os trabalhos relacionados. A metodologia proposta é descrita na Seção 3. Na Seção 4, os resultados experimentais são discutidos. E por fim, as conclusões e os trabalhos futuros são apresentados na Seção 5.

2. Trabalhos Relacionados

Tipicamente, as soluções encontradas na literatura para o problema de sumarização são divididas em duas fases: inicialmente, o vídeo é segmentado (freqüentemente em tomadas) e em seguida os quadros-chave são extraídos, de acordo com algum critério. As primeiras soluções propostas para o problema selecionava para cada tomada um (o primeiro) [27] ou dois (o primeiro e o último) [29] quadros-chave. Uma clara desvantagem desta abordagem é que, se a tomada apresentar uma dinâmica intensa, o primeiro (ou último) quadro pode não ser o mais representativo. Diante de problemas como este, diferentes abordagens, como métodos que baseiam-se em técnicas de agrupamento, têm sido propostas.

Zhuang et al. [32] propõem uma técnica de agrupamento não-supervisionada para a produção de resumos. O vídeo é segmentado em tomadas e os agrupamentos são obtidos utilizando as características extraídas através do histograma de cor, para o espaço de cor HSV. O algoritmo utiliza um limiar δ para controlar a densidade da classificação. Antes de um quadro ser classificado em um determinado agrupamento, a similaridade entre o quadro e o centróide é computada. Caso este valor seja menor que δ, então o quadro não está próximo o suficiente para ser inserido neste agrupamento. Para cada agrupamento, o quadro mais próximo do centróide é selecionado como quadro-chave. Segundo os autores, o método proposto é eficiente e eficaz. Entretanto, não foram realizadas avaliações do método e dos resumos gerados que possam validar estas afirmações.

Hanjalic e Zhang [15] desenvolvem um procedimento automático para sumarização de vídeo através da validação da análise de agrupamentos [16]. O método proposto está dividido em três etapas. Primeiro, um algoritmo particional de agrupamento é aplicado n vezes para todos os quadros do vídeo, os quais são representados através do histograma de cor para o espaço YUV. O número de agrupamentos inicialmente é igual a um e é incrementado de um em um a cada aplicação do algoritmo. Desta maneira, diferentes possibilidades de agrupamentos são obtidas. Na segunda etapa, a combinação ótima de agrupamentos é determinada automaticamente através da validação da análise de agrupamentos. E por último, para cada agrupamento, o quadro mais próximo do centróide representa o quadro-chave.

Gong e Liu [12] propõem um técnica para sumarização de vídeo baseada na decomposição em valores singulares (do inglês *Singular Value Decomposition* (SVD)). Inicialmente, o vídeo é segmentado em quadros. Para reduzir o número de quadros a serem analisados, os autores utilizam uma sub-amostragem dos quadros (um quadro a cada 10 quadros). A extração de características destes é realizada através do histograma de cor para o espaço RGB. Cada quadro é divido em blocos de dimensão 3 x 3 e para cada bloco é calculado o histograma, gerando assim nove histogramas para cada quadro. Para reduzir a dimensão do vetor de características é aplicado o algoritmo SVD. A partir dos vetores redimensionados, as distâncias entre os vetores são calculadas. O menor valor representar o limiar utilizado para gerar os agrupamentos. Para cada agrupamento, o quadro mais próximo do centróide representa o quadro-chave.

Mundur et al. [24] apresentam um método baseado na triangulação de Delaunay para agrupar os quadros similares do vídeo. Inicialmente, o vídeo é segmentado em quadros e é feita uma sub-amostragem dos quadros (um quadro a cada 30 quadros). Para a extração de características é aplicado o histograma de cor no espaço HSV. Cada quadro é representado por um vetor de características com 256 dimensões e a seqüência do vídeo é representada por uma matriz A. Para reduzir a dimensão da matriz é aplicada a técnica de análise dos componentes principais (do inglês *Principal Component Analysis* (PCA)), que gera uma matriz B de dimensão n x d, onde n é número total de quadros selecionados (sub-amostragem) e d é o número de componentes

principais. Assim, cada quadro é representado por um vetor d-dimensional ou por um ponto no espaço d-dimensional. Em seguida, o diagrama de Delaunay é construído para n pontos em d-dimensões. Os agrupamentos são obtidos de acordo com as arestas de separação no diagrama de Delaunay. A remoção destas arestas identificam os agrupamentos no diagrama. Para cada agrupamento, o quadro-chave é representado pelo quadro mais próximo do centróide. Apesar do método proposto ser inteiramente automático (não exige que o número de agrupamentos seja pré-definido e não utiliza limiares), ele consome muito tempo para gerar um resumo, cerca de 10 vezes a duração do vídeo.

