EURASIP Journal on Applied Signal Processing 2002:6, 561–565


# Temporal Segmentation of MPEG Video Streams

**Janko Calic**
*Multimedia and Vision Research Lab, Department of Electronic Engineering, Queen Mary, University of London, Mile End Road, London E1 4NS, UK*
*Email: janko.calic@elec.qmul.ac.uk*

**Ebroul Izquierdo**
*Multimedia and Vision Research Lab, Department of Electronic Engineering, Queen Mary, University of London, Mile End Road, London E1 4NS, UK*
*Email: ebroul.izquierdo@elec.qmul.ac.uk*

*Received 30 July 2001 and in revised form 9 February 2002*

Many algorithms for temporal video partitioning rely on the analysis of uncompressed video features. Since the information relevant to the partitioning process can be extracted directly from the MPEG compressed stream, higher efficiency can be achieved utilizing information from the MPEG compressed domain. This paper introduces a real-time algorithm for scene change detection that analyses the statistics of the macroblock features extracted directly from the MPEG stream. A method for extraction of the continuous frame difference that transforms the 3D video stream into a 1D curve is presented. This transform is then further employed to extract temporal units within the analysed video sequence. Results of computer simulations are reported.

**Keywords and phrases:** shot detection, video indexing, compressed domain, MPEG stream.

## 1. INTRODUCTION

The development of highly efficient video compression technology combined with the rapid increase in desktop computer performance, and a decrease in the storage cost, have led to a proliferation of digital video media. As a consequence, many terabytes of video data stored in large video databases, are often not catalogued and are accessible only by the sequential scanning of the sequences. To make the use of large video databases more efficient, we need to be able to automatically index, search, and retrieve relevant material.

It is important to stress that even by using the leading edge hardware accelerators, factors such as algorithm complexity and storage capacity are concerns that still must be addressed. For example, although compression provides tremendous space savings, it can often introduce processing inefficiencies when decompression is required to perform spatial processing for indexing and retrieval. With this in mind, one of the initial considerations in development of a system for video retrieval is an attempt to enhance access capabilities within the existing compression representations.

Since the identification of the temporal structures of video is an essential task of video indexing and retrieval [1], shot detection has been generally accepted to be a first step in the indexing algorithm implementation. We define a *shot* as a sequence of frames that were (or appear to be) "continuously captured from the same camera" [2]. A *scene* is defined as a "collection of one or more adjoining shots that focus on an object or objects of interest" [3].

Shot change detection algorithms can be classified, according to the features used for processing, into uncompressed and compressed domain algorithms. Algorithms in the uncompressed domain utilize features extracted from the spatial video domain: pixel-wise difference [4], histograms [5], edge tracking [6], and so forth. These techniques are computationally demanding and time-consuming, and thus inferior to the approach based on the compressed domain analysis.

Development in this area is particularly focused on the use of the prevalent MPEG compression standard. Pioneering work by Arman et al. [7] introduced the initial approach to the compressed domain shot detection by analysing the Discrete Cosine Transform (DCT) coefficient subsets and their correlation. Yeo and Liu [8] analysed the sequence of the reduced images extracted from DC coefficients in the transformation domain called the *DC sequence*. Sethi and Patel [9] used DC sequence histograms to apply $\chi^2$ statistical test. Continuing in the similar manner, Lee et al. [10] exploited information from the first few AC coefficients in the transformation domain, and tracked binary edge maps to parse the video sequence. Although utilizing DCT coefficients appeared to be a much faster approach than the spatial domain analysis, processing time needed to apply motion compensation remained an obstacle to this approach. On the

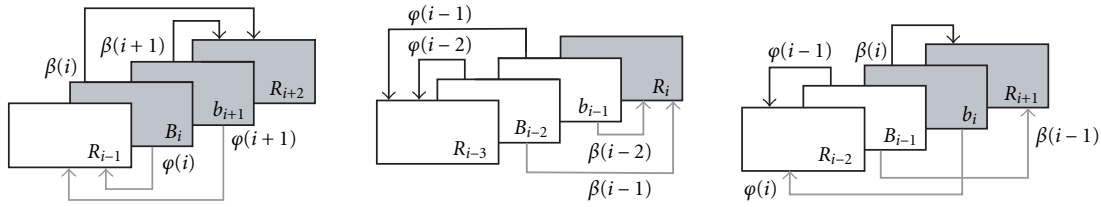



Figure 1: Possible positions of the cut in a frame triple.

other hand, the algorithms that omitted motion compensation and analysed only I frames, required a second pass to accurately detect the shot change at B or P frames.

