ABSTRACT

Multimedia data accessibility depends on a precise indexing, involving a computational cost. This paper proposes a new fast method of segmentation and indexing in order to fill out in an automatic way several MPEG7 [1] fields (e.g. camera and objects movement). In order to accelerate segmentation process, we exploit most of the information contained in MPEG1-2 flow [2], the decompression is restricted to entropic decoding and inverse quantization, the estimation of the camera movement is obtained from MPEG1-2 movement prediction. Segmentation in homogeneous color zones is obtained by a "split and merge" algorithm improved by a B-Splines active contour segmentation regularization [3].

1. INTRODUCTION

With the increase of the quantity of multimedia data, MPEG7 [1] gives a standardized and performed indexing form. Our purpose is to present an automatic algorithm for indexing with a high time constraint, meanwhile, we lose in precision. We use MPEG1-2 flow (e.g. format of TV satellites, DVD) and exploit most of the analysis carried out during compression. Movement prediction enables to calculate the camera movement; DCT coefficients of error images bring information on the prediction accuracy. DCT coefficients are given by entropic decoding and inverse quantization. We calculate neither the inverse DCT nor the image reconstruction at the pixel level.

The outline of this paper is as follows: section 2 introduces what we use thereafter: firstly, a summary of the MPEG1-2 standard; secondly, the principle of spatio-temporal segmentation method developed in [3]. Section 3 presents methods implemented for the camera movement estimation and the objects segmentation. Finally, section 4 shows experimental results.

2. NECESSARY KNOWLEDGE

2.1. MPEG1-2 Standard (video part)

MPEG1-2 exploits the strong temporal correlation between successive images in a film. It cuts the film into groups of pictures (GOP) beginning with an Intra (I) followed by Predicted (P) and Bidirectional predicted (B):

* I type images: JPEG like coding which uses DCT, quantization and entropic coding.

* P type images: for each macroblock, the coder searches for corresponding zone in the previous image (I or P). This leads to field of displacement vector (\( \vec{F} \)) which can be assimilated to a movement estimator. If the prediction is precise, the error macroblock has a low average and standard deviation.

* B type images: two predictions are given, one forward (\( \vec{F}_e \)) from previous P or I (like for P type macroblocks) and the other, backward (\( \vec{F}_b \)), from following P or I.

2.2. Active contour segmentation

Boundaries are regularized with a continuous model in order to improve the segmentation in uniform colors zone. Thanks to a method described by [4], we can modify the boundary modelized by cubic B-Spline interpolation [3], on which we apply the force \( F^{(i,j)} \). We use this evaluation directly:

\[
F^{(i,j)} = k_{in}^{(i,j)} - k_{out}^{(i,j)} + \lambda \kappa^{(i,j)}
\]

(1)

with \( \kappa \) the contour curvature and

\[
k_{in}^{(i,j)} = \left( S_n^{(i,j)} - S_{n+1}^{(i,j)} \right)^2, \quad k_{in} = 0.037, \quad \lambda = 0.005
\]

(2)

\( S_n^{(i,j)} \) represents the value of pixel \((i,j)\) in image \(n\) B-Spline method gives directly the contour curvature avoiding the necessity of an heavy computational load.
3. IMPLEMENTED ALGORITHM

The broad outline of the implemented algorithm is firstly the apparent camera movement estimation; secondly, images are split into homogeneous luminance and chrominances zones. Results are refined by an active contour segmentation. The next step is merging: the zone movement is then used. After objects extraction and monitoring, the objects movements are estimated. With all these data, some MPEG7 fields are filled out.

3.1. SFV\(^1\) estimation

In order to estimate apparent camera movement, it is necessary to obtain the apparent movement between two given images (SFV). Unfortunately, Intra images do not contain movement vectors.

Considering a time continuous movement, we interpolate the movement vector given in flow in order to obtain the movement with the previous image: \( \vec{N}_i \) (the SFV for the image \( i \), attached with a block \((i,j)\)).

* Intra Case (only for Intra images): \( \vec{N}_i = \frac{\vec{F}_{i+1} - \vec{F}_{i}}{2} \).

* Predict Case (for all Predict macroblock): \( \vec{N}_i = \frac{\vec{F}_{i+1} - \vec{F}_{i}}{n_{b} + 1} \).

* Bidirectional Case: \( \vec{N}_i = \frac{\vec{F}_{i+1} - \vec{F}_{i}}{n_{b} + 1} \).