Furini et al. [7] propõem uma abordagem, denominada VISTO (*VIsual STOryboard*), que aplica um algoritmo de agrupamento para selecionar os quadros mais representativos. VISTO está divido em três etapas. Primeiro, os quadros do vídeo são descritos através do histograma de cor no espaço HSV. Cada quadro é representado por um vetor de características com 256 dimensões. Na segunda etapa, os autores utilizam uma versão melhorada do algoritmo *Furthest-Point-First* (FPF) [9] para obter os agrupamentos. Para determinar o número de agrupamentos é calculada a distância par-a-par entre os quadros. Se a distância for maior que um limiar Γ, então o número de agrupamentos é incrementado. A última etapa consiste em eliminar quadros redundantes e inexpressivos (por exemplo, um quadro todo preto). Para avaliar a qualidade dos resumos, os autores propõem uma avaliação subjetiva. Um conjunto de 20 pessoas são questionadas, de acordo com uma escala que varia entre um e cinco, se o resumo gerado representa bem o conteúdo do vídeo original. A média da opinião das pessoas indica a qualidade do resumo. Os resumos gerados pelo método proposto são comparados com os resumos gerados em [24] e os resumos do Open Video.

De acordo com a análise das abordagens pesquisadas na literatura, pôde-se observar que na seleção dos quadros-chave são utilizadas diversas características visuais e estatísticas. Estes atributos podem influenciar consideravelmente no aumento do custo computacional e principalmente na qualidade do resumo resultante. Outro ponto identificado é que a extração de características pode produzir matrizes com dimensões elevadas. Conseqüentemente, técnicas matemáticas são utilizadas para tentar reduzir o tamanho da matriz, como visto em [12, 24], o que requer mais tempo de processamento. Uma deficiência evidente nos trabalhos analisados é a falta da avaliação dos resultados e da comparação com outras abordagens.

3. Metodologia Proposta

Na Figura 1 é ilustrado o diagrama do metodologia proposta para a sumarização de vídeo. Inicialmente, o vídeo é segmentado em quadros (passo 1). Em seguida (passo 2),

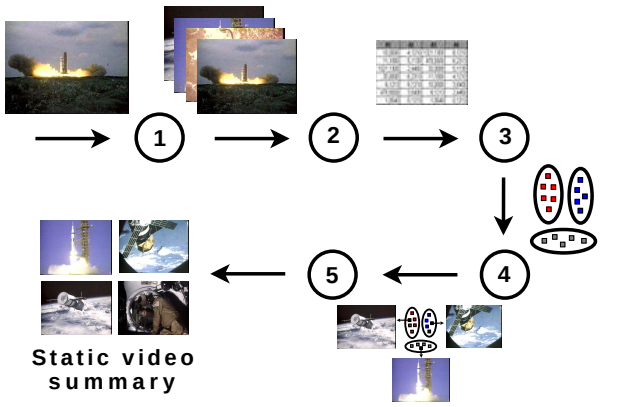


Figura 1. Diagrama do método proposto para a sumarização de vídeo.

são extraídas as características dos quadros. Nesta etapa, é utilizada uma sub-amostra dos quadros que compõem o vídeo. No passo 3, os quadros são agrupados através de um algoritmo de agrupamento. No passo 4, o quadro mais representativo (quadro-chave) de cada agrupamento é selecionado. Para gerar o resumo estático (passo 5), os quadros-chave semelhantes são eliminados com o propósito de manter a qualidade do resumo. E por fim, os quadros-chave são dispostos em ordem cronológica para facilitar o entendimento visual do resultado. Os passos da metodologia proposta são detalhados a seguir.

3.1. Segmentação do Vídeo

A segmentação temporal do vídeo é, geralmente, o primeiro passo para a sumarização de vídeos. Esta tem como objetivo separar o vídeo em seus componentes básicos (tomadas, quadros, por exemplo) [18]. Na literatura, a segmentação do vídeo em tomadas, mais conhecida como detecção de tomadas (*shot boundary detection*) [4], é largamente utilizada como etapa inicial na geração de resumos em diversos trabalhos, tais como [3, 6, 13, 15, 20, 21, 26, 32].