Meng et al. [11] presented an original approach by utilizing only features directly embedded in MPEG stream: statistics on the numbers and types of prediction vectors used to encode P and B frames. Likewise, Kobla et al. [12] detect shot changes using discontinuous difference metrics and validate the changes by analysing the DCT data. A step forward was done by Pei and Chou [13] where they matched patterns of macroblocks (MB) types within abrupt or gradual change with the expected shapes combining it with partially spatial information. However, these methods have not shown real-time processing capabilities and none of them generated continuous output, essential for further scalable analysis.

In this paper, the main goal is to develop a new approach to the fundamental problems of a system for real-time video retrieval, searching, and browsing. The initial research objectives are directed towards the performance of the core video processing algorithms in the compressed domain, using the established international video standards: MPEG-1–2, H.263, and in future MPEG-4. This approach should introduce improvements in video retrieval with low access latency, as well as advances in processing speed and algorithm complexity. A method for extraction of a continuous frame difference that transforms the 3D video stream into a 1D curve is presented. This transform is then further employed to extract temporal units within the analysed video sequence.

This paper is organized as follows. In Section 2, the algorithm for detection of abrupt shot changes is presented. Section 3 describes the gradual transition detection algorithms that are built on the similar approach, as well as adds some interesting conclusions. Overall results are presented in Section 4, while Section 5 brings final conclusions and a summary of the paper.

## 2. SCENE CHANGE DETECTION

MPEG-2 encoders compress video by dividing each frame into blocks of size 16 × 16 called *macroblocks* (MB) [14]. An MB contains information about the type of temporal prediction and corresponding vectors used for motion compensation. The character of the MB prediction is defined in an MPEG variable called *MBType*. It can be *intra* coded, *forward* referenced, *backward* referenced, or *interpolated*. Within a video sequence, a continuously strong interframe reference will be present as long as no significant changes occur in the scene. The "amount" of interframe reference in each frame and its temporal changes can be used to define a metric, which measures the probability of scene change in a given frame. We propose to extract MBType information from the MPEG stream and to use it to measure the "amount" of interframe reference. Scene changes are then detected by thresholding the resulting function.

Without loss of generality, we assume that a *group of pictures* (GOP) in the analysed MPEG stream has the standard frame structure: [IBBPBBPBBPBBPBB]. Observe that this frame structure can be split into groups of three having the form of a triplet: IBB or PBB. In the sequel, both types of the reference frames (I or P) are denoted by $R_i$, first bidirectional frame of the triplet as $B_i$, while the second bidirectional frame is denoted as $b_i$. Thus, the MPEG sequence can be analysed as a group of frame-triplets in the form $R_1B_2b_3R_4B_5b_6\cdots R_iB_{i+1}b_{i+2}\cdots$.

This convention can be easily generalized to any other GOP structure. The possible locations of a cut in a frame-triplet are depicted in Figure 1. If the first referenced frame $B_i$ is the first frame in the next shot, the next reference frame $R_{i+2}$ predicts a significant percentage of interframe MBs in both $B_i$ and $b_{i+1}$. If the scene change occurs at $R_i$, then the previous bidirectional frames $B_{i-2}$ and $b_{i-1}$ will be mainly referenced to $R_{i-3}$. Finally, if the scene change occurs at $b_i$, then $B_{i-1}$ will be referenced to $R_{i-2}$ while $b_i$ will be referenced to $R_{i+1}$.

If two frames are strongly referenced, then most of the MBs in each frame will have the corresponding type, forward, backward, or interpolated, depending on the type of reference. Thus, we can define a metric for the visual frame difference by analyzing the percentage (or simply the number) of MBs in a frame that are forward referenced and/or backward referenced.

Let $\Phi_T(i)$ be the set containing all forward referenced MBs and $B_T(i)$ the set containing all backward referenced MBs in a given frame with index $i$ and type $T$. Then we denote the cardinality of $\Phi_T(i)$ by $\varphi_T(i)$ and the cardinality of $B_T(i)$ by $\beta_T(i)$. The frame difference metric $\Delta(i)$ is defined as

$$\Delta(i) = \begin{cases} \beta_B(i) + \beta_b(i+1), & \text{if the } i\text{th frame is a } B \text{ frame,} \\ \varphi_B(i-2) + \varphi_b(i-1), & \text{if the } i\text{th frame is an } R \text{ frame,} \\ \varphi_B(i-1) + \beta_b(i), & \text{if the } i\text{th frame is a } b \text{ frame.} \end{cases} \tag{1}$$

Since $\Delta(i)$ is a frame-to-frame difference metrics, peaks in the $\Delta(i)$ are presenting the strong and abrupt changes in

the video content. The cut positions are determined by thresholding using either predefined constant threshold or an adaptive threshold.

