



Exacta

ISSN: 1678-5428

exacta@uninove.br

Universidade Nove de Julho

Brasil

Riyuiti Hirakawa, Andre; Miyadaira Amâncio, Silvio; Saraiva, Antonio Mauro; Cugnasca, Carlos  
Eduardo; Correa, Pedro Luiz

ViBee - algoritmo de segmentação aplicado na compressão de vídeos em Weblabs de abelhas

Exacta, vol. 7, núm. 1, enero-marzo, 2009, pp. 99-107

Universidade Nove de Julho

São Paulo, Brasil

Disponível em: <http://www.redalyc.org/articulo.oa?id=81012760010>

- ▶ Como citar este artigo
- ▶ Número completo
- ▶ Mais artigos
- ▶ Home da revista no Redalyc

redalyc.org

Sistema de Informação Científica

Rede de Revistas Científicas da América Latina, Caribe , Espanha e Portugal  
Projeto acadêmico sem fins lucrativos desenvolvido no âmbito da iniciativa Acesso Aberto

# ViBee - algoritmo de segmentação aplicado na compressão de vídeos em Weblabs de abelhas

Andre Riyuiti Hirakawa

Professor Associado  
Departamento de Engenharia de Computação e Sistemas Digitais – Escola Politécnica da USP. São Paulo – SP [Brasil]  
ahiraka@gmail.com

Silvio Miyadaira Amâncio

Pesquisador Mestre  
Departamento de Engenharia de Computação e Sistemas Digitais – Escola Politécnica da USP. São Paulo – SP [Brasil]  
amanciosilvio@yahoo.com.br

Antonio Mauro Saraiva

Professor Titular  
Departamento de Engenharia de Computação e Sistemas Digitais – Escola Politécnica da USP. São Paulo – SP [Brasil]  
amsaraiv@usp.br

Carlos Eduardo Cugnasca

Professor Associado  
Departamento de Engenharia de Computação e Sistemas Digitais – Escola Politécnica da USP. São Paulo – SP [Brasil]  
ccugnas@gmail.com

Pedro Luiz Correa

Professor Doutor  
Departamento de Engenharia de Computação e Sistemas Digitais – Escola Politécnica da USP. São Paulo – SP [Brasil]  
pedro.correa@usp.br

A crescente disponibilização de serviços na internet tem viabilizado o surgimento de laboratórios a distância, os Weblabs. Nesse contexto, a transmissão de vídeo é primordial para uma efetiva visualização e interação remota. Entretanto, as restrições de banda dos diversos segmentos que integram a internet não permitem que vídeos de alta qualidade e em tempo real possam ser transmitidos, requisitos essenciais para que exista efetiva monitoração e atuação a distância. Considerando-se um Weblab, chamado de Webbee, por meio do qual se pretende monitorar o comportamento das abelhas, esses requisitos se tornam ainda mais importantes. Codificadores comerciais (MPEG4 e WMV) aplicados em vídeo de abelhas apresentam grande redução no tamanho do arquivo, mas também de qualidade, o que inviabiliza suas aplicações para o Webbee. Este trabalho apresenta a proposta de um novo algoritmo para codificação de vídeo de abelhas e redução da taxa de dados, mantendo a qualidade da informação. São utilizados conceitos de segmentação de vídeo por identificação, reconhecimento e rastreamento de objetos. Os resultados dos testes realizados mostram a adequação da proposta com redução de dados equivalente aos dos codificadores comerciais e com índice de similaridade de 93%.