Para identificar os limites de uma tomada, é necessário detectar a transição entre duas tomadas consecutivas. Os tipos de transições são divididos em transições abruptas e transições graduais. Atualmente, a detecção de transições abruptas têm sido realizada com sucesso, diferentemente da detecção de transições graduais que continua sendo um problema, haja visto a quantidade de soluções propostas na literatura. A maioria das abordagens geralmente falham quando a ação acontece durante a transição gradual, o que pode resultar em resumos que não representam o conteúdo original do vídeo. Além disso, a abordagem baseada em tomadas pode apresentar redundâncias, uma vez que diferen-

tes tomadas podem conter conteúdo similar. Por exemplo, em vídeos de notícias, o apresentador pode aparacer várias vezes em diferentes tomadas e conseqüentemente quadros semelhantes podem compor o resumo.

Outro tipo de segmentação é a separação do vídeo em quadros. Neste tipo de segmentação, não há uma análise temporal do vídeo. Cada quadro é tratado separadamente, o vídeo é decomposto em um conjunto de imagens. Este tipo de segmentação é aplicada em [7, 12, 24, 30] e também é utilizada neste trabalho. Para a realização desta etapa foi utilizado o aplicativo FFmpeg[2].

3.2. Pré-amostragem

Um vídeo é composto por uma seqüência de quadros. Para assegurar que os seres humanos não percebam qualquer descontinuidade no fluxo de quadros, é necessária uma taxa de quadros por segundo de pelo menos 25 fps (*frame per second*) [14], isto é, 90.000 imagens em uma hora de vídeo. É intuitivamente óbvio que, para uma taxa de quadros de 25 fps, os 25 quadros mostrados a cada segundo contêm informação redundante. Diante disso, em vez de considerar todos os quadros do vídeo, o método proposto analisa apenas um subconjunto de quadros (uma pré-amostragem).

Pré-amostragem é uma técnica amplamente utilizada para reduzir o número de quadros do vídeo a serem analisados, como utilizada em [24], por exemplo. Conseqüentemente, quanto maior for a taxa de amostragem, menor será o tempo de execução para criar os resumos. No entanto, dependendo das características do vídeo, a qualidade dos resultados pode ser comprometida. Vídeos com tomadas muito longas tendem a ter vantagem com a técnica de pré-amostragem. Por outro lado, vídeos com tomadas muito curtas podem vir a sofrer com a não representação de parte importante de seu conteúdo. Esta relação de *perda de informação* versus *tamanho da tomada* está diretamente associada ao parâmetro utilizado para selecionar as amostras a serem consideradas durante a etapa de sumarização.

Neste trabalho foram experimentadas diferentes taxas de amostragem.

3.3. Extração de Características

Imagens podem ser descritas por características como cor, forma e textura. Diferentes imagens apresentam diferentes cores, formas e/ou texturas. Estas propriedades podem ser mensuradas e a medida é denominada *característica da imagem*. Estas características são normalmente descritas por um vetor de características, que é uma representação numérica de uma imagem. O vetor de características é um vetor d-dimensional contendo esses valores, que pode ser usado para discriminar as imagens entre si. Neste trabalho são utilizadas as características de cor da imagem. Para gerar o vetor de características são utilizadas ferramentas simples para análise de imagens: histograma de cor ou perfil de linha (vertical, horizontal e diagonal).

O histograma de cor representa a distribuição da freqüência de ocorrência dos valores cromáticos em uma imagem. Além de ser uma técnica computacionalmente simples, o histograma de cor é robusto a pequenas mudanças do movimento da câmera. Ele pode ser descrito computacionalmente através de uma estrutura de dados unidimensional. O tamanho desta estrutura deve ser igual ao número de cores possíveis no espaço de cor utilizado para representar a imagem. Considerando que imagens coloridas tipicamente têm as cores representadas em um espaço de cor com 256 níveis para cada canal de cor, uma redução se faz necessária para viabilizar a aplicação do algoritmo de agrupamento. Para isto, optou-se por fazer a quantização das cores utilizadas na representação das imagens durante a formação dos histogramas de cores. A quantidade de cores quantizadas para representar cada canal no espaço RGB é determinada pelo parâmetro *número de cores quantizadas*.

O perfil de linha representa os valores cromáticos presentes em uma determinada linha da imagem. Como uma linha da imagem não é suficiente para identificar a similaridade entre as imagens, é analisado mais de um perfil de linha para cada imagem. O número de perfis de linha analisado é determinado pelo parâmetro *intervalo entre perfis de linha*. Por exemplo, se o valor do parâmetro for igual a 10, então será gerado o perfil de linha da imagem a cada 10 linhas.