## 3. GRADUAL CHANGES DETECTION

The next step in the implementation of a shot change detection algorithm is the detection of gradual changes. Unlike cuts, gradual transitions do not show such a significant change in any of the features, and thus are more difficult to detect. Furthermore, there are various types of gradual changes: *dissolves*, where the frames of the first shot become dimmer, while the frames from the second one become brighter and superimposed; *wipes*, where the image of the second shot replaces the first one in a regular pattern, such as vertical line, and so forth. Since there is inevitably additional processing in feature analysis for gradual changes extraction, real-time implementation is even more unachievable than it is for basic cut detection.

To reduce additional processing for gradual changes, a new approach is applied. Since the change of features during a gradual transition lasts longer than the analysed frame-triplet unit, it is essential to include a component in a difference metrics that will be proportional to the overall change in a GOP. The difference metrics formula for a frame with index $i$ and type $T(i)$ is becoming a linear combination of cardinalities of macroblock type sets within one GOP:

$$\begin{aligned}\Delta(i) = {} & k_{\varphi B}\varphi_B + k_{\varphi b}\varphi_b + k_{\beta B}\beta_B + k_{\beta b}\beta_b \\ & + k_{\imath B}\imath_B + k_{\imath b}\imath_b + k_{\pi B}\pi_B + k_{\pi b}\pi_b.\end{aligned} \tag{2}$$

In addition to the previously defined sets $\Phi_T(i)$ and $B_T(i)$, sets of intracoded MBs are denoted by $I_T(i)$, while interpolated MBs are denoted by $\Pi_T(i)$. Cardinalities of the corresponding sets are denoted by $\varphi_T(i)$, $\beta_T(i)$, $\imath_T(i)$, and $\pi_T(i)$. The metric $\Delta(i)$ is proportional to visual changes within the frame triplet as well as to longer alterations during the gradual transitions. During a gradual change, the number of intracoded macroblocks is increasing, because of the lack in visual similarity with both reference frames. On the contrary, the number of interpolated macroblocks is falling, so that we can use this behaviour to enhance the metric sensitivity.

After noise suppression depicted in Figure 2, the same depending on the frame type, there are three different linear combinations of variables $\varphi_T(i)$, $\beta_T(i)$, $\imath_T(i)$, and $\pi_T(i)$ for both bidirectional frames in a frame triplet. Each linear combination has two main coefficients that are directly proportional to the visual content change within predicted and reference frame in a frame triplet ($k = +1$), and two that are inversely proportional ($k = -1$) to it. Additional factors $k_\pi$ and $k_\imath$ are describing overall change in a triplet, one in direct ($k_\imath$) and one in inverse ($k_\pi$) proportion. The coefficient values are determined by the rule of thumb, and are presented in Table 1.

The raw difference metric has a strong noise that makes further processing of the data almost impossible. However, we know that the source of this noise is in the discontinuous

Table 1: Coefficients in the linear combination $\Delta(i)$.

| | $T(I) = R$ | $T(I) = B$ | $T(I) = B$ |
|---|---|---|---|
| $k\varphi B$ | +1 | −1 | +1 |
| $k\varphi b$ | +1 | −1 | −1 |
| $k\beta B$ | −1 | +1 | −1 |
| $k\beta b$ | −1 | +1 | +1 |
| $k\imath B$, $k\imath b$ | +0.5 | | |
| $k\pi B$, $k\pi b$ | −0.5 | | |

nature of the difference metrics. Since the metrics value is determined separately for each frame and the content change is based on frame triplets, low-pass filtering with kernel proportional to triplet length would eliminate the noise. The filter with Gaussian pulse response is applied

$$h(i) = \frac{1}{\sqrt{2\pi\sigma}} e^{-i^2/2\sigma^2}, \tag{3}$$

where $i \in [-4\sigma, 4\sigma]$ and $\sigma = 1.5$. The value for $\sigma$ is chosen to maximize the smoothing within one frame triplet.