**Palavras-chave:** Abelhas. Compressão. Segmentação. Weblab. Vídeo.



## 1 Introdução

### 1.1 O Projeto ViNCES

O Webbee, instalado no laboratório de abelhas do Instituto de Biociências da Universidade de São Paulo (IBUSP), é um Weblab pertencente ao projeto Virtual Network Center of Ecosystem Services (ViNCES) (AMANCIO et al., 2006) que possui bibliotecas e laboratórios virtuais e visa prover um ambiente, com base na Web, para pesquisa e educação sobre abelhas nativas. Além disso, facilita a realização de experimentos remotos e aumenta o conhecimento da sociedade sobre a importância das abelhas nativas. No Webbee são realizadas a coleta e transmissão de áudio, vídeo e colhidas informações sobre as condições de temperatura e umidade obtidas no interior de uma colmeia de abelhas da espécie mandassaia (*Melipona quadrifasciata athidioides*). Os dados são utilizados para estudos relacionados ao seu comportamento e disponibilizados no laboratório chamado BBBee. Atualmente, é realizada a transmissão de vídeo da área de recrutamento da colmeia por meio de uma rede Ethernet 10/100 convencional. Em razão das restrições da rede, as taxas de transmissão de vídeo foram limitadas a até 512 kbps (kilobits por segundo). O vídeo transmitido sofre uma compressão que reduz o volume de dados necessários a sua transmissão; no entanto, acarreta perdas de qualidade e provoca distorção na imagem apresentada. A estrutura do sistema de captura, codificação e transmissão de vídeo é composta por uma câmera analógica Sony modelo CCD-PC1, de resolução padrão (352×240 pixels), instalada em um computador Pentium IV 3.0 GHz, com 1Gb RAM, executando o Windows Media Services e Windows Media Encoder. (AMANCIO et al.; 2006)

Atualmente, a baixa capacidade de transmissão e a insuficiente taxa de compressão provida pelos Codificadores/Decodificadores (CODECs) di-

ficultam a realização de estudos mais detalhados, uma vez que, para obter baixas taxas de transmissão, é preciso que seja realizada compressão de vídeo. Em muitos casos, isso dificulta a visualização do vídeo da colmeia e, principalmente, da movimentação das abelhas. Além da deterioração da qualidade das imagens, é comum a ocorrência de grandes atrasos na entrega do vídeo, ocasionados, principalmente, pelos processos de compressão e streaming (ZHANG; e LU, 2001).

Este trabalho propõe a utilização de técnicas de processamento de imagens que ajudem a identificar e rastrear objetos de vídeo automaticamente, gerando um algoritmo de compressão com base na segmentação de objetos que será aplicado aos vídeos do interior das colmeias de abelhas. As técnicas de segmentação de vídeo possuem atualmente diversas aplicações, como o uso em aplicações de vigilância, para melhoria de compressão e qualidade, e em indexação de conteúdos (ZHANG; LU, 2001; VIBHA et al; 2008), permitindo enriquecer e facilitar buscas por conteúdos, automaticamente ou com ajuda do usuário (EMNA, 2008). Entre as características específicas dos vídeos obtidos nessas colmeias, podemos citar ausência de movimentação da câmera, cor restrita a tons de vermelho (vídeos monocromáticos) e ocorrência de períodos e localidades sem movimentação.

Neste estudo, busca-se utilizar a segmentação do vídeo para melhoria de sua qualidade, principalmente nas localidades desejadas, ou seja, filmagem das abelhas em movimento.

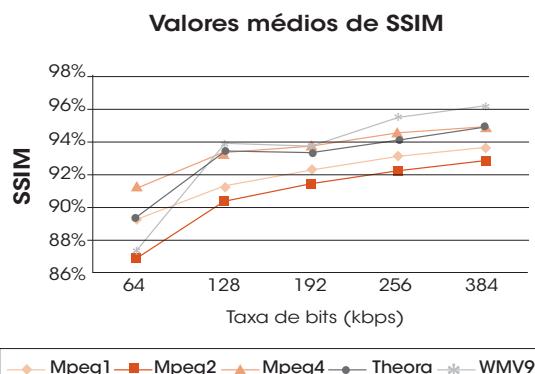
## 2 Compressão de vídeo

Entre os diversos padrões de vídeo existentes – ou CODECs –, é possível citar como os mais conhecidos e utilizados, os padrões MPEG (PURI; CHEN, 2000), WMV (LOOMIS; WASSON, 2007) e Audio and Video Interleave (AVI) (ROBBINS, 2006).

Para avaliar a qualidade da imagem após o processo de codificação e decodificação são utilizadas métricas objetivas, com diferente complexidade e impacto, sobre o Sistema Visual Humano (HVS). Entre elas está a Similaridade Estrutural (SSIM), adotada para avaliar a qualidade do vídeo resultante, que separa as alterações de contraste e luminância de distorções estruturais no espaço. Estudos comparativos sobre distorções demonstram a sua superioridade em relação aos métodos mais tradicionais, como a Relação de Sinal/Ruído de Pico (PSNR) e, o Erro Médio Quadrático (MSE) (WANG et al; 2003).

