An Intensity-Driven Dissolve Detection Adapted to Synthetic Video Contents

Bogdan IONESCU[a,b], Patrick LAMBERT[b]
[a] LAPI-ETTI, University "Politehnica" of Bucharest, 061071, Romania
bionescu@alpha.imag.pub.ro.
[b] LISTIC, Polytech Annecy-Chambery, University of Savoie, 74944 France
patrick.lambert@univ-savoie.fr.

January 31, 2013

Submitted to SPIE International Journal on Electronic Imaging

ABSTRACT

In this paper we approach the problematic of video temporal segmentation. We propose an intensity-based dissolve detection approach that is able to perform on animated video contents. It uses the hypothesis that during a dissolve, the amount of fading-out and fading-in pixels should be significant compared to other visual transitions. We use this information as a visual discontinuity function. Instead of just applying a global threshold to filter these values, as most of the existing approaches do, we use a twin-thresholding approach and the shape analysis of the discontinuity measure. This allows us to reduce false detections caused by steep intensity fluctuations, as well as to retrieve dissolves caught up in other visual transitions (e.g. caused by movement, color effects, etc.). Experimental tests conducted on more than 452 dissolve transitions show that whether classic approaches tend to fail, the proposed method is able to provide good performance achieving average precision and recall ratios above 94% and 79.6%, respectively.

Keywords: dissolve detection, video gradual transitions, animated movies.

1 Introduction

Most of the existing video analysis techniques rely on temporal segmentation as a first processing step, because it provides a basic understanding of the movie temporal structure. At its highest level of granularity, temporal segmentation means parsing the video into its basic temporal units or *video shots*. A shot is defined as a continuous sequence of frames recorded between a camera switch on and off. In order to constitute the final video (usually denoted the final cut), in the editing phase video shots are linked together by means of *video transitions*. From this perspective, temporal segmentation roughly means detecting the video transitions that make the connection between consecutive shots.

There are two categories of video transitions: sharp and gradual. The most frequent are the *sharp transitions*, or cuts. A cut is a direct concatenation of two consecutive shots and produces an important visual discontinuity in the visual flow. Depending on video genre, 30 minutes of video may account for up to 300 cuts [8]. On the other hand, we have the *gradual transitions*, such as fades, dissolves, mattes, wipes, etc. [7]. Gradual transitions are short-in-time visual effects. Their occurrence frequency in the sequence is more reduced, being at least one order measure less than for the cuts. From the gradual transitions, the most commonly used within entertainment videos are the *fades* and *dissolves*. Fades are a gradual emerging of a certain image from a constant image, typically black (i.e. a fade-in sequence) or vice-versa, the gradual disappearance of an image into a black frame (i.e. a fade-out sequence). Dissolves are closely related to fades (at some level they can be perceived as the superposition of a fade-out with a fade-in transition) and involve a gradual transition at pixel-intensity level of a certain image into another (several examples are depicted in Figure 7) [18].

Compared to cuts, for which most of the actual detection techniques are highly accurate and common detection ratios are above 95% (see early TRECVid campaign [17]), gradual transitions are much difficult to detect. This is mainly due to the highly complex content transformations involved.

Dissolve detection is one of the still open issues, current detection ratios being in average situated around 80%. Dissolves are much more complex transitions, even compared to fades, due to many constraints:

- are hard to be temporally or spatially separated,
- most of the specific parameters retrieved at image level have similar time evolution as some camera/object movements,
- may undergo many variations (there are cross-dissolves, additive-dissolves),
- may involve similar color distributions or spatial layouts of the start and end image, etc.

The existing dissolve detection techniques can be grouped around several main directions, namely: *pixel intensity-based*, *feature-based*, *transformed-based* and *other approaches*.

As for cut or fade detection, *pixel intensity-based methods* provide in general the best invariance to image fluctuations and noise, providing some of the most accurate results. For instance, an approach is to use *image differences*, e.g. [23] inspired by cut detection methods (also one of the first approaches) uses a twin-thresholding of the distance between intensities of consecutive frames. For a gradual transition, accumulated should be greater than a certain threshold while for consecutive frames should stay underneath a second threshold (inferior to the first one). This approach address however gradual transitions in general. A similar approach is the one proposed in [12] that uses instead of frame differences the Accumulation of Histogram Differences (AHD) and support points. Authors report better performance compared to the classic twin-thresholding approach.