3.4. Técnica de Agrupamento

A técnica de agrupamento (*clustering*) consiste na classificação não-supervisionada de padrões em agrupamentos (*clusters*) [16]. Esta abordagem consiste na organização de um conjunto de padrões (usualmente representados na forma de vetores de características ou pontos em um espaço multidimensional) em agrupamentos, de acordo com alguma medida de similaridade. Intuitivamente, padrões pertencentes a um dado agrupamento devem ser mais "similares" entre si do que em relação a padrões pertencentes a outros agrupamentos. Neste trabalho foi utilizado o método de agrupamento *k-means* [23] e a distância euclidiana, como medida de similaridade. O *k-means* foi adotado por ser uma abordagem simples, bastante difundida e por fornecer bons resultados no processo de classificação não-supervisionada de dados [5].

O método *k-means* requer a definição prévia do número de agrupamentos (k). A idéia central desta técnica é a maximização da distância entre os agrupamentos, de mesmo

2 http://ffmpeg.mplayerhq.hu/

modo que minimiza as distâncias entre os elementos de cada agrupamento. O algoritmo consiste nos seguintes passos:

1. Padronize todos os dados, descrevendo cada variável (x_i) conforme a fórmula a seguir:

$$x_i = \frac{x_i - \mu}{\sigma}$$

2. Divida os casos em *k* agrupamentos;
3. Calcule o centróide de cada agrupamento, isto é, o ponto médio do agrupamento;
4. Para cada caso, calcule a distância euclidiana em relação ao centróide de cada agrupamento;
5. Transfira o caso para o agrupamento cuja distância ao centróide é mínima;
6. Repita os passos 3, 4 e 5 até que nenhum caso seja mais transferido.

Para cada agrupamento obtido, o quadro mais próximo do centróide é selecionado como quadro-chave.

3.5. Eliminação dos Quadros-chave Semelhantes

O objetivo desta etapa consiste em eliminar possíveis quadros-chave semelhantes. Para isto, os quadros-chave são comparados entre si de acordo com a característica extraída (histograma ou perfil de linha). A semelhança é baseada em um limiar δ, determinado através de experimentação visual. Caso o valor medido seja menor que δ, então o quadro-chave é eliminado. Em seguida, os quadros-chave resultantes são dispostos em ordem cronológica para facilitar o entendimento visual do resumo.

4. Resultados Experimentais

Os experimentos foram realizados nos vídeos extraídos do repositório *The Open Video Project* [1]. Para analisar o método proposto, uma amostra de 20 vídeos foi escolhida aleatoriamente. Todos os vídeos estão no formato MPEG1, com resolução 320 x 240 *pixels* a 30 quadros por segundo e categorizados como documentários. A duração total do conjunto de teste é aproximadamente 45 minutos, sendo que a duração de cada vídeo varia entre 1 e 4 minutos. Na Tabela 1 são listadas algumas informações dos vídeos usados. Para implementação computacional, foi utilizada a linguagem de programação C++. Os resultados foram conduzidos em uma máquina com processador Intel® Core™ Duo 1.83 GHz e 2GB de memória RAM.

Nome	#Quadros	Duração
video1	2.494	1:23
video2	2.775	1:32
video3	3.269	1:49
video4	3.302	1:50
video5	3.458	1:55
video6	3.534	1:58
video7	3.537	1:58
video8	3.609	2:01
video9	3.620	2:00
video10	3.630	2:01
video11	3.895	2:10
video12	4.267	2:22
video13	4.273	2:22
video14	4.306	2:23
video15	4.662	2:35
video16	5.874	3:16
video17	6.019	3:20
video18	6.099	3:23
video19	6.449	3:35
video20	6.902	3:50
Total	85.974	44:23

Tabela 1. Vídeos utilizados nos experimentos.

4.1. Análise da Taxa de Amostragem

Para analisar a taxa de amostragem, os testes foram realizados em oito vídeos escolhidos aleatoriamente da amostra de 20 vídeos. Para cada taxa utilizada, a qualidade do resumo e o tempo necessário para produzi-lo foram medidos com a finalidade de determinar quantos quadros do vídeo devem ser analisados. As taxas foram definidas de acordo com o gênero do vídeo utilizado neste trabalho (documentários), os quais são compostos por tomadas longas.

O tempo de execução para a geração dos resumos foi computado utilizando todos os quadros do vídeo, um quadro a cada 30, 45, 60, 75 e 90 quadros (veja a Tabela 2). A partir dos testes realizados, foi observado que a qualidade dos resumos produzidos não apresentou nenhuma diferença significativa, exceto para a taxa de um quadro a cada 90 quadros. Para comparar o desempenho do método proposto para as diferentes taxas de amostragem (um quadro a cada 75 quadros com as outras taxas), foi aplicado o teste-t (*student*) com 95% de confiança [17], o qual demonstrou que, em média, as alternativas são estatisticamente diferentes, exceto para as taxas um quadro a cada 60 quadros e um quadro a cada 75 quadros. Como esta taxa de amostragem apresentou um desempenho ligeiramente melhor, então os próximos experimentos foram realizados utilizando-a.