Quando aplicados a vídeos para obter altas taxas de compressão, os CODECs comercialmente disponíveis causam grandes reduções na SSIM, como apresentado na Figura 1, o que pode dificultar a visualização de detalhes das abelhas.

Na aplicação ao weblab de abelhas, efeitos de *blurring* (borramento causado nas bordas da imagem quando ocorre compressão excessiva) e *blocking* (bordas de blocos com grande degrau de transição em contraposição às naturais, que são suaves – resultado da quantização dos coeficientes da Transformada Discreta de Cosenos – DCT) dificultam ou impedem a correta identificação, como apresentado na Figura 2.



**Figura 1:** Valores de SSIM obtidos de diferentes codecs em um vídeo de resolução de 352 x 240 pixels [1]

Fonte: AMANCIO et al, 2006.



(a)



(b)

**Figura 2:** Comparação entre o vídeo sem compressão (a) e após compressão com o WMV9 a 92 kbps (b)

Fonte: AMANCIO et al, 2006.

### 3 Algoritmo de compressão desenvolvido

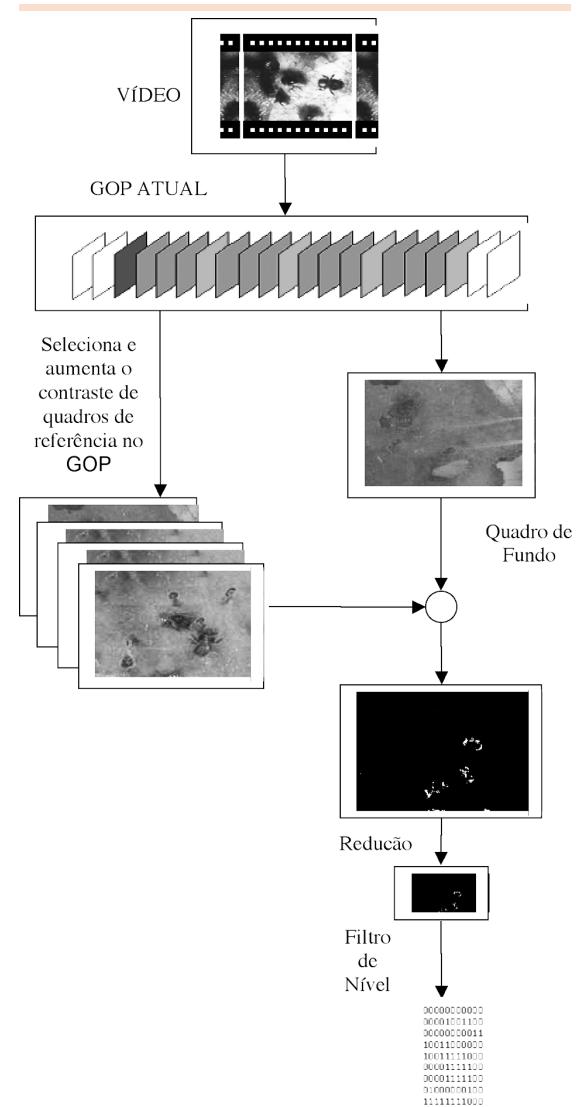
Nos CODECS comercialmente disponíveis, a taxa de bits é mantida em torno de um valor padrão de acordo com a qualidade desejada, mesmo quando não há movimentação observada nos vídeos. Considerando-se as características dos vídeos sem compressão obtidos do BBBee, a cena é definida como sendo composta basicamente por um plano de fundo estático e pelos objetos em movimento (abelhas). O principal fator que diferencia o fundo da colmeia é a ausência de movimentos



que podem ser detectados pela aplicação de fluxo óptico ou pela subtração entre quadros separados temporalmente (GIUSTO et al; 2002). Um dos algoritmos simples de segmentação rápida conhecidos, o Block Matching Algorithm (BMA), subtrai dois quadros consecutivos e os divide em blocos de tamanho  $N \times N$  ( $N =$ número de pixels – elementos que compõem as imagens), sendo o bloco de um quadro atual buscado no quadro anterior por meio de cálculo de média de diferença absoluta, entretanto tal técnica exige grande processamento (ZHU; LI, 2002).

