FAST RELIABLE MULTI-SCALE MOTION REGION DETECTION IN VIDEO PROCESSING

Jiangbo Lu [*,†], *Gauthier Lafruit* [†], *and Francky Catthoor* [*,†]

[*] Department of Electrical Engineering, University of Leuven, Belgium
[†] Multimedia Group, IMEC, Kapeldreef 75, B-3001, Leuven, Belgium

ABSTRACT

Motion region detection is an important vision topic usually tackled by a background subtraction principle, which has some practical restrictions. We hence propose a multi-scale motion region detection technique that can fast and reliably segment foreground motion regions from two successive video frames. The key idea is to leverage multi-scale structural aggregation to effectively accentuate real motion changes while suppressing trivial noisy changes. Consequently, this technique can be effectively applied to motion region-of-interest (ROI) based video coding. Our experiments show that the proposed algorithm can reliably extract motion regions and is less sensitive to thresholds than single-scale methods. Compared with a H.264/AVC encoder, the proposed semantic video encoder achieves a bitrate saving ratio of up to 34% at the similar video quality, besides an overall speedup factor of 2.6 to 3.6. The motion-ROI detection can process a 352×288 size video at 20 fps on an Intel Pentium 4 processor.

***Index Terms*—** Motion detection, image change detection, video coding, video signal processing, pose recognition

1. INTRODUCTION

Motion region detection is one of important early vision tasks for many high-level semantic video processing applications, e.g., automated video surveillance. Different from the traditional background subtraction paradigm [1], segmenting motion regions based on the trained background models, we propose to detect motion regions simply from two successive video frames. The techniques from this category can bypass some practical constraints, such as the initial background training period, and still suffice to provide salient motion regions for several applications, e.g., semantic video coding.

By far, many algorithms have been proposed for motion detection and image saliency detection. However, striking a good trade-off between the detection quality and computational load still remains a challenge. An efficient static region detection scheme is adopted for *bi-level video* in [2], but it can not deal with complicated scenes well or distinguish separate moving objects. To construct a scale-invariant saliency map from a static image, a hybrid multi-scale approach is proposed in [3], and yet it involves a complicated image segmentation stage as well. The coarse-to-fine strategy in [4] performs a coarse-level detection at a more than 10-times reduced image scale to achieve the computational efficiency, but the detection quality is hence compromised. Moreover, an integrated algorithm fully exploiting the cross-scale interrelation is not presented.

Aiming at an integrated fast and reliable solution, we propose a hybrid motion region detection technique using multi-scale structural change aggregation to accentuate the signals, while suppressing noise at different levels of processing. We further propose a few specific algorithm changes to reduce the complexity. Finally, a promising motion-ROI based video coding scheme is proposed, resulting in much improved performance. Its key idea is to encode the motion foreground regions only, while repeating the background scene.

2. A HIERARCHY APPROACH TO MOTION REGION DETECTION USING MULTI-SCALE AGGREGATION

To reliably segment moving foreground objects from the input video frames, we use a bottom-up hierarchical approach that consists of two different levels: 1) pixel level processing and 2) region level processing. At the pixel level, the proposed multi-scale structural change analysis is adequate to identify semantically important image changes, by filtering out the noise. Subsequently, we propose a series of fast and effective processing at the region level to group the detected motion pixels and further cull out spurious noisy regions.

2.1. Pixel level: multi-scale structural change detection

Instead of extracting feature contrast at a fixed scale, the proposed multi-scale feature space analysis is essentially meant to build up a reliable motion saliency map, by aggregating the support from different scales. Since image noise is inherently structureless whereas the real motion changes possess strong correlations across different scales, such a multi-scale aggregation actually functions as an adaptive filter, where signals are largely accentuated and the intensity of noise is effectively suppressed. The proposed multi-scale pixel level processing is depicted in Fig. 1, with the description as follows:

Step 1: Noise reduction. For the current luma frame $I_t(x, y)$ at time t, we use a median filter to reduce the noise and denote the resulting image as $I_t^{'}$. Note that whenever appropriate, we omit (x, y) from notations denoting two dimensional images, e.g., $I_t^{'}$.