Vídeos	Tempo de Execução (s)					
	Todos	30	45	60	75	90
video2	59,11	2,07	1,43	1,12	0,90	0,79
video8	85,76	2,92	1,43	1,62	1,26	1,09
video9	87,34	2,95	2,00	1,55	1,27	1,13
video11	83,86	2,93	1,98	1,53	1,20	1,08
video12	96,39	3,15	2,17	1,68	1,28	1,21
video17	162,08	4,71	3,47	2,51	2,04	1,72
video18	160,53	4,87	3,44	2,58	2,09	1,81
video20	172,85	5,43	3,80	3,01	2,26	2,00
média	113,49	3,63	2,59	1,95	1,54	1,35

Tabela 2. Tempo de execução para a sumarização de vídeos utilizando todos os quadros do vídeo, um quadro a cada 30, 45, 60, 75 e 90 quadros.

4.2. Análise das Características e Avaliação dos Resultados através de Usuários

Para analisar as técnicas de extração de características descritas na Seção 3.3 foi aplicado um projeto fatorial 2^k [17], com $k = 2$ ou $k = 3$ se a técnica utilizada foi histograma ou perfil de linha, respectivamente. Para simplificar o modelo experimental, os fatores que afetam o desempenho do processo de sumarização de vídeo foram representados em dois níveis, como mostrado a seguir:

1. Número de agrupamentos: 15 ou 35 agrupamentos.
2. Número de cores quantizadas: 16 ou 256 cores.
3. Intervalo entre perfis de linha: 10 ou 40 linhas.

Para estes experimentos, são utilizados os mesmos vídeos do experimento anterior. Os resultados são mostrados na Tabela 3 e na Tabela 4, onde a configuração do 5º experimento da Tabela 4 apresentou o melhor desempenho para a sumarização. Com 95% de confiança, os resultados obtidos através da utilização do histograma e dos perfis de linha apresentaram diferenças estatisticamente significantes. No entanto, para o mesmo valor de confiança, não houve diferença entre os perfis de linha.

A	B	Tempo de Execução (s)
15	16	**1,50**
35	16	1,56
15	256	1,68
35	256	1,91

Tabela 3. Média do tempo de execução para a sumarização de vídeos utilizando histograma. *A* indica o número de agrupamentos e *B* o número de cores quantizadas.

A	B	C	Tempo de Execução (s)		
			Horizontal	Vertical	Diagonal
15	16	10	0,87	0,99	1,10
35	16	10	1,26	1,27	1,29
15	256	10	0,81	0,95	1,12
35	256	10	0,96	1,13	1,45
15	16	40	**0,65**	0,90	0,82
35	16	40	0,67	1,05	1,05
15	256	40	0,86	**0,72**	**0,79**
35	256	40	1,11	0,78	0,90

Tabela 4. Média do tempo de execução para a sumarização de vídeos utilizando perfis de linha. *A* indica o número de agrupamentos, *B* o número de cores quantizadas e *C* o intervalo entre os perfis de linha.

Entretanto, como a qualidade do resumo é mais importante do que o tempo necessário para produzi-lo, então os resumos gerados para os experimentos que apresentaram o melhor desempenho, utilizando histograma e perfil de linha, foram avaliados por 10 pessoas objetivando-se medir a qualidade destes. Antes de avaliar o resumo, as pessoas podem assistir o vídeo quantas vezes acharem necessário. Em seguida, de acordo com uma escala que varia de 1 a 5 (1 = ruim, 2 = pobre, 3 = satisfatório, 4 = bom, 5 = excelente), as pessoas devem responder a seguinte pergunta: qual a relevância de cada imagem (quadro-chave) do resumo de acordo com o conteúdo do vídeo? Na Tabela 5 é apresentada a média da opinião das pessoas, a qual mede a

Vídeos	Pontuação			
	Histog.	Perfil de Linha		
		Horiz.	Vert.	Diag.
video2	3,6	3,7	3,5	3,7
video8	2,9	3,4	3,1	2,8
video9	2,8	3,3	2,9	2,9
video11	3,3	3,7	3,5	3,3
video12	3,3	3,4	3,5	3,3
video17	3,7	3,4	3,3	3,1
video18	3,4	3,5	3,4	3,3
video20	3,3	3,3	3,2	3,1
média	3,3	**3,5**	3,3	3,2

Tabela 5. Resultado da avaliação dos usuários.

qualidade do resumo.