O algoritmo proposto (ViBee) foi elaborado com base no processo de compressão do MPEG, utilizando compressão intra (compressão sem referência a outras imagens) e inter (compressão com referência a outras imagens). Para um grupo de quadros é armazenada uma imagem de referência (quadro de fundo), sobre a qual é aplicada compressão intra, no formato Joint Photographic Experts Group (JPEG). Com essa imagem são realizadas operações de subtração com os quadros posteriores daquele Grupo de Imagens (GOP) para verificar em quais pontos houve movimento (valores de diferença acima de um threshold – nível mínimo a partir do qual se considera que houve movimento – estabelecido). O filtro de nível considera que valores acima desse threshold apresentam alterações suficientes para gravar o trecho naquela localidade. Os locais marcados são gravados como vídeos e, posteriormente, para cada um dos trechos gravados é realizada a compressão MPEG4, por meio da utilização somente de quadros I (de maior qualidade), para reduzir a quantidade de dados necessária (Figura 3). A informação de posicionamento de cada objeto é registrada para permitir a posterior reconstrução do vídeo.

As imagens do GOP são convertidas em tonalidades de cinza de 8 bits (256 níveis) para que seja realizada a subtração entre quadros a partir da luminosidade das imagens (eq. 1). Essa conver-



**Figura 3: Estrutura do algoritmo de compressão desenvolvido**

Fonte: AMANCIO, 2008

são não acarreta perdas de informação, uma vez que o vídeo original contém somente imagens em tonalidades de vermelho.

$$\text{QuadroC} = \left( \frac{\text{QuadroR} + \text{QuadroG} + \text{QuadroB}}{3} \right) \quad (1)$$

A subtração entre dois quadros é realizada ponto a ponto (eq. 2). Como resultado, obtém-se

um quadro de diferenças também em tonalidades de cinza.

$$\text{Subtração}_{\text{Quadro}_i} = |(x,y)_{\text{Quadro}_i} - (x,y)_{\text{QuadroFundo}}| \quad (2)$$

De forma genérica, a operação de subtração sobre o quadro pode ser dada pela equação 3.

$$\text{Subtração}_{\text{GOP}} = \sum_{i=1}^n \text{Subtração}_{\text{Quadro}_i}$$

(3)

Para que sejam mapeados os blocos correspondentes, esse quadro de diferenças é reduzido de acordo com o tamanho dos blocos (isto é, caso o bloco seja do tamanho 8x8, a imagem é reduzida oito vezes na dimensão x, e oito, na y), de forma que cada ponto da imagem represente um bloco (eq. 4).

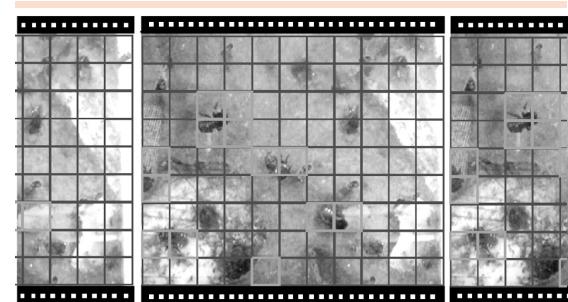
$$\text{Média}_{\text{Bloco}(i,j)} = \frac{\sum_{i=1, j=1}^{\dim x, \dim y} |(x_i, y_j)|}{\dim x \cdot \dim y} \quad (4)$$

Na qual:  $\dim x$  e  $\dim y$  são as dimensões dos blocos x e y, respectivamente

Ao resultado da redução é aplicado o filtro de nível que define a presença ou ausência de movimento em cada bloco (valores acima ou abaixo do nível, respectivamente). O valor do filtro é obtido experimentalmente, podendo variar de 0 a 255, e será mais próximo de 0 se houver maior quantidade de pontos de baixa intensidade em cada bloco subtraído, porém esse valor deve ser suficientemente alto para eliminar ruídos do vídeo (eq. 5). Como resultado são geradas matrizes de blocos com movimento na cor vermelha (Figura 4).

$$\text{Quadro}_{ij} C = (\text{Quadro}_{ij} - \min C) \cdot \left( \frac{255}{\max C - \min C} \right)$$

(5)



**Figura 4: Pontos marcados como movimento e mapeamento aos quadros do vídeo**

Fonte: AMANCIO, 2008.