Metric with suppressed noise is calculated as a convolution of Gaussian filter pulse response and the raw noisy metrics

$$\Delta = \Delta_N \otimes h. \tag{4}$$

After noise suppression, the same filtering procedure is applied to eliminate small spurious peaks and to smooth the difference metrics function. As in noise suppression, the filtering kernel is Gaussian, but with parameter $\sigma = 3$. The positions of the central points in the shot change are determined by locating local maxima of the smooth metrics curve. Continuing the process of Gaussian filtering with increasing kernel value, the scale-space of metric curves is generated. It enables a multiresolution analysis of the temporal structure within the analysed video sequence.

## 4. RESULTS

The collection of C++ classes called MPEG development classes, implemented by Dongge and Sethi [15], are used as the main tool for manipulating the MPEG streams, while Berkeley mpeg2codec was used as the reference MPEG codec. Some test sequences were produced by Multimedia and Vision Research Lab, Queen Mary, University of London, while some others were provided by the School of Electronic Engineering, Dublin City University, Dublin, Ireland.

To show the typical behaviour of the first algorithm, a sample of MPEG-2 video sequence is generated with three abrupt shot changes at the 6th, 16th, and 23rd frame.

As depicted in Figure 3, the first cut is positioned at rear b frame, and as proposed, it is clear that the level of forward reference is high at previous B frame $\varphi(5)$, and that at the present frame there is strong backward referencing $\beta(6)$. In the same way, for the 16th I type frame there are significant levels of $\varphi(13)$ and $\varphi(14)$, and the 23rd B type frame has strong $\beta(23)$ and $\beta(24)$.

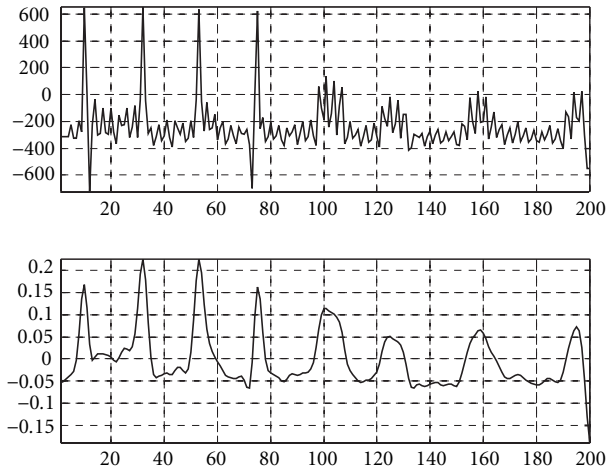


Figure 2: Noise suppression.

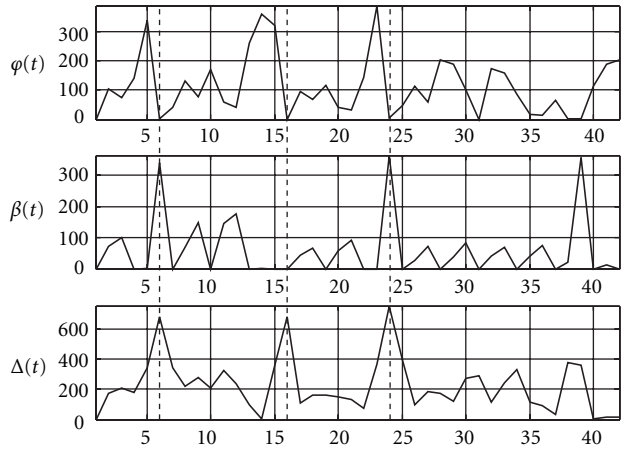



Figure 3: Cut detection example.

Stages of the noise suppression and smoothing process are depicted in Figure 4. It shows noisy raw metric, metric after the noise suppression, and the smoothed metric.

To evaluate the algorithms behaviour, statistical comparison "based on the number of missed detections (MD) and false alarms (FA), expressed as recall and precision" [2] is applied

$$\text{Recall} = \frac{\text{Detects}}{\text{Detects} + \text{MDs}}, \qquad \text{Precision} = \frac{\text{Detects}}{\text{Detects} + \text{FAs}}. \tag{5}$$

Performance comparison and dataset is based on the work of Gargi et al. [2]. Dataset is generated as an MPEG-1 sequence with resolution 320×240, having the same sequence length (1200 seconds), same number and type of transitions for a particular programme type (news, sports, and sitcom) as in the performance evaluation. Manually detected positions of the shot boundaries are taken as the ground truth, defining in that way the number of missed detections and