Another approach is to use the *mathematical definition* of a dissolve, i.e. a dissolve sequence is a linear combination of the two shot intensities via two monotonic linear functions, one monotonically increasing (i.e. a fade-out) and one monotonically decreasing (i.e. a fade-in) [14]. A dissolve is therefore the superposition of a fade-out and a fade-in sequence and the mean and variance of pixel intensities should have a linear and quadratic behavior respectively (method details are presented

in the next section). For instance [20] uses the assumption that the first order derivative of pixel variance should monotonically increasing from a negative value to a positive one.

Other approaches rely on the *optical effect* produced with dissolves, i.e. the two fading sequences, e.g. [19] detects the amount of pixels whose intensity is either monotonously increasing or decreasing in an certain observation window (the minimum size of a dissolve). Dissolves are detected if this measure is higher than a certain threshold which is determined on a statistically basis (the method is detailed in Section 2). A more recent approach is reported in [24] where the normalized variance auto-focus function is employed to detect dissolve candidates as "high-low-high" patterns, i.e., a monotone decreasing to a local minimum followed by a monotone increasing to a normal value. Detection is performed using simple thresholding followed by a refinement using Support Vector Machine classification.

In what concerns *feature-based detection approaches*, most of the existing approaches relay mainly on *contour/edge information*, being highly vulnerable to object/camera movement. They use the assumption that during a dissolve, object contours from the start image disappear progressively while the contours from the final image appear. For instance, [22] defines an edge change ratio (ECR) using the number of edge pixels which disappear from the current image and the number of edge pixels which appear in the next analyzed frame (see Section 2), or [13] which proposes an edge-based contrast measure (EC) emphasizing contour points contrasting the relation between weak contour points and the significant ones. Currently, other more complex approaches have been developed, e.g. [21] uses the trajectories of SURF key points (Speeded Up Robust Features) and dissolves are detected by analyzing the curve of the proportion of sub-trajectories with monotonous variation through each frame.

Transformed-based approaches are performing the detection on a transformed domain like the DCT (Discrete Cosinus Transform) or FFT (Fast Fourier Transform). For instance, in [3] a dissolve is detected whether the amount of pixel blocks showing important difference of DCT coefficients and the degree of randomness of motion vectors are above a certain threshold, or [2] which enhances

the conventional solution whereby energy histograms of the DC coefficients are used to compute the distance between consecutive frames, by amplifying transitions through the attenuation of low-pass-filtered frame distances with two sliding windows. Their main advantage is given by real-time performance capabilities, as frame reconstruction is no longer needed. However, given the current available computational power, this advantage tends to more and more accessible to the regular techniques as well.

Finally, *other approaches* attempt to exploit some other sources of information or more unconventional strategies. For instance, in [4] the structure of the video sequence is modeled through the states of a Hidden Markov Network where arcs correspond to allowable progressions of states, [15] uses Visual Rhythm Spectrum (i.e. a spatio-temporal slice model of the sequence) or [6] which uses type-2 fuzzy logic (fuzzy histograms and fuzzy co-occurrence matrix). In general methods from this category are less specialized, addressing the detection either from the generic point of view, or targeting several gradual transitions (not only the dissolves).

The remainder of the paper is organized as follows. In Section 2 we detail some of the most relevant approaches used to model the discontinuity produced by dissolves, present some practical implementations and situate our work accordingly. Section 3 describes the proposed approach. Section 4 discusses the experimental results while Section 5 concludes the paper.

2 Previous work

In this section we discuss some of the most relevant hypothesis used in the literature to model the discontinuity produced by dissolves in the video flow. Several practical implementations are also presented.

One of the most popular intensity-based dissolve model is funded on the mathematical definition of a dissolve. If $S_1(x, y, t)$ and $S_2(x, y, t)$ denote the intensities of two different shot sequences (where (x, y) are the spatial coordinates and t the temporal dimension), then the dissolve sequence $D(x, y, t)$

of duration T may be expresses as [14]:

$$D(x,y,t) = f_1(t) \cdot S_1(x,y,t) + f_2(t) \cdot S_2(x,y,t) \tag{1}$$

where $t = 0, ..., T$ and $f_1()$, $f_2()$ are two monotonic functions defined as:

$$f_1(t) = 1 - \frac{t}{T}, \;\; f_2(t) = \frac{t}{T} \tag{2}$$

with $0 \leq t \leq T$. For a given t, we may compute the variance of pixel intensities [14] which leads to:

$$\sigma^2\{D(x,y,t)\} = \left(1 - \frac{t}{T}\right)^2 \cdot \sigma^2\{S_1(x,y)\} + \left(\frac{t}{T}\right)^2 \cdot \sigma^2\{S_2(x,y)\} \tag{3}$$

where $S_1()$ and $S_2()$ are independent, and further, after developing parenthesis, we obtain the following relation:

$$\sigma^2\{D(x,y,t)\} = c \cdot (t-a)^2 - b \tag{4}$$

where a, b and c are time independent constants:

$$a = \frac{T \cdot \sigma^2\{S_1(x,y)\}}{\sigma^2\{S_1(x,y)\} + \sigma^2\{S_2(x,y)\}} \tag{5}$$

$$b = \frac{\sigma^2\{S_1(x,y)\} \cdot \sigma^2\{S_2(x,y)\}}{\sigma^2\{S_1(x,y)\} + \sigma^2\{S_2(x,y)\}} \tag{6}$$

$$c = \frac{\sigma^2\{S_1(x,y)\} + \sigma^2\{S_2(x,y)\}}{T^2} \tag{7}$$

Therefore, one may observe that during a dissolve, the variance of pixel intensities should have a parabolic behavior with respect to time. This can be emphasized by employing first and second order derivatives. For example, a practical implementation of this hypothesis is to be found in [20]. In this case, dissolves are detected whenever the first order derivative of pixel variance monotonically increases from a negative value to a positive one. Tested on two hours of video footage the method reported a recall and precision of 82.2% and 75.1%, respectively.

Another relevant approach is to model the optical effect produced by a dissolve sequence as the superposition of a fade-out with a fade-in sequence. Starting from eq. 1 we may assess the difference between two dissolve frames as:

$$D(x,y,t+1) - D(x,y,t) = \frac{1}{T}\left[S_1(x,y) - S_2(x,y)\right] = \begin{cases} C_1 > 0 & S_1(x,y) > S_2(x,y) \\ C_2 < 0 & S_1(x,y) < S_2(x,y) \\ 0 & S_1(x,y) = S_2(x,y) \end{cases} \tag{8}$$

where $D()$ is the dissolve sequence, $0 \leq t \leq T$ with T the dissolve duration and C_1 and C_2 are two constants [19]. Therefore, during a dissolve, pixel intensity change is always defined as either ascending or descending. An example of using this model is the approach proposed in [19]. It measures the amount of fading-in and fading-out pixels in a certain observation window. In order to model the behavior of a dissolve, pixels are to be classified into three categories, namely: pixels whose intensity is either monotonously increasing or decreasing (N_{prop}), pixels whose intensity remains unchanged (N_{fs}) and finally, pixels whose intensity changes are not according to eq. 8 (N_{opp}, most of the pixels undergoing motion are part of this category). Further, these changes are captured with a global function, thus:

$$S_N^w = \begin{cases} \frac{N_{prop}}{N_{fs}+N_{opp}} & N > \tau \\ 0 & otherwise \end{cases} \tag{9}$$

where N is the number of pixels whose intensities changed within the observation window w and τ is a certain threshold. A dissolve is detected whether S_N^w is greater that a second threshold, statistically determined using a binomial distribution model. This method reported a maximum recall and precision of 85% and 82%, respectively.

Other approaches use feature-level information, e.g., contour/edge information, feature points, to model the dissolve. This approach is motivated by the fact that in practice, during a dissolve, scene objects are to gradually disapear/apear. The basic idea remains the same but is applied to edges. During a dissolve new intensity edges should appear far from the locations of old edges, while old edges disappear far from the location of new edges. One of the first approaches using this model is the one in [22] that proposes to measure the visual discontinuity with the following measure denoted Edge Change Ratio:

$$\rho_{t,t+1} = max\{\rho_{in}, \rho_{out}\} \tag{10}$$

where ρ_{in} denotes the fraction of edge pixels in the frame at the moment $t+1$ which are entering the scene, while ρ_{out} is the fraction of edge pixels in the frame at the moment t which are exiting the scene. At the beginning of a dissolve, ρ_{out} should be predominant while at the end ρ_{in}. The

highest peak is obtained for the dissolve middle frame. Tested on several sequences the method achieved very good detection, however precision and recall have not been reported.