Na qual:  $i$  = valor que varia entre 1 a (resolução horizontal do quadro original) / N;

$j$  = 1 a (resolução vertical do quadro original) / N;  $\min C$  = valor de intensidade mínimo no quadro e  $\max C$  = valor de intensidade máximo no quadro.

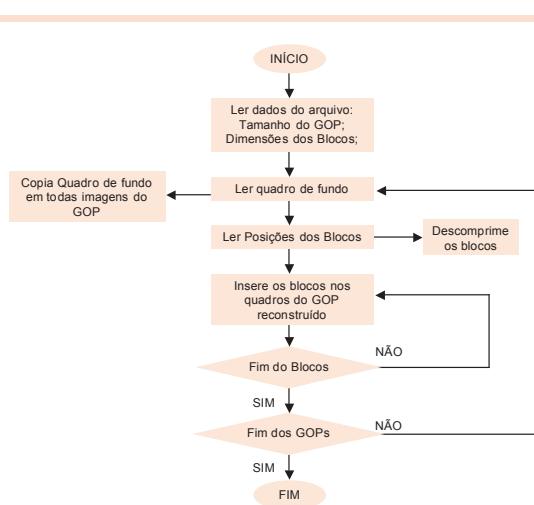
O formato geral do arquivo de informações de um vídeo comprimido em ViBee é composto de: quantidade de quadros; tamanho do GOP; tamanho do bloco; valor do nível; matrizes.

## 4 Algoritmo de reconstrução

No processo de reconstrução do vídeo (cujo algoritmo é representado na Figura 5), para cada quadro a ser reconstruído, as informações de posição e existência de trechos de vídeo e intervalos entre quadros intra definem quais blocos devem ser descomprimidos por meio de um decodificador MPEG4. Para os demais blocos (em que não houve movimentação), a imagem é gerada do quadro de fundo. As imagens são inseridas quadro a quadro, no vídeo reconstruído, sem compressão e reconstituindo o vídeo original.

## 5 Testes e resultados

O algoritmo ViBee foi testado sob diversas situações para avaliar seu desempenho com rela-



**Figura 5: Algoritmo de reconstrução**

Fonte: AMANCIO, 2008.

ção à qualidade e quantidade de dados gerados. A Tabela 1 apresenta algumas estimativas de taxas e resultados obtidos da prática, ao comprimir diversos trechos de vídeo, variando-se o valor do threshold. Em cada linha, identifica-se um trecho distinto, contendo quantidades diferentes de abelhas em movimento (ex.: na terceira linha, há cerca de 4 a 5 abelhas em movimento, com expectativa de gerar uma quantidade média de 8 a 15 blocos com movimento, e taxa resultante entre 131 e 215 kbps, em que se verificaram, nos testes, aproximadamente 11 blocos com movimento, produzindo uma taxa média de 200 kbps). Os valores de quantidades médias de blocos e taxas de

bits foram colhidos por meio de simulações que variaram o valor threshold (entre 50 e 65).

No teste de quantidade de dados *versus* SSIM por alteração de nível, o vídeo original, de resolução de 352x288 pixels, monocromático, com duração de 40s (ou 1200 quadros), quando não está comprimido, tem tamanho de 356,7MB. Foram alterados parâmetros como distância entre quadros intra e threshold e o tamanho dos blocos. Os resultados dos testes são mostrados nas Figuras 6 e 7, em que se pode observar menor redução do tamanho dos arquivos e a melhoria da qualidade medida pela métrica SSIM ao diminuir o threshold.