3.2. ACM\(^2\) determination

ACM is defined by the movement vector \( \vec{N}_i \). We assume that background surface is larger than moving objects surface; otherwise, we estimate the movement of the greatest object and no more the camera movement. We study the simplified system with six unknowns [1]:

\[
\begin{align*}
\vec{u}_x &= -\frac{1}{f}(T_X - xT_Z) + y.R_Z \\
\vec{u}_y &= -\frac{1}{f}(T_Y - yT_Z) - x.R_Z
\end{align*}
\]  

(\(x,y\), position on the retina; \(\vec{u}_x, \vec{u}_y\) coordinates of \( \vec{N}_i \); \(f\), focal distance; \(Z\), distance from the object to the retina; \(T_X\) (resp. \(T_Y\) or \(T_Z\)), translation according to \(X\)-axis (resp. \(Y\) or \(Z\)) called Pan (resp. Tilt or Zoom); \(R_Z\), rotation according to the \(Z\)-axis.

Assuming a case of a pinhole camera in an orthographic model, we approximate \(\frac{1}{f}\) by a constant, and obtain a system with four unknowns \((T_X, T_Y, T_Z, R_Z)\).

The estimation of considered movement \((\vec{u}_x, \vec{u}_y)\) is given by minimizing the following criterion:

\[
J = \sum_{i,j} \left\{ [\vec{u}_x - \vec{u}_{x,i,j}]^2 + [\vec{u}_y - \vec{u}_{y,i,j}]^2 \right\}
\]  

(4)

By solving the system \( \frac{\partial J}{\partial T} = 0 \) with \( L \in \{T_X, T_Y, T_Z, R_Z\} \), we obtain an estimation of the four unknowns. This approach gives a quicker result than in [5] and [6].

To improve the camera movement precision, incoherent vectors are rejected, which leads to keep the background vectors. By assuming that the error movement on the background follows a centered normal distribution, and by eliminating the vectors outside the confidence interval fixed at 90\%, we iterate estimation.

The choice of the maximum amplitude search of the similar block in the previous (or following) image \( P \) or \( I \) is a fixed parameter of the coder and the vector movement is estimated only in order to maximize the similarity and not a real movement calculation [7]. Therefore, this maximum value is not sent in flow; it can not exceed \( \pm 7 \) pixels between two consecutive images in IBBP case. The quality of movement estimation depends on the arbitrary choice made by the coder. The larger this value is, the longer the search becomes. Fortunately, such an important movement of the main object is not common.

3.3. By colors interest zones detection

Our aim is to segment each image in zones of uniform color (luminance and chrominances). We have the average values by block for luminance and macroblock for the chrominances, and we wish to obtain major zones. It is not necessary during decompression process to determinate the inverse DCT transform. For an Intra, these values are obtained, from a multiplicative constant, by coefficients DC of DCT blocks. The movement vectors, the averages of the previous \( P \) or \( I \) blocks (resp. previous and following) and average values of error image blocks, allow the reconstruction of the average values of a \( P \) block (resp. a \( B \) block). Same method is applied on chrominances macroblocks [8].

\(^{1}\)Standardized Forward Vector: movement between the macroblock in the image and the previous one [5]

\(^{2}\)Apparent Camera Movement

Fig. 1. Coded with MPEG Software Simulation Group 1994 coder

In Fig. 1, the car executes a translation of six pixels between the two images. In I-b, movement vectors correspond nearly to the real movement of the object, which can not be obtained in I-a. Error image allows prediction relevance check.

\[ a \rightarrow \text{Max. search: 3 pixels} \quad b \rightarrow \text{Max. search: 7 pixels} \]
3.3.1. 3D distance criterion

We introduce a color distance value between two blocks (in the same image or in two successive images) involving the knowledge of their belonging to the same object. The standardization of each component by image mean, allows the estimation robustness. In 3D, a block \((i,j)\) and another block are close if they are in a 4-vicinity in the current image but also if they are in a 25-vicinity (centered in \((i,j)\)) in the previous or following image. It is thus necessary for each block to look at 54 neighbors.

For a block \((i,j)\), in image \(n\), its distance with a close block is defined by:

\[
\text{dist}_{i,j}^n(f) = \frac{\sum_{l \in L} \left( \frac{f_{n}^{l} - f_{m}^{l}}{\text{mean}_{p}^{l} \lambda_{t}} \right)^{2}}{\sum_{l \in L} \lambda_{t}}
\]

\(f_{n}^{l}\) and \(f_{m}^{l}\), mean\(_{p}\) and \(\lambda_{t}\), stand respectively for the average value of two neighbor blocks in image \(n\) and \(m\), the average value in image \(p\) and a weight value.

Note : for the separation of two blocks in the same macroblock of an image, \(\lambda_{ACr}\) and \(\lambda_{ACs}\) are pointless.