Regardless the features used, all of the existing dissolve detection approaches (see also Section 1) are inherently designed and validated on natural movies because of their high popularity.

In this paper we address a particular application domain, namely the artistic animated movies [1], and propose a dissolve detection method which is able to cope with the constraints of this domain [9]. The artistic animated movie industry witnessed a spectacular development and gain in popularity in the last years, becoming a major entertainment industry. There are a lot of international festivals promoting this genre, like Canada - Ottawa International Animation Festival, Portugal - CINANIMA International Animation Film Festival, France - Annecy International Animated Film Festival [1] (e.g. it involves more than 31,000 movie titles, 22,924 companies and 60,879 professionals, being the equivalent of the Cannes International Film Festival in the animated industry). Due to their distinctive creation process and contents, animated videos raise new processing challenges. Any un-natural (artificial generated) visual contents falls basically in this category of video footage.

Artistic animated movies are very different from natural movies and even cartoons in many respects [10]. First, they are created using a large variety of animation techniques: paper drawing, salt animation, 3D synthesis, puppet animation, etc. Therefore, contrary to natural movies, many animated movies are created frame by frame thus affecting the continuity of the visual flow. In the dissolve context, this may rend inefficient the general assumptions on the gradual or parabolic evolution of some intensity-based parameters, like the variance [20]. Also this affects the motion content which is usually discontinuous (e.g. stop motion). Each animated movie has a specific color palette, as colors are selected and mixed by the artists to express particular feelings, therefore there is a strong color similarity between shots. We mainly deal with fiction or highly abstract movies, rich in visual effects. Usually, events don't follow any physical rules: characters appear/dissapear, they can take any shape, color, etc. Contour/edge information changes often from one image to another and exiting/entering contour pixels may not necessarily be related to dissolve transitions [22]. These

particularities are synthesized in Figure 1. Overall, we deal with very complex visual contents and particularly, dissolve transitions usually show atypical patterns (see the examples in Figure 7).

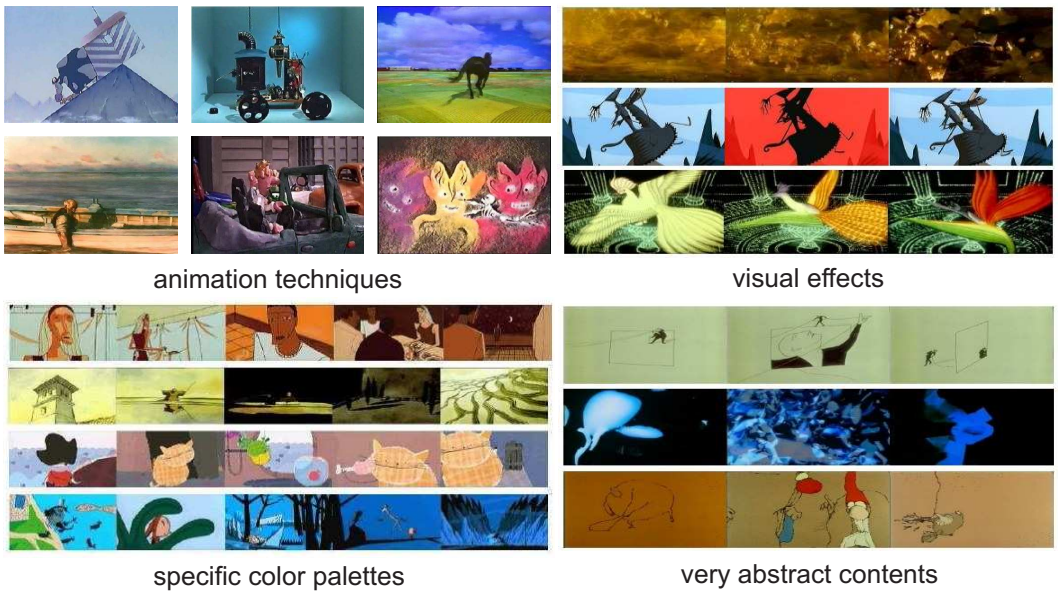


Figure 1: Specificity of the animated movie domain. First figure depicts examples of various animation techniques (from left to right and top to bottom): paper drawing, object animation, 3D synthesis, glass painting, plasticine modeling and color salts. Movies from Annecy International Animated Film Festival [1].