Os valores de threshold dos testes (50, 55, 60 e 65 da Figura 6) foram selecionados por meio de testes de eficiência (menores quantidades de dados *versus* maior SSIM). Maiores valores de nível geram menos dados; em contrapartida, a qualidade do vídeo reconstruído passa a ser inferior. Pode-se constatar que o bitrate resultante é basicamente definido pelo threshold (o bitrate aumenta significativamente ao reduzir o seu valor), sendo a qualidade obtida (SSIM) maior para menores níveis de threshold e GOPs com menores quantidades de quadros (Tabela 2).

Alterando-se o tamanho do bloco também ocorrem grandes variações de bitrates (Tabela 3), tendo, entretanto, tais mudanças pouco impacto na melhoria no SSIM.

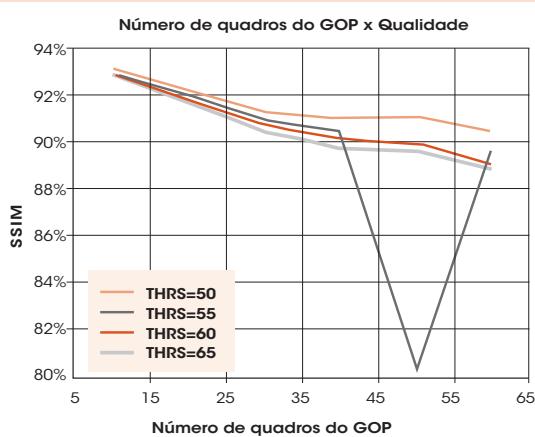
**Tabela 1: Estimativa versus resultados de quantidades de dados obtidas nas simulações para um trecho de 1 s de vídeo de resolução de 352x288 pels**

Quantidade de abelhas em movimento	Quantidade média de blocos esperada	Taxa de bits esperada (kbps)	Qt de blocos verificada (desvio padrão)	Taxa de bits obtida (kbps)
0	0	35	0 (0)*	37 (0)
2-3	4 ~ 9	83 ~ 143	6,6 (0,7)	139 (12,3)
4-5	8 ~ 15	131 ~ 215	10,9 (2,09)	201,5 (18,7)
5-6	10 ~ 12	155 ~ 179	19,61 (2,95)	344,2 (33,5)
Em todo o quadro	99	1223	99 (0) **	1380 (0)

\* simulado alterando o valor do threshold para 255;

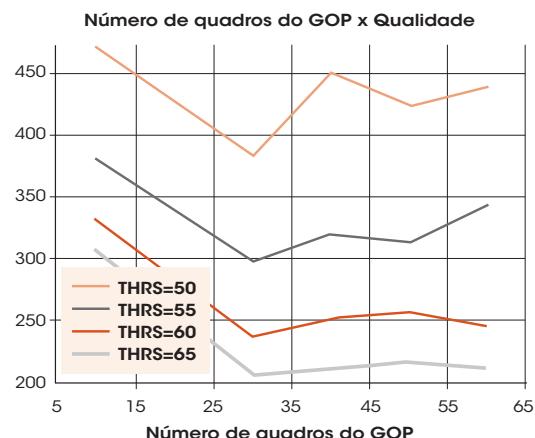
\*\* simulado alterando o valor do threshold para 0.

Fonte: AMANCIO, 2008.



**Figura 6: Número de quadros do GOP versus SSIM**

Fonte: AMANCIO, 2008.



**Figura 7: Número de quadros do GOP versus Bitrate**

Fonte: AMANCIO, 2008.

Na Tabela 4, os resultados do ViBee utilizando os critérios de qualidade (SSIM) e a quantidade de dados para diferentes trechos do mesmo vídeo, que apresentam diferentes quantidades de abelhas em movimento, são comparados com o padrão MPEG4,. As diferentes taxas de bits desse padrão foram geradas, configurando-se o CODEC para tal. Podemos constatar que o aumento da quantidade de abelhas em movimento, presentes no vídeo, não

**Tabela 2: Alteração de quadros I por segundo versus Bitrate e SSIM**

Quantidade de abelhas no quadro	Bitrate (kbps)	SSIM	Duração do GOP (quadros)
2 a 3 abelhas	156	93,07%	16
	137	92,93%	20
	131	92,63%	25
	130	92,69%	30
4 a 5 abelhas	223	92,82%	16
	209	92,56%	20
	192	91,88%	25
	180	92,45%	30
5 a 6 abelhas	385	92,57%	16
	352	92,58%	20
	306	92,34%	25
	331	92,42%	30

Fonte: AMANCIO, 2008.