Step 2: Construct Gaussian image pyramid $\mathbf{G}_t$ from $I_t^{'}$:

$$\mathbf{G}_t = \left\{ G_{l,t} : G_{l,t} = \downarrow(G_{l-1,t}) \text{ and } G_{0,t} = I_t^{'}, l = 0, 1, ..., N-1 \right\}, \quad (1)$$

where $\downarrow(\cdot)$ is a Gaussian downsampling filter. Depending on the input frame resolution, N is typically set to 2 or 3 in our implementation to achieve a good trade-off between the quality and complexity.

Step 3: Apply the Laplacian operator to $\mathbf{G}_t$ and attain multi-scale Laplacian maps of the input image, $\mathbf{L}_t$. We adopt the second derivative of the pixel intensity to extract the underlying structure and eliminate the impact of the first order illumination changes [2]. For a 3×3 kernel, the Laplacian of a pixel is simply given by 8 times the central pixel's intensity minus the sum of its neighboring pixels.

$$\mathbf{L}_t = \nabla^2 \mathbf{G}_t = \partial^2 \mathbf{G}_t / \partial x^2 + \partial^2 \mathbf{G}_t / \partial y^2 . \quad (2)$$

Step 4: Compute SAD (Sum of Absolute Difference) between $\mathbf{L}_t$ and $\mathbf{L}_{t-1}$, and the resulting $\mathbf{D}_t$ captures the structural changes for two successive frames at different scales, as shown in Fig. 1(c-e).

$$\mathbf{D}_t(x, y) = \sum_{u=x-1, v=y-1}^{u=x+1, v=y+1} |\mathbf{L}_t(u, v) - \mathbf{L}_{t-1}(u, v)| . \quad (3)$$

Step 5: Aggregate SAD of Laplacian maps ($\mathbf{D}_t$) from each image scale (i.e., $D_{l,t}, l = 0$ to $N-1$) into a single saliency map S_t:

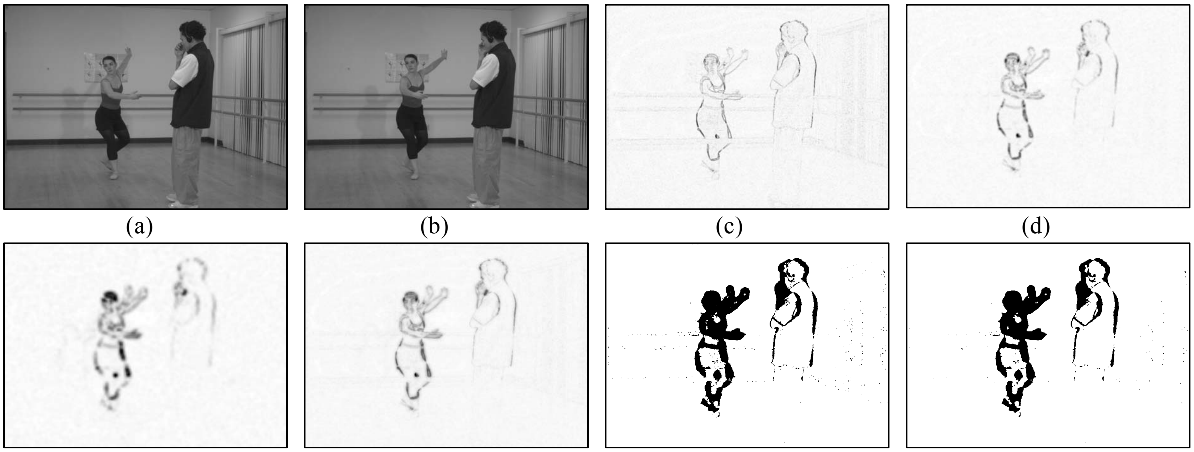

Fig. 1. The flowchart of the pixel level processing. (a) Previous luma frame I_{t-1} (b) Current luma frame I_t (c-e) SAD of Laplacian difference $\mathbf{D}_t$ at the image scale 0, 1, and 2, respectively (f) $\widehat{S}_t$: aggregating multi-scale $\mathbf{D}_t$ and normalizing the result S_t to $[0, 255]$ (g) Thresholding $\widehat{S}_t$ (h) Applying the median filter and morphological *closing* (dilation-erosion) operation. For ease of printing, the difference images are negated.

$$S_t = AGGREGATE(\mathbf{D}_t) = \sum_{l=0}^{N-1} \uparrow^l (D_{l,t}) \; , \tag{4}$$

where $\uparrow^l (\cdot)$ denotes performing the Gaussian upsampling operation l-times. We normalize S_t to get a gray-level image $\widehat{S}_t$, as in Fig. 1(f).