The interest in dissolve detection in animated movies has a double motivation. First, similar to natural movies, there is the analysis purpose, e.g. understanding the temporal structure. Second, as we deal with artistic contents, there is a content description purpose. With animated movies gradual transitions have a well defined meaning in the movie's narration. High amounts of gradual transitions are related to a specific movie contents, for instance many artistic animated movies basically replace cuts with gradual transitions, which confers mystery to the movie (see movies "Paradise", "Cœur de Secours", "Le Moine et le Poisson" [1] [11]).

To address these constraints, we propose a straightforward efficient dissolve detection that exploits the hypothesis advanced in [19] (see Section 2) according to which the pixel intensity in terms of amount of fading-out and fading-in pixels should be high during dissolves. The main novelty of our method is in the way we carry out the localization of the dissolves within the discontinuity function. Instead of just applying a global threshold, as most of the existing approaches do, we use

a twin-thresholding approach and the shape analysis of the signal. This approach allows to reduce false detections caused by steep intensity fluctuations (due to noise, movement, visual effects, etc.), as well as to retrieve dissolves caught up in other visual effects or scene movements (very frequent in animated movies). Additionally, to overcome the restraint visual continuity of the animated movies, fading-out and fading-in pixels are selected at intensity level from a reduced time window of only several frames. This work is an extension of the paper presented in [9].

3 Dissolve detection

The diagram of the proposed dissolve detection is presented in Figure 2. For the detection, we use only pixel intensity information which is obtained with the Y luminance component after converting initial RGB images to $YCbCr$ color space [16]. Additionally, to reduce computational load, images are down-sampled to a lower resolution (e.g. around 120×90 pixels). Tests proved that whether the detection results are similar with the ones obtained using higher resolution frames, the gain in computational time is significant (see Section 4).

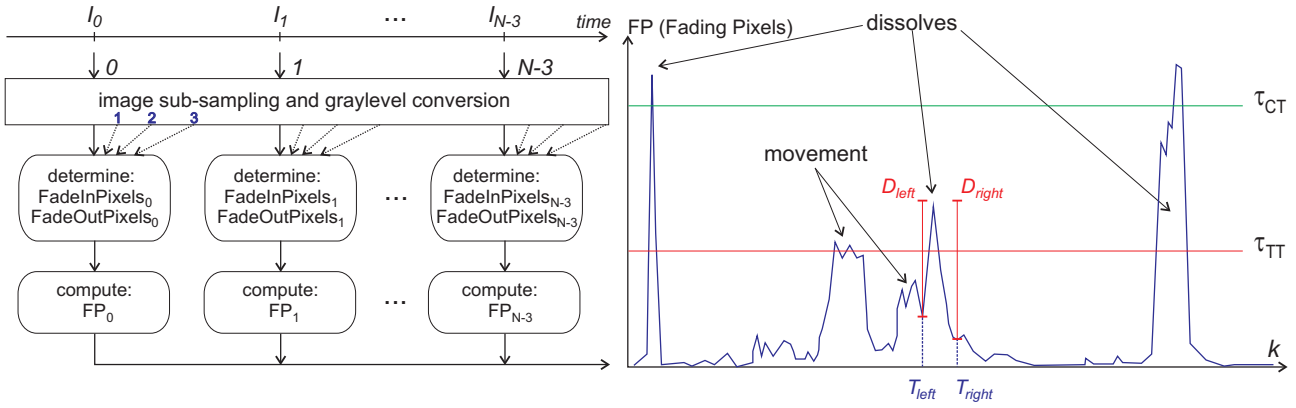


Figure 2: Diagram of the proposed dissolve detection (I_k is the frame at time index k, N is the sequence length, FP represent the fading pixels, $1, 2, 3$ denote frames from time window w). Left image exemplifies the twin-thresholding mechanism used for the detection.

For each analyzed frame at time index k, denoted I_k, we first determine the number of fading-out pixels (denoted FOP_k), i.e. pixels whose intensity decreases during next w frames, and fading-in pixels (denoted FIP_k) whose intensity increased during previous w frames. Due to the reduced