**Tabela 3: Alteração do tamanho do bloco versus Bitrate e SSIM**

Quantidade de abelhas no quadro	Bitrate (kbps)	SSIM	Tamanho do quadro (pels)
2 a 3 abelhas	1109	92,74%	8
	295	92,68%	16
	131	92,63%	32
4 a 5 abelhas	1101	92,16%	8
	573	92,34%	16
	192	91,88%	32
5 a 6 abelhas	6652	92,35%	8
	1166	92,61%	16
	306	92,34%	32

Fonte: AMANCIO, 2008.

interfere no tamanho final do padrão MPEG4, mas aumenta a quantidade de dados no ViBee desenvolvido, conforme esperado. Por outro lado, essa quantidade causa alterações na SSIM. Por exemplo, para um vídeo de tamanho de 1,7 MB, após compressão com o MPEG4, a SSIM verificada para 4 ou 5 abelhas em movimento atingiu 91,31% contra uma SSIM de 92,22% para o mesmo vídeo comprimido com o ViBee, que gerou cerca de 1,6MB (milhões de bytes), com uma quantidade maior de abelhas em movimento (entre 5 e 6). Um vídeo do mesmo tamanho no MPEG4 apresentou SSIM de 91,26% no MPEG4 e ocasionou aumento para cerca de 2,8MB no ViBee, com uma SSIM de 92,34%.



**Tabela 4: Compressão com o padrão MPEG4 versus ViBee, em resolução de 352x288 pixels**

Qt de abelhas	Tamanho (MB)	MPEG 4					ViBee	
		SSIM	Tamanho (MB)	SSIM	Tamanho (MB)	SSIM	Tamanho (MB)	SSIM
2 a 3	1,4	89,92%	1,7	91,57%	2,2	93,59%	1,106	92,63%
4 a 5	1,4	89,72%	1,7	91,31%	2,2	93,44%	1,595	92,22%
5 a 6	1,4	89,71%	1,7	91,26%	2,2	93,43%	2,776	92,34%

Fonte: AMANCIO, 2008.

## 5 Considerações finais

O algoritmo ViBee utilizou as técnicas de segmentação de vídeo, com base na subtração de quadros, para realizar a compressão dos vídeos do BBBee, tendo sido desenvolvidos dois sistemas complementares para sua compressão e descompressão, respectivamente.

Os resultados de métricas de avaliação da qualidade de vídeo, obtidos da realização de testes em diferentes situações de movimentação de objetos no vídeo, permitiram avaliar e apontar os pontos fracos e fortes da proposta.

As características de qualidade e bitrates observadas são inerentes à forma de implementação do algoritmo, em que se objetivou obter maior qualidade nas localidades específicas que continham movimento de abelhas, porém, conforme o esperado, em algumas situações (como nos casos em que há movimentação em grande parte do vídeo), as vantagens do sistema são perdidas, e se constatam menores taxas de compressão ante resultados de qualidade menos expressivos (ex. Tabela 4).

Os principais efeitos verificados sobre a qualidade do vídeo e quantidade de dados gerados foram obtidos por meio da variação de parâmetros de entrada do sistema: o nível de detecção de movimento (threshold) e o tamanho do GOP. Pode-se constatar, ainda, que identificações de movimento incorretas (Figura 6, para threshold=55, a 50 quadros) podem causar grandes variações nas taxas de bits, deteriorando a funcionalidade do CODEC. Nos resultados

apresentados na Tabela 4, o threshold de valor 52 foi alcançado por intermédio de diversos testes, indicando que tal procedimento pode ser melhorado com a aplicação de alguma forma automática de definição de threshold.

Apesar de haver sido testada somente a técnica de segmentação de objetos por meio da diferença entre quadros, na maioria dos casos em que havia pouca movimentação no interior da colmeia, a qualidade do vídeo medida foi superior à dos padrões disponíveis, e o algoritmo, mantido simples.