O resultado da avaliação dos usuários mostrou que o uso do perfil horizontal para a sumarização de vídeo fornece resultados com qualidade superior em relação às demais técnicas utilizadas. Assim, para a melhor configuração experimental obtida – 16 cores, intervalo de 40 linhas entre os perfis (ou seja, 6 perfis horizontais) e diferentes número de agrupamentos (15, 20, 25, 30 e 35 agrupamentos) – são gerados os resumos dos 20 vídeos do conjunto de teste (veja a Tabela 1). A média do tempo de execução (em segundos) do algoritmo para cada agrupamento foi 0,59, 0,64, 0,63, 0,64 e 0,65, respectivamente. Os resumos também foram avaliados pelos mesmos 10 usuários. O melhor resultado (25 agrupamentos) apresentou uma pontuação média de 4,3 e o pior resultado (15 agrupamentos) 3,6 pontos.

4.3. Comparação com os Resumos do Open Video

Para os 20 vídeos do conjunto de teste, foi realizada uma comparação[3] dos resumos produzidos neste trabalho e os resumos do Open Video. Para não favorecer/prejudicar o método proposto, o número de quadros-chave do resumo gerado foi baseado no número de quadros-chave dos resumos do Open Video (ou seja, se o resumo do Open Video contém cinco quadros-chave, então o número de quadros-chave para o método proposto também é fixado em cinco quadros-chave). Assim, o número máximo de quadros-chave será o mesmo para ambas abordagens.

Na Tabela 6 são apresentados a qualidade e o número de quadros-chave de cada resumo. De acordo com os resultados experimentais obtidos, os resumos produzidos pelo método proposto apresentaram qualidade superior para nove vídeos dentre os 20 vídeos; Cinco resumos obtiveram a mesma pontuação; E para seis vídeos, os resumos gerados pelo Open Video apresentaram qualidade superior. Pode-se notar ainda que o melhor resultado alcançado pelo método proposto apresentou uma pontuação igual a 4,4 em contraste ao melhor resultado do Open Video que foi igual a 4,0. As piores pontuações das duas abordagens foi igual a 3,3. Além disso, o método proposto apresentou cinco resumos com pontuação maior ou igual a 4,0, enquanto que o Open Video obteve apenas uma pontuação igual a 4,0.

Para ilustrar esta comparação são mostrados os resultados para três vídeos. Na Figura 2, o resumo gerado pelo método proposto apresentou qualidade superior em relação ao resumo do Open Video. Por outro lado, na Figura 3 o resumo do Open Video mostrou um resultado melhor. E, na Figura 4 os resumos apresentam a mesma pontuação.

3 http://www.dcc.ufmg.br/ sandra/videosummarization

Videos	Pontuação		#Quadros-chave	
	Open Video	Método Proposto	Open Video	Método Proposto
video1	4,0	4,4	20	10
video2	3,8	3,8	14	9
video3	3,5	3,5	18	10
video4	3,8	4,1	12	9
video5	3,9	3,5	7	7
video6	3,3	3,3	12	10
video7	3,0	3,6	12	8
video8	3,8	3,7	12	7
video9	3,4	3,3	6	6
video10	3,4	3,7	12	8
video11	3,8	3,8	29	15
video12	3,6	3,8	26	10
video13	3,7	4,0	8	6
video14	3,8	3,5	10	6
video15	3,7	3,6	12	10
video16	3,7	4,0	6	5
video17	3,7	4,0	19	9
video18	3,8	3,8	22	13
video19	3,3	3,6	13	8
video20	3,8	3,6	19	11

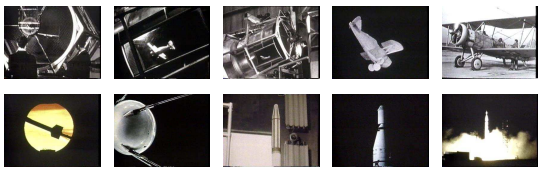
Tabela 6. Comparação da qualidade entre os resumos gerados pelo método proposto e pelo Open Video.