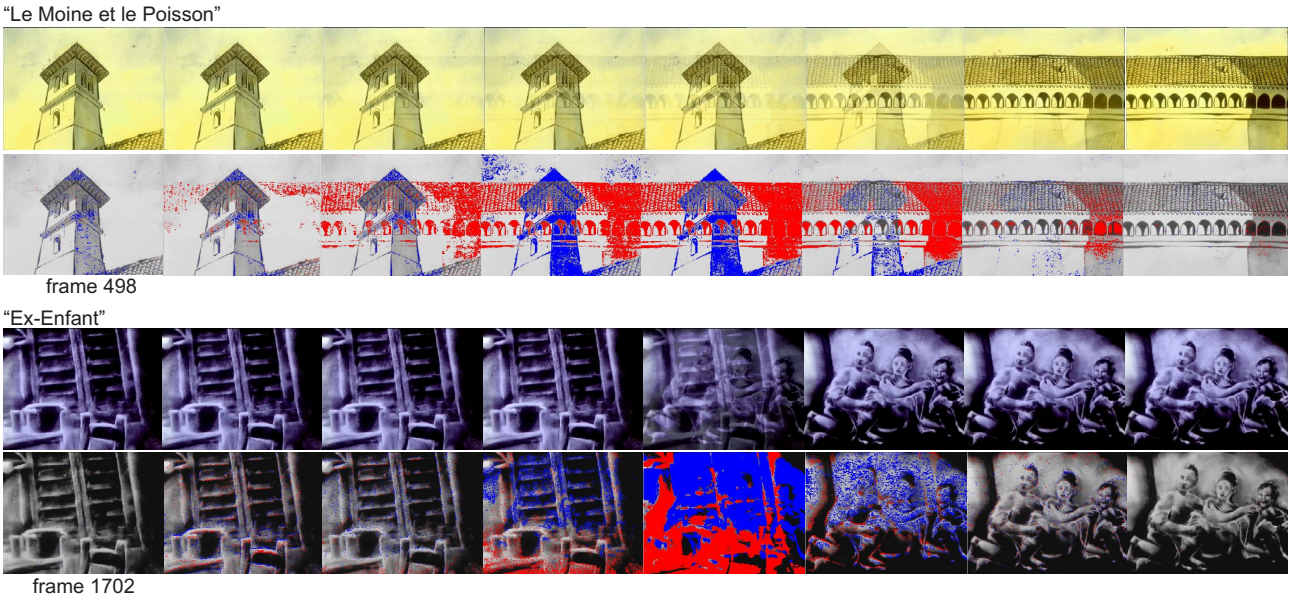


Figure 3: The amount of fading-in (FIP, depicted with Red) and fading-out pixels (FOP, depicted in Blue) during dissolve transitions. For each movie, the first line of images corresponds to the original frames while the second one to the corresponding pixel intensities (movies from CITIA [1], ”Le Moine et le Poisson” uses a gouache water-based painting technique while in ”Ex-Enfant” all visual objects are cast by light shadows). The oX is the temporal axis.

visual continuity of animated movies, we have restricted the search window to only a few frames. The selection of w is discussed at the end of this section.

In Figure 3 we have illustrated the two proportions of fading pixels for two example of dissolve transitions. One may observe that in spite of the color resemblance of the two shots, e.g. in ”Le Moine et le Poisson” the two shots have similar intensities of yellow-based color distributions while in ”Ex-Enfant” of black and indigo, the proposed measure still captures the fading process. At the beginning of the dissolve there are more FOP than FIP. As the first image starts disappearing, the number of FOP increases but also FIP as the final image starts to appear. Both ratios reach their maximum for the middle frame of the dissolve. Finally, as the final image emerges more and more, FIP become more predominant than FOP, which finally disappear in the end.

In animated movies the constant presence of displacements/movements or of color effects make this process to be likely unbalanced, i.e. proportions of FOP and FIP are not equal during dissolve (e.g. this is slightly visible in Figure 3 in the second example from movie ”Ex-Enfant”). Therefore,

instead of monitoring high values of FOP and FIP, independently, we determine a normalized visual discontinuity function by taking a simple ratio of the two values, thus:

$$FP_k = \frac{(FOP_k + FIP_k)}{H \cdot W} \tag{11}$$

where $H \cdot W$ is the image size. Defined in this way, as stated before, ideally, FP should reach its maximum for the dissolve middle frame (when both shot images are as much as visible).

The main novelty of our approach lies however in the way we carry out the localization of a dissolve within the FP function. We propose the following twin-thresholding approach (the process is depicted in Figure 2):

Case I: if FP_k for the current frame I_k is greater than a first threshold, denoted τ_{CT} (Certain Threshold), I_k is very likely to be the middle frame of the dissolve, being characterized by, both, high values of FOP_k and FIP_k respectively. If this value is a local maximum (both, previous and next values are decreasing), then a dissolve is declared in the time interval $[k - t_{max}/2; k + t_{max}/2]$, where t_{max} is an average estimate of a maximum dissolve length.