Step 6: Threshold the normalized gray-level image $\widehat{S}_t$ using an empirical value τ to generate a binary change mask, B_t (Fig. 1(g)):

$$B_t(x, y) = \begin{cases} 1 & \text{if } \widehat{S}_t(x, y) > \tau \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

Step 7: Apply the median filter and morphological *closing* (dilation & erosion) operation to B_t, so as to rule out isolated noise and fill gaps and holes in the motion regions, as shown in Fig. 1(h).

To avoid the computation redundancy in the video processing loop, we propose to store multi-scale Laplacian maps $\mathbf{L}_{t-1}$ rather than the previous luma frame I_{t-1}. This change actually leads to an overall speedup factor of **1.2** to **1.3** on an Intel Pentium 4 processor.

2.2. Region level: connectivity analysis and noise pruning

Although multi-scale structural change detection at the pixel level can segment the foreground motion regions very well, a clean map of correct motion blobs is hardly obtainable without enforcing the connectivity constraint (Fig. 2(b) vs. Fig. 1(h)). Therefore, we choose to employ a two-scan *connected component labeling algorithm* [5], which consists in assigning a unique label to each maximal connected region of foreground pixels (Fig. 2(a)). Since the noise at this stage are typically stray groups with a size smaller than the smallest real motion regions, they can safely be culled by restricting the labeled motion area to cover a minimum number of pixels (Fig. 2(b)).

To speed up the execution, the *connected component labeling algorithm* adopts an array rather than the pointer based rooted trees to store the label equivalence information. Moreover, a path compression strategy is incorporated to accelerate the key Union-Find process [6], and it reduces the complexity by **40%** for this specific part, compared with the implementation without using this scheme.

Usually, it is desirable that the detected motion blobs can be clustered into separate moving objects or bounding-boxes for high-level semantic video processing. To meet this requirement with little complexity overhead, we propose a fast histogram analysis approach that does not need recursive neighborhood distance checking or region growing. More clearly, we assume that the moving objects distribute principally in the horizontal direction, and we collect the distribution histogram of motion pixels for each column of the detected motion map (with a 2D histogram for the general cases). Based on such a histogram, bounding-boxes are constructed to contain all the motion pixels, and they are split whenever a minimum horizontal gap ϵ is satisfied between two neighboring groups (see Fig. 2(c)). As an option, bounding-boxes that are too narrow along the horizontal direction can be culled out. Finally, to favor semantic video coding (in Sect. 3), we align the bounding-boxes to the macroblock (MB) boundaries, and they are extended horizontally and vertically by 1 MB size to further guarantee the motion region detection results.

3. A NOVEL APPLICATION OF THE PROPOSED TECHNIQUE IN MOTION-ROI BASED VIDEO CODING

Thanks to its fast and reliable nature, the proposed motion region detection technique has a good potential in several high-level video processing applications, e.g., object tracking, pose recognition and object-based video coding. In this paper, we focus on a novel semantic video coding scheme that greatly benefits from the motion-ROI concept, without altering the well-established MB coding pipeline.

The proposed video coding method mainly targets at encoding video contents captured by stationary cameras, which find wide applications in video surveillance, news broadcast, and video conference. Typically, one of the unique characteristics associated with such stationary camera applications is that the moving foreground objects are of dominant interests, because they deliver critical semantic meaning compared with the static background scenes.

The key idea of our proposed video coding technique is to encode and transmit only the motion foreground regions defined by the metadata of bounding-boxes (e.g., Fig. 2(e)), while repeating the background regions from the previous reconstructed frame. Figure 3 illustrates the basic principle. In fact, this joint motion-ROI tracking and background replication scheme brings two clear advantages for efficient video coding: 1) the compressed bitrate can be largely reduced at the similar video quality, because only a limited number of MBs are encoded for each frame; 2) our experiments further show that the overall execution speed of motion-ROI based video coding