### ViBee – segmentation algorithm applied to reduce video bitrates in bee weblabs

Services provided by Internet allow the creation of new distance applications, as remote laboratories (Weblabs). In this context, the broadcasting of video and images are essential for the success and effectiveness of the experiments. However, the bandwidth of the communication channel does not allow transmission of high-definition video in real-time, which are basic requirements for remote monitoring and actuation activities. Considering a Weblab, also called Webbee, through which it is intended to analyze the behavior of bees, these requirements become more important. Commercial videos, CODEC (MPEG4 and WMV), applied to videos of bees reduce the bitrates, as well as the quality, what makes impracticable the applications for the Webbee. Therefore, this paper presents the proposal of a new algorithm for codification of video of bees and the reduction of bitrates, maintaining the quality of the information. The algorithm includes the video

segmentation process by identification, recognition and tracking of objects. The test results show an adaptation of the proposal with reduction of data equivalent to the commercial CODECs, with 93% of similarity.

**Key words:** Bees. Compression. Segmentation. Video. Weblab.

## Referências

AMANCIO, S. M.; HIRAKAWA, A. R.; JUNIOR, E. L.; HILARIO, S. D.; KLEINERT, A. M. P.; FONSECA, V.L.I.; GIANNINI, T. C.; Study on real time video transmission for aid remote research of bees. Proceedings of the computers in agriculture and natural resources, 4th World Congress Conference, 24-26, jul. 2006.

\_\_\_\_\_. Redução da Taxa de Transmissão de Vídeo Digital por meio de Segmentação de Objetos Aplicado ao Estudo de Abelhas . Dissertação (Mestrado) – Escola Politécnica da USP, São Paulo, 2008.

ZHANG, D.; LU, G., Segmentation of Moving Objects in Image Sequence. A review. gippsland school of computing and information technology, Monash University, Churchill, Vic 3842, Australia, 2001.

VIBHA L.; HEGDE, C.; SHENOY, P. D.; VENUGOPAL K. R.; PATNAIK L. M.; Dynamic object detection, tracking and counting in video streams for multimedia mining, IAEENG International Journal of Computer Science, ago. 2008. Disponível em: [http://www.iaeng.org/IJCS/issues\\_v35/issue\\_3/IJCS\\_35\\_3\\_16.pdf](http://www.iaeng.org/IJCS/issues_v35/issue_3/IJCS_35_3_16.pdf). Acesso em: 23/March/2009.

EMNA, F., HANENE, B.; ABDELMAJID, B. Soccer Video Edition Using a Multimodal Annotation. Proceedings of world academy of science, engineering and technology, v. 33, set. 2008, p 220-224

PURI, A.; CHEN, T. Multimedia Systems, Standards and Networks. Marcel Dekker, 2000.

LOOMIS, J. ; WASSON, M. VC-1 Technical Overview, Microsoft Corporation, 2007. Disponível em: <http://www.microsoft.com/windows/windowsmedia/howto/articles/vc1techoverview.aspx#Introduction>. Acesso em: 23/March/2009.

ROBBINS, J.N. Web design in a nutshell: a desktop quick reference. O'Reilly, 2006.

WANG, Z.; SIMONCELLI, E. P.; BONK, A. C. Multi scale structural similarity for image quality assessment. Center for Neural Science and Courant Inst. Of Math Sci., New York Univ., New York, NY. Nov. 2003.

GIUSTO D. D., F.; MASSIDDA, C.; PERRA. A Fast algorithm for video segmentation and object tracking - Multimedia communications Lab, University of Cagliari, Italy. Digital Signal Processing, 2002. 14th International Conference on. p. 697-700, 2002.

ZHU, H.; Li, Z. A Video segmentation algorithm based on spatial-temporal information. College of communication and information engineering, University of Electronic Science and Technology of China, Jul. 2002.

Recebido em 18 dez. 2008 / aprovado em 6 fev. 2009

### Para referenciar este texto

HIRAKAWA, A. R. et al. ViBee – algoritmo de segmentação aplicado na compressão de vídeos em Weblabs de abelhas. *Exacta*, São Paulo, v. 7, n. 1, p. 99-107, jan./mar. 2009.