Case II: on the other hand, if FP_k for image I_k is greater than a second threshold, denoted τ_{TT} (Tolerant Threshold), but still beneath τ_{CT}, then the image is considered to be a potential dissolve middle frame. Further validation is to be performed and consists mainly on the shape analysis of FP values in the neighborhood of the frame I_k.

In the case of a dissolve, values of FP_k should decrease, both, on the positive and negative time axis. Therefore, we seek for the time moments $T_{left} < k$ and $T_{right} > k$, when FP_k starts increasing again, thus:

$$FP_{T_{left}} < FP_{T_{left}-1} \ \wedge \ FP_{T_{right}} < FP_{T_{right}+1} \tag{12}$$

(see Figure 2).

To quantify the relevance of FP_k with respect to neighbor values, we compute the height of the

peak on both sides, thus:

$$D_{left} = |FP_k - FP_{T_{left}}|, \quad D_{right} = |FP_k - FP_{T_{right}}| \tag{13}$$

Similar to the previous case, we decide that a dissolve occurred in the time interval $[k - t_{max}/2; k + t_{max}/2]$ if the distance values are greater then a fraction of FP_k, that is:

$$D_{left} > 0.5 \cdot FP_k \ \wedge \ D_{right} > 0.5 \cdot FP_k \tag{14}$$

In this way, we assure that FP_k is a local maximum, significant enough compared to neighbor values and which has and increase on both sides of at least 50%, compared to local neighbor minimum.

Intensity fluctuations may also result in several representative peaks of the FP_k function during the same dissolve. Therefore, we may by mistake select multiple frames as dissolve middle frames within same transition. To avoid this situation, the final step consists on fusing close overlapping dissolves.

3.1 Parameter tuning

The proposed method involves the choice of several threshold values. The most important are the value of w (the time window on which we assess the number of fading-in and fading-out pixels, FP), τ_{CT} (Certain Threshold - above which FP triggers the detection of a dissolve) and τ_{TT} (Tolerant Threshold - above which FP may potentially reveal a dissolve and additional verification of signal shape is performed; see Figure 2). Thresholds have been empirically determined after the manual analysis of several representative dissolve examples for various animation techniques.

For the selection of τ_{CT} and τ_{TT} we use a global approach. Similarly as for cut detection in animated movies [8], we take advantage of the fact that animated movies in general share similar color properties [11], also specific to the domain (see Figure 1). Therefore global threshold values may suffer little changes from one movie to another. This is also visible in Figure 6 where we display FP for several movies. The two thresholds were determined empirically after the manual analysis of several representative animated movies. A common setup is to take τ_{CT} around 0.4 and

τ_{TT} around 0.1. Increasing these values will result in reducing the false detections but also diminish the good detections while decreasing the values will do the opposite, thus increasing the number of good detections but in the same time the number of false detections.

To determine the optimal value for w we have conducted a preliminary test on movie "Le Moine et le Poisson" that contains 61 dissolve transitions. Due to the reduced visual continuity of animated movies, we expect that w should be of only a few frames. We performed the dissolve detection for w varying from 1 to 5. Figure 4 plots the number of good detections and false detections achieved for different values of w.

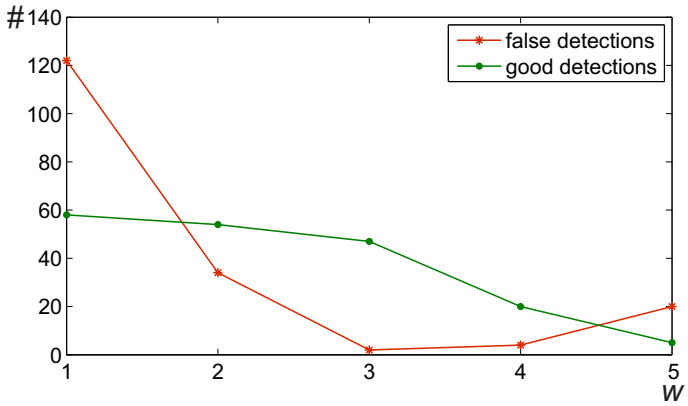


Figure 4: Number of good and false dissolve detections against different sizes of the time window w (movie "Le Moine et le Poisson" [1]).