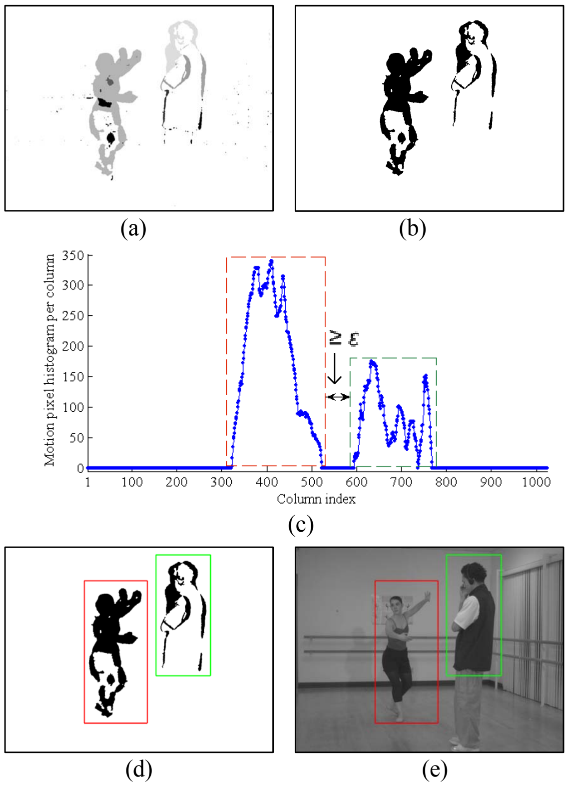

Fig. 2. The region level processing. (a) Fast connected component analysis (b) Culling small-size noisy regions to yield a binary map (c) Distribution histogram of motion pixels along the horizontal axis (d) Clustering blobs by bounding-boxes (aligned and extended by 1 MB size) (e) Superimposing the bounding-boxes on the input frame.

(including the detection overhead) is about three times faster than that of conventional frame-based video encoders, though motion region detection is needed as a preprocessing step. This indicates that the proposed multi-scale algorithm is very computationally efficient.

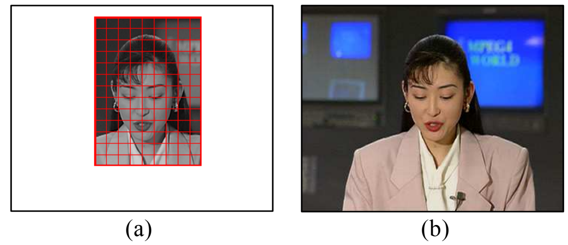

Fig. 3. Motion-ROI based video coding. (a) Encoding only the MBs belonging to the detected motion bounding-boxes (b) The reconstructed frame by stitching motion regions with static background.

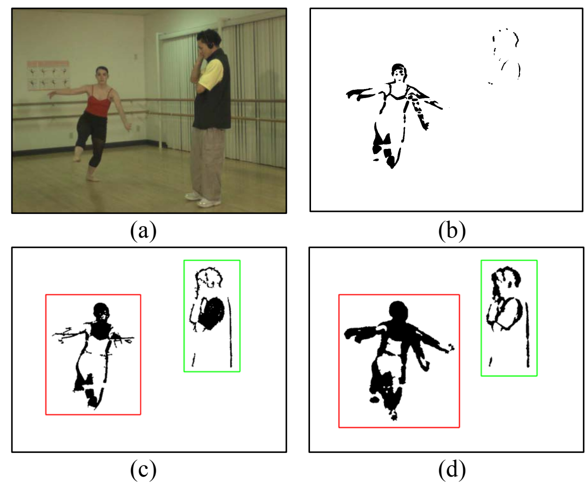

Fig. 4. Motion detection results for *Ballet* (at frame 6). (a) Current frame (b) Gaussian hypothesis test (c) The single-scale variant of the proposed algorithm (d) The proposed multi-scale algorithm.

4. EXPERIMENTAL RESULTS AND DISCUSSION

The proposed multi-scale motion region detection algorithm is implemented in C++ and OpenCV APIs [7]. We use two scales to detect motion regions for CIF size video sequences, i.e., *Hall* and *Akiyo*, and three scales for *Ballet* at 1024×768. The proposed semantic video codec is implemented on H.264 JM software, version 10.1 [8]. Baseline profile is used to configure the encoder. We set the number of reference frames to 1, and all frames except for the first one are encoded as P-frames. R-D optimization and CAVLC entropy encoding are enabled. Fast full search is adopted for motion estimation, and the search range is set to ± 16. Note that we have not conducted any special code-level optimization to either multi-scale motion detection or H.264 JM software. All the experiments are performed on a 3.2GHz Intel Pentium 4 processor with 1GB RAM.