For each block \((i,j)\) of an image \(n\), we determinate the value of \(\text{dist}_{i,j}^n(f)\). The belonging to a same object of two blocks is evaluated according a variable threshold. The determination of the threshold is explained in 3.3.2.

3.3.2. Iterative merging

To make the method more robust, we take a rather small threshold, and we carry out several iterations by increasing the threshold at each turn of algorithm loop. The distance is calculated not between two colors of block but between the mean color of each zone. As the first turn of loop, each zone is reduce to one block. The use of the same threshold for any kind of sequences is surely not optimal. Assuming that the probability density function of \(\text{dist}_{i,j}^n(f)\) is an exponential distribution, threshold can be weighted, by the mean or by the standard deviation, involving a robust parameterization.

To refine adaptive merging, the standard deviation of each zone is used [9] i.e., if the merging increases significantly the standard deviation, merging is canceled.

With the contour model (1), boundaries are modified involving a fine tuning.

3.4. Merging color zones using movement

To fill out MPEG7 moving objects fields, color zones with a coherent movement (9] belonging to the same object) must be merged. The difference between the MPEG1-2 movement vector and the ACM vector provides results that can only be used in the background. According to (3.2), objects can move more than 7 pixels and the overlapping zones reduce dramatically the number of usable vectors. Each macroblock can be classified in two categories. In the first are included macroblocks whose vectors belong to the confidence interval (high probability to be a part of the background). The second includes macroblocks whose vectors are rejected as well as the ones coded without movement prediction.

For objects in movement, after iterative merging step, a same color zone can be followed during successive images. This apparent movement is extracted and zones having the same movement are joined together.

3.5. Objects movement estimation

For objects, the number of MPEG vectors is too low and not often pertinent. The apparent movement of object gives the translational object displacement between two images, so we estimate the object displacement (\(T_X\) and \(T_Y\)). The \(T_Z\) (zoom) estimation needs several images.

3.6. Indexing

We obtain, in real time, the apparent camera movement estimation, the extraction of moving objects in the sequence, the estimation of their movement in the scene (reduced for this moment to \(T_X\) and \(T_Y\)).

It is now possible to monitor an object from an image to the next one; to obtain the image numbers where it appears and disappears; to estimate the movement of its including rectangular minimum box and its percentage occupancy. According to this information, some MPEG7 fields are filled out.

4. EXPERIMENTAL RESULTS

4.1. ACM examples

We compare our method with an elaborated method [10] with six parameters, block 5x5, a precision prediction of a quarter of pixel (MPEG1-2 has a precision of half pixel) and a search at 15 pixels. The two methods are not optimized.

On the truncated sequence Stefan Edberg (COST 211) (images from 17 to 160), the cost of our method (ACM and segmentation) is 18 seconds (on Intel PIII 500 Mhz), unlikely the others (ACM only) takes ten hours.

On Fig. 2-a, excepted between images 85 to 110, we see that our method gives good results (±0.75 pixel in comparison with the other method). The use of the movement vectors of MPEG1-2 will never give a movement of Pan or Tilt higher that 7. We avoid taking into account the erroneously estimated movement. The graph of the standard deviation of error macroblocks (Fig. 2-b) shows a significant increase
where the real movement is larger than the searched movement. We use this criterion to validate the camera movement estimation or to detect when a more precise algorithm is needed.

Here is another example, with, between each frame, a Pan near -1 pixel, Tilt near 1 pixel, Rotation near -0.01 radian and Zoom near -0.03 (Fig. 3).

**Fig. 3.** Comparison between the two methods concerning the estimation of the camera Rotation and Zoom

### 4.2. Segmentation example

The following images are extracted from sequence Hall (COST 211). First image (4-a) stands for the ruptures between blocks of uniform luminance and chrominances zones. Second image (4-b) stands for the merging obtained by the movement vectors of each zone.

![image](image)

**Fig. 2.** a - Comparison between the two methods concerning the estimation of the camera Pan, b - Standard deviation of DC error macroblocks values

**Fig. 4.** a - Segmentation by luminance and chrominances, b - Merging by movements, c - Moving object zoom

### 5. CONCLUSION

The proposed method avoids resorting to complete MPEG1-2 flow decompression. It allows to take the greatest advantage on the informations provided by the flow as vectors movement for images B and P, or DCT coefficients. Remaining on the concept of block and macroblock coefficients allows time cost decrease (near real time) with a lesser precision loss. The camera movement estimation is near to the correct movement with a precision of ±0.75 pixel for translations (Fig. 2).

In the future, in order to still refine the result, it will be possible to pass at the pixel level only in one small neighborhood around the object.

### 6. REFERENCES