One may observe that a very reduced value of w will result in a high number of false detections while increasing w will decrease significantly the good detections. Increasing more the value of w (above 5) will result in both very low accuracy and a high number of false detections. The best compromise would be to take $w = 3$ where good and false detections are both optimal which is the value used in our experiments.

The validation of our approach is presented in the following section.

4 Experimental results

To test our approach we have selected movies created with a high diversity of animated techniques that fall in two categories of contents:

- highly complex (abstract, very complex visual contents, motion discontinuity - denoted $\uparrow$),
- average complexity (average amount of visual effects, motion content less discontinuous - denoted $\leftrightarrow$).

Movies are presented in Figure 5[1]. The test data set consists of 61 minutes of video footage and contains a total number of 452 dissolve transitions.

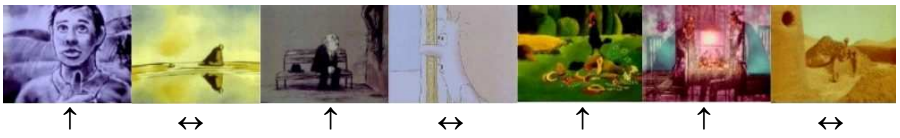


Figure 5: Test data set (movies from CITIA [1], from left to right: "Ex-Enfant", "Le Moine et le Poisson", "M. Pascal", "Une Bonne Journée", "Paradise", "Cœur de Secours" and "The Sand Castle").

To assess performance we use classic precision and recall:

$$P = \frac{GD}{GD + FD}, \; R = \frac{GD}{GD + ND} \tag{15}$$

where GD is the number of good detections (true positives), FD is the number of false detections (false positive) and ND is the number of non-detections (false negative), where $GD + ND = 452$ (i.e. the total number of dissolves). The detection results are summarized with Table 1.

Overall, we score 360 good detections and a very reduced number of false detections, i.e. only 23 (for most of the sequences < 2). This leads to an average precision of 94% and a recall of 79.6%. At sequence level, precision and recall ranges from $[86.3; 100]\%$ and $[70.2; 100]\%$, respectively. The

[1]movies are available for free preview or for purchasing at CITIA (`http://www.citia.info/`) or on YouTube `http://www.youtube.com/`).

Table 1: Dissolve detection results.

movie	count	GD	FD	P	R
"Ex-Enfant"↑	75	65	8	89%	86.7%
"Le Moine et le Poisson"↔	61	47	2	95.9%	77.1%
"M. Pascal"↑	98	76	2	97.4%	77.6%
"Une Bonne Journée"↔	19	19	0	**100%**	**100%**
"Paradise"↑	60	44	7	86.3%	73.3%
"Cœur de Secours"↑	67	47	2	95.9%	70.2%
"The Sand Castle"↔	72	62	2	96.9%	86.1%

highest detection ratio is obtained for movie "Une Bonne Journée" which has a more accessible content ($P = R = 100\%$), while the lowest detection ratio is obtained for the movie "Paradise" due to its very complex content ($P = 86.26\%$, $R = 73.33\%$).

We attempted to compare our approach against other reference methods from the literature. We use two confirmed approaches, namely: the assessment of the variance of pixel intensities, which during dissolves should yield a quadratic behavior [20], and the use of Edge Change Ratio [22] for which edges were obtained with a Canny edge detector with automatic thresholding [5]. Methods are presented in Section 2. The experimental results proved that classic methods tend to be rather inefficient when used on this particular type of video contents.

Figure 6 exemplifies the three approaches for several representative movies. Due to the discontinuous nature of the motion content and to the presence of visual effects, variance of pixel intensities do not follow a parabolic shape. Instead, it has an unpredictable behavior (see the Green line in Figure 6, e.g. movie C) or unexpectedly decreasing or increasing during dissolves (e.g. movies A, B or D). On the other hand, contour information (i.e. ECR), whether for some particular cases it provides good discrimination (similar to FP, see in Figure 6 the Black line for movie C), in general is either non-discriminant (see movie A where ECR has small values during dissolves) or highly sensitive to noise and visual changes (see movies B and D where important peaks are due not to dissolves but to noise, fading effects and important changes in object structure).

On the other hand, the proposed method provides good results in all situations (see the good detections in Figure 6). Thanks to the shape analysis, which adapts to local contents, it is discrim-