4.1. Motion region detection results

Figure 4 and 5 clearly show that the proposed hybrid multi-scale approach can more reliably detect the motion blobs over Gaussian hypothesis test [1], as well as the single-scale variant of the proposed method. Therefore, proper bounding-boxes defining motion regions can be attained by our algorithm for various sequences with different visual features. *Ballet* is a multi-view video sequence from Microsoft Research, where some parts of the foreground dancer have low intensity contrasts against the background. *Hall* represents a video clip captured under unstable (or flickering) lighting conditions.

Because motion regions can be reliably tracked by our multi-scale technique, we find that the extension of bounding-boxes can be safely reduced to 1 MB size (even to 0), compared to a conservative value of 3 in our single-scale implementation. Such an algorithm upgrade can reduce the average bounding-box area, resulting in about **29%** less MBs to be coded by the proposed video codec for *Ballet*.

Furthermore, owing to this multi-scale processing, real feature changes can be accentuated while the noise is largely suppressed. Hence, the proposed algorithm is less sensitive to the threshold setting than the traditional single-scale image change detection techniques. For instance, for *Hall* (at frame 37) in Fig. 5, the valid threshold ranges (normalized to [0,1]) using Gaussian hypothesis test, the single-scale variant of the proposed algorithm, and the proposed multi-scale algorithm are **7%**, **4%**, and **14%**, respectively.

4.2. The performance of the motion-ROI based video encoder

Because we use the background replication for the static regions while only coding the foreground motion regions, the conventional frame-based PSNR becomes inappropriate in assessing the video quality. Moreover, the frame-based PSNR tends to be highly influenced by the original video's background noise variations (discarded in our encoder for they are classified as trivial noisy changes in the preprocessing steps), which would hence yield unfair PSNR figures. In order to properly emulate the perceptual quality in such situations, an objective quality metric is proposed [9], whose basic idea is to unevenly weight the errors in different image areas according to

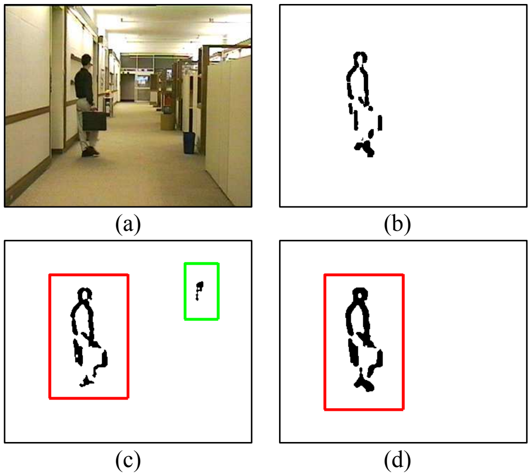

Fig. 5. Motion detection results for *Hall* (at frame 37). (a) Current frame (b) Gaussian hypothesis test (c) The single-scale variant of the proposed algorithm (d) The proposed multi-scale algorithm.

semantic partitions, e.g., background replication has a small impact on the overall image quality, compared to foreground regions. This method therefore favors constant foreground (FG) PSNRs, which has the highest impact on the visual quality. Table 1 presents the R-D performance comparison between the original H.264 encoder and the proposed semantic video encoder. At the negligible FG-PSNR changes and similar frame-level subjective quality, the proposed motion-ROI based video coding leads to a bitrate saving ratio of up to **34.1%**, compared with the original H.264 video encoder.

From the complexity aspect, we observe from Fig. 6 that the proposed semantic video encoder (including motion detection overhead) runs **2.6** to **3.6** times faster than the original H.264 encoder. We can still achieve a speedup factor of 2.3 to 3.2, when the simplified UMHexagonS [8] is used for fast motion estimation in both encoders. The reasons for such a significant speedup are two-fold: firstly, because a lower number of foreground MBs are encoded while the background MBs are skipped, the proposed coding process is largely accelerated. Secondly, since the proposed multi-scale algorithm is designed for a good balance between the quality and the speed, the complexity overhead due to this additional preprocessing is very limited. In fact, the speed test of our proposed multi-scale method indicates that a real-time frame-rate of **20** fps is reached for an image size of 352×288 on our 3.2GHz Intel Pentium 4 platform.

4.3. Extending the application to pose recognition

Not limited to its application in semantic video coding, the proposed motion region detection technique can also be applied to pose recognition by generating timed Motion History Image (tMHI) [10]. Rather than demanding an appropriate background model to extract silhouette maps, our algorithm can identify motion blobs from two successive frames, which suffice for constructing tMHI. Our experiments (not reported here due to the limited space) show that the movement of the dancer's arms in *Ballet* can be encoded in a single gray-level tMHI image, facilitating the high-level pose recognition.