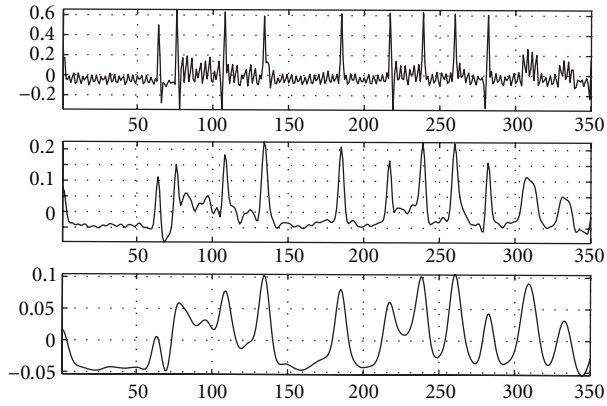


Figure 4: Noise suppression and curve smoothing.

Table 2: Algorithm comparison results.

| | Detects | | MD | | FA | | Recall | | Precision | |
|---|---|---|---|---|---|---|---|---|---|---|
| MA | 932 | 0 | 27 | 289 | 14820 | 21 | 97% | 0% | 6% | — |
| MB | 473 | 75 | 486 | 214 | 3059 | 922 | 49% | 26% | 13% | 8% |
| MD | 754 | 83 | 205 | 182 | 105 | 952 | 79% | 31% | 88% | 8% |
| ME | 862 | 32 | 97 | 233 | 4904 | 218 | 90% | 12% | 15% | 13% |
| SP | 672 | — | 287 | — | 35 | — | 70% | — | 95% | – |
| Δ | 943 | 193 | 16 | 96 | 67 | 124 | 98% | 67% | 93% | 61% |

false alarms. Next to the presented algorithm labelled Δ, four algorithms based on MPEG compressed domain features and one based on spatial domain are evaluated. Results of the algorithm comparison are presented in Table 2.

Algorithm labelled MA utilizes correlation of DCT coefficient vectors and MBType statistics [7], while MB uses variance and prediction statistics of macroblock prediction types [11]. MD analyses DC sequence differences [8], whilst ME applies $\chi^2$ statistical test on DC coefficients [9]. Two algorithms from the spatial domain family showed the best results: 1D bin-to-bin colour histogram comparison in LAB colour space, and 3D histogram intersection in Munsell colour system.

## 5. CONCLUSIONS

A novel scene change detection technique based on the motion variables extracted from the MPEG video stream is proposed. First, a method for abrupt changes detection that uses interframe reference derived only from the statistics of the macroblock types was introduced. Second, the similar interframe reference metrics was applied in the algorithm for gradual shot detection. Improved frame difference metric that utilizes additional MBType information enables the detection of longer transitions. Finally, the experimental results were introduced in Section 4.

Performance comparison with other compressed domain algorithms shows much better results in terms of both recall and preciseness. Furthermore, implementation of the algorithm on PC 750 MHz workstation runs four times faster than real-time requirement for CIF MPEG-1 stream. Unlike

the most MPEG-based video partitioning methods, this algorithm generates continuous 1D frame difference metric, suitable for further steps of video indexing. A scale-space of curves can be generated to index the sequence in hierarchical and scalable way.

Possibilities of improving the real-time gradual shot changes detection by using multidimensional clustering of MPEG compressed features are investigated. Also, the multiresolution analysis of the temporal structure for hierarchical and scalable video indexing is in the development process.

## ACKNOWLEDGMENTS

The research leading to this paper has been supported by the UK EPSRC, project Hierarchical Video Indexing Project, grant number R01699/01.

## REFERENCES

[1] H. J. Zhang, "Content-based video browsing and retrieval," in *The Handbook of Multimedia Computing*, B. Furht, Ed., CRC Press, Boca Raton, Fla, USA, 1999.

[2] U. Gargi, R. Kasturi, and S. Strayer, "Performance characterization of video-shot-change detection methods," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 10, no. 1, pp. 1–13, 2000.

[3] J. S. Boreczky and L. A. Rowe, "A comparison of video shot boundary detection techniques," in *Storage and Retrieval for Image and Video Databases IV*, I. K. Sethi and R. C. Jain, Eds., vol. 2670 of *Proceedings SPIE*, pp. 170–179, 1996.