5. Conclusões e Trabalhos Futuros

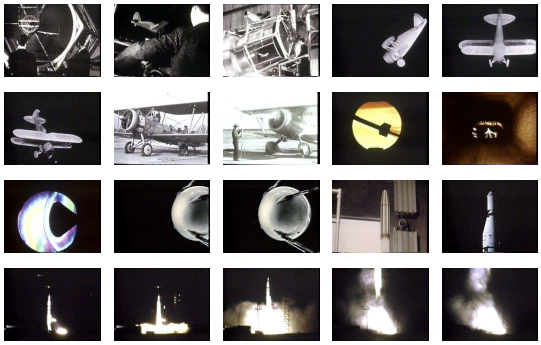
Neste trabalho foi proposta uma abordagem simples e eficiente para a sumarização de vídeo. Para produzir os resumos foi utilizada características de baixo-nível das imagens, extraídas através do histograma de cor e perfil de linha. O método também utiliza o algoritmo de agrupamento *k-means*. Os resultados gerados apresentaram qualidade com baixo consumo de tempo. Os resumos produzidos pelo método proposto foram comparados com os resumos do Open Video, sendo que na maioria dos casos, os resumos do método proposto apresentaram qualidade superior em relação aos resumos do Open Video.

Com a elaboração deste trabalho observou-se a necessidade da melhoria de algumas etapas e a inclusão de outras. O desenvolvimento destas idéias provavelmente contribuirá positivamente para o processo de sumarização proposto neste trabalho.

- **Pré-processamento dos quadros**: Eliminar quadros inexpressivos, como um quadro completamente preto (ou outra cor).
- **Espaço de cor**: Utilizar um espaço mais robusto à



(a) Método proposto



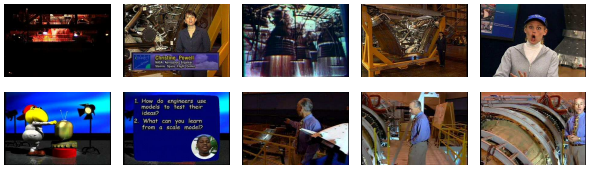
(b) Open Video

Figura 2. Resumos produzidos para o *video1*.

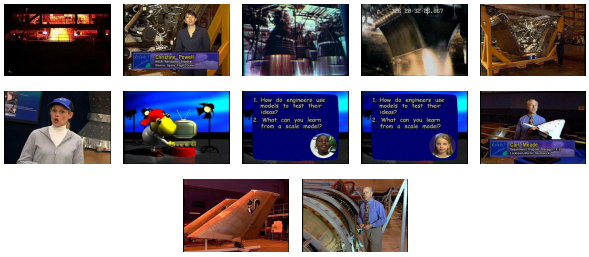
(a) Método proposto

(b) Open Video

Figura 3. Resumos produzidos para o *video5*.



(a) Método proposto



(b) Open Video

Figura 4. Resumos produzidos para o *video6*.

variação de cor, como por exemplo o HSV, aplicado em [7, 24, 32].

- **Características**: Usar outras características, como por exemplo forma, textura.
- **Número de agrupamentos**: Definir uma métrica para estimar o número de agrupamentos "ideal" para cada vídeo, conforme suas características.
- **Refinamento dos agrupamentos**: Selecionar os agrupamentos significativos. Por exemplo, em [31] são selecionados apenas os agrupamentos que são grandes o suficiente para ser um agrupamento-chave (*key cluster*). Um agrupamento é grande o suficiente se o seu tamanho é maior que N/G, onde N representa o número de quadros do agrupamento e G o número de agrupamentos.
- **Diferentes gêneros de vídeos**: Aplicar o método proposto para diferentes gêneros de vídeos.

Referências

[1] The Open Video Project. http://www.open-video.org.

[2] J. Ćalić, D. P. Gibson, and N. W. Campbell. Efficient layout of comic-like video summaries. *IEEE Transactions on Circuits and Systems and Video Technology*, 17(7):931–936, 2007.

[3] I.-C. Chang and K.-Y. Chen. Content-selection based video summarization. *Digest of Technical Papers International Conference on Consumer Electronics (ICCE)*, pages 1–2, 2007.

[4] C. Cotsaces, N. Nikolaidis, and L. Pitas. Video shot detection and condensed representation: A review. *IEEE Signal Processing Magazine*, 23(2):28–37, 2006.

[5] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*, chapter Unsupervised Learning and Clustering, page 654. Springer-Verlag New York, Inc., 2001.

[6] F. Dufaux. Key frame selection to represent a video. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, volume 2, pages 275–278, 2000.

[7] M. Furini, F. Geraci, M. Montangero, and M. Pellegrini. Visto: visual storyboard for web video browsing. In *Proce-*

edings of the ACM International Conference on Image and Video Retrieval (CIVR), pages 635–642, 2007.