5. CONCLUSION AND FUTURE WORK

A fast and reliable motion region detection algorithm is proposed to segment moving foreground objects from the input videos. The key contribution is our multi-scale structural change aggregation scheme, in addition to an integrated hierarchical motion detection and noise pruning approach, which yields a good trade-off between the quality

Table 1. R-D coding performance comparison between the original (Ori.) H.264 encoder and the proposed (Pro.) video encoder.

	QP	Average **FG-PSNR** for P frames (dB)			Average bitrate for P frames (kbps)		
		Ori. PSNR	Pro. PSNR	d-PSNR	Ori. bitrate	Pro. bitrate	d-bitrate
Akiyo	28	37.02	36.99	**-0.03**	75.39	63.56	**15.69%**
	32	34.09	34.10	**0.01**	40.01	34.06	**14.88%**
	36	31.41	31.46	**0.05**	23.12	19.36	**16.28%**
Hall	28	35.94	35.91	**-0.03**	237.89	156.85	**34.06%**
	32	32.96	32.94	**-0.02**	105.58	83.58	**20.84%**
	36	30.06	30.05	**-0.01**	54.70	47.00	**14.09%**
Ballet	28	40.78	40.72	**-0.06**	491.79	394.82	**19.72%**
	32	39.22	39.16	**-0.07**	312.39	260.52	**16.60%**
	36	37.40	37.30	**-0.10**	204.24	177.90	**12.90%**

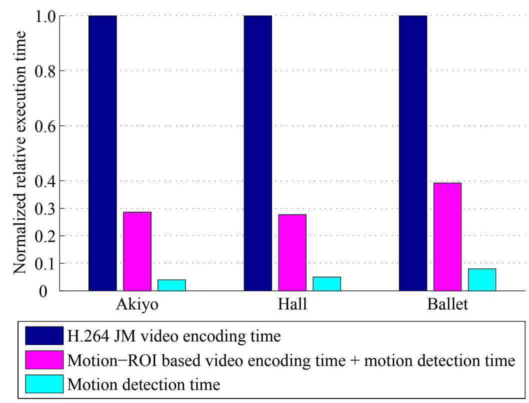

Fig. 6. Relative execution time comparison of different schemes.

and processing speed. Based on this technique, the performance of the proposed motion-ROI based video encoders is greatly boosted.

Future work will focus on improving the proposed algorithms in tackling abrupt illumination variations and dynamic scenes. Further accelerating the motion detection will make it more advantageous.

6. REFERENCES

[1] R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam, "Image change detection algorithms: a systematic survey," *IEEE Tran. on Image Processing*, vol. 14, pp. 294–307, Mar. 2005.

[2] J. Li et al., "Bi-level video: video communication at very low bit rates," in *ACM Multimedia*, 2001, vol. 9, pp. 392–400.

[3] F. Lius and M. Gleicher, "Region enhanced scale-invariant saliency detection," in *IEEE ICME*, 2006, pp. 1477–1480.

[4] A. Bevilacqua, L. D. Stefano, A. Lanza, and G. Capelli, "A novel approach to change detection based on a coarse-to-fine strategy," in *IEEE ICIP*, 2005, vol. 2, pp. 434–437.

[5] A. Rahimi, "Fast connected components on images," 2001, http://people.csail.mit.edu/rahimi/connected/.

[6] K. Wu, E. Otoo, and A. Shoshani, "Optimizing connected component labeling algorithm," in *Proc. SPIE Medical Imaging Conference*, 2005, vol. 5747, pp. 1965–1976.

[7] OpenCV Library, http://www.intel.com/technology/computing/opencv/index.htm.

[8] H.264 JM Reference Software Version 10.1, http://iphome.hhi.de/suehring/tml/download/old_jm/jm10.1.zip.

[9] A. Cavallaro, O. Steiger, and T. Ebrahimi, "Semantic video analysis for adaptive content delivery and automatic description," *IEEE Tran. on CSVT*, vol. 15, pp. 1200–1209, Oct. 2005.

[10] G. Bradski and J. Davies, "Motion segmentation and pose recognition with motion history gradients," in *IEEE WACV*, 2000, pp. 238–244.