[4] H. J. Zhang, A. Kankanhalli, and W. Smoliar, "Automatic partitioning of full-motion video," *Multimedia Systems*, vol. 1, no. 1, pp. 10–28, 1993.

[5] A. Nagasaka and Y. Tanaka, "Automatic video indexing and full-video search for object appearances," in *Proc. IFIP 2nd Working Conf. Visual Database Systems*, pp. 113–127, Budapest, Hungary, 30 September–3 October 1992.

[6] R. Zabih, J. Miller, and K. Mai, "A feature-based algorithm for detecting and classifying scene breaks," in *Proc. ACM Multimedia '95*, pp. 189–200, San Francisco, Calif, USA, November 1996.

[7] F. Arman, A. Hsu, and M.-Y. Chiu, "Feature management for large video databases," in *Proc. IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases I*, vol. 1908, pp. 2–12, San Jose, Calif, USA, November 1993.

[8] B.-L. Yeo and B. Liu, "Rapid scene analysis on compressed video," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 5, no. 6, pp. 533–544, 1995.

[9] I. K. Sethi and N. Patel, "A statistical approach to scene change detection," in *Proc. IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases III*, vol. 2420, pp. 329–338, San Jose, Calif, USA, February 1995.

[10] S.-W. Lee, Y.-M. Kim, and S. W. Choi, "Fast scene change detection using direct feature extraction from MPEG compressed videos," *IEEE Transactions on Multimedia*, vol. 2, no. 4, pp. 240–254, 2000.

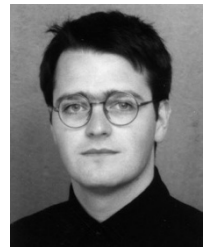
[11] J. Meng, Y. Juan, and S. F. Chang, "Scene change detection in a MPEG compressed video sequence," in *Proc. SPIE/IS&T Symposium on Electronic Imaging Science and Technology: Digital Video Compression: Algorithms and Technologies*, vol. 2419, pp. 14–25, San Jose, Calif, USA, February 1995.

[12] V. Kobla, D. S. Doermann, K.-I. Lin, and C. Faloutsos, "Compressed domain video indexing techniques using DCT and motion vector information in MPEG video," in *Proc. IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases V*, vol. 3022, pp. 200–211, San Jose, Calif, USA, February 1997.

[13] S.-C. Pei and Y.-Z. Chou, "Efficient MPEG compressed video analysis using macroblock type information," *IEEE Transactions on Multimedia*, vol. 1, no. 4, pp. 321–333, 1999.

[14] D. LeGall, J. L. Mitchell, W. B. Pennbaker, and C. E. Fogg, *MPEG Video Compression Standard*, Chapman & Hall, New York, NY, USA, 1996.

[15] L. Dongge and I. K. Sethi, "MDC: a software tool for developing MPEG applications," in *Proc. IEEE International Conference on Multimedia Computing and Systems*, vol. 1, pp. 445–450, Florence, Italy, 1999.



**Janko Calic** received the B.Eng. Honours Degree in electronic engineering at University of Belgrade, Yugoslavia and is currently a Ph.D. research student at Multimedia and Vision Research Lab, Queen Mary, University of London. His research interests include multimedia processing and analysis for content-based indexing and retrieval, multimedia understanding and semantic analysis.

**Ebroul Izquierdo** is a Lecturer in the Department of Electronic Engineering, Queen Mary, University of London. He received his Ph.D. in applied mathematics from the Humboldt University, Berlin, Germany. Dr. Izquierdo worked from 1993 to 1997 at the Heinrich-Hertz Institute for Communication Technology, Berlin, developing and implementing techniques for stereo vision, disparity and motion estimation, video compression, 3D modelling and immersive telepresence. Dr. Izquierdo has been involved in research and management of projects in Germany, the UK and two European projects in the Multimedia field. Currently, he is the coordinator of the European IST project BUSMAN and represents QMUL in the European Network of Excellence SCHEMA. Dr. Izquierdo is a Chartered Engineer, member of the IEEE, the IEE and the British Machine Vision Association. He represents the UK in the European COST211 forum. He is a member of the management committee of the Information Visualization Society, the IASTED technical committee on Image Processing, the program committee of the IEEE conference on Information Visualization, the international program committee of EURASIP&IEEE conference on Video Processing and Multimedia Communication and the COST211 sponsored European Workshop on Image Analysis for Multimedia Interactive Services. Dr. Izquierdo has published over 100 technical papers including chapters in books.