[8] M. Furini, F. Geraci, M. Montangero, and M. Pellegrini. On using clustering algorithms to produce video abstracts for the web scenario. In *Proceedings of the IEEE Consumer Communication and Networking (CCNC)*, pages 1112–1116. IEEE Communication Society, January 2008.

[9] F. Geraci, M. Pellegrini, P. Pisati, and F. Sebastiani. A scalable algorithm for high-quality clustering of web snippets. In *Proceedings of the ACM Symposium on Applied Computing (SAC)*, pages 1058 –1062, 2006.

[10] A. Girgensohn. A fast layout algorithm for visual video summaries. In *Proceedings of the International Conference on Multimedia and Expo (ICME)*, pages 77–80, Washington, DC, USA, 2003. IEEE Computer Society.

[11] A. Girgensohn, J. Boreczky, and L. Wilcox. Keyframe-based user interfaces for digital video. *IEEE Computer*, 34(9):61–67, 2001.

[12] Y. Gong and X. Liu. Video summarization using singular value decomposition. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2174–2180, Los Alamitos, CA, USA, 2000. IEEE Computer Society.

[13] Y. Hadi, F. Essannouni, and R. O. H. Thami. Video summarization by k-medoid clustering. In *Proceedings of the ACM Symposium on Applied Computing (SAC)*, pages 1400–1401, 2006.

[14] R. I. Hammoud. *Interactive Video Algorithms and Technologies*. Springer Berlin Heidelberg, 2006.

[15] A. Hanjalic and H. Zhang. An integrated scheme for automated video abstraction based on unsupervised cluster-validity analysis. *IEEE Transactions Circuits & Systems for Video Technology*, 9(8):1280–1289, 1999.

[16] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: A review. *ACM Computer Surverys*, 31(3):264–323, 1999.

[17] R. Jain. *The Art of Computer Systems Performance Analysis*. John Wiley and Sons, Inc., 1992.

[18] I. Koprinska and S. Carrato. Temporal video segmentation: a survey. *Signal Processing: Image Communication*, 16(5):477–500, 2001.

[19] Y. Li, T. Zhang, and D. Tretter. An overview of video abstraction techniques. Technical report, HP Laboratory, HP-2001-191, July 2001.

[20] Z. Li, K. Katsaggelos, and B. Gandhi. Temporal rate-distortion based optimal video summary generation. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, pages 693–696, Washington, DC, USA, 2003.

[21] Z. Li, G. M. Schuster, and A. K. Katsaggelos. Minmax optimal video summarization. *IEEE Transactions on Circuits and Systems and Video Technology*, 15(10):1245–1256, 2005.

[22] X. Liu, T. Mei, X.-S. Hua, B. Yang, and H.-Q. Zhou. Video collage. In *Proceedings of the ACM International Conference on Multimedia*, pages 461–462, 2007.

[23] J. B. MacQueen. Some methods for classification and analysis of multivariate observations. In L. M. L. Cam and J. Neyman, editors, *Proceedings of The Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press, 1967.

[24] P. Mundur, Y. Rao, and Y. Yesha. Keyframe-based video summarization using Delaunay clustering. *International Journal on Digital Libraries*, 6(2):219–232, 2006.

[25] S. Pfeiffer, R. Lienhart, S. Fischer, and W. Effelsberg. Abstracting digital movies automatically. Technical report, University of Mannheim, 1996.

[26] J. Rong, W. Jin, and L. Wu. Key frame extraction using inter-shot information. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, pages 571–574, 2004.

[27] Y. Tonomura, A. Akutsu, K. Otsuji, and T. Sadakata. Videomap and videospaceicon: Tools for anatomizing video content. In *Proceedings of the INTERCHI Conference on Human Factors in Computing Systems*, pages 131–136, Amsterdam, The Netherlands, 1993. IOS Press.

[28] B. T. Truong and S. Venkatesh. Video abstraction: A systematic review and classification. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 3(1), 2007.

[29] H. Ueda, T. Miyatake, and S. Yoshizawa. Impact: an interactive natural-motion-picture dedicated multimedia authoring system. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 343 –350. ACM Press, 1991.

[30] I. Yahiaoui, B. Mérialdo, and B. Huet. Automatic video summarization. In *Multimedia Content-Based Indexing and Retrieval (MCBIR)*, 2001.

[31] H. J. Zhang, J. Wu, D. Zhong, and S. W. Smoliar. An integrated system for content-based video retrieval and browsing. *Pattern Recognition*, 30(4):643–658, 1997.

[32] Y. Zhuang, Y. Rui, T. S. Huang, and S. Mehrotra. Adaptive key frame extraction using unsupervised clustering. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, volume 1, pages 866–870, 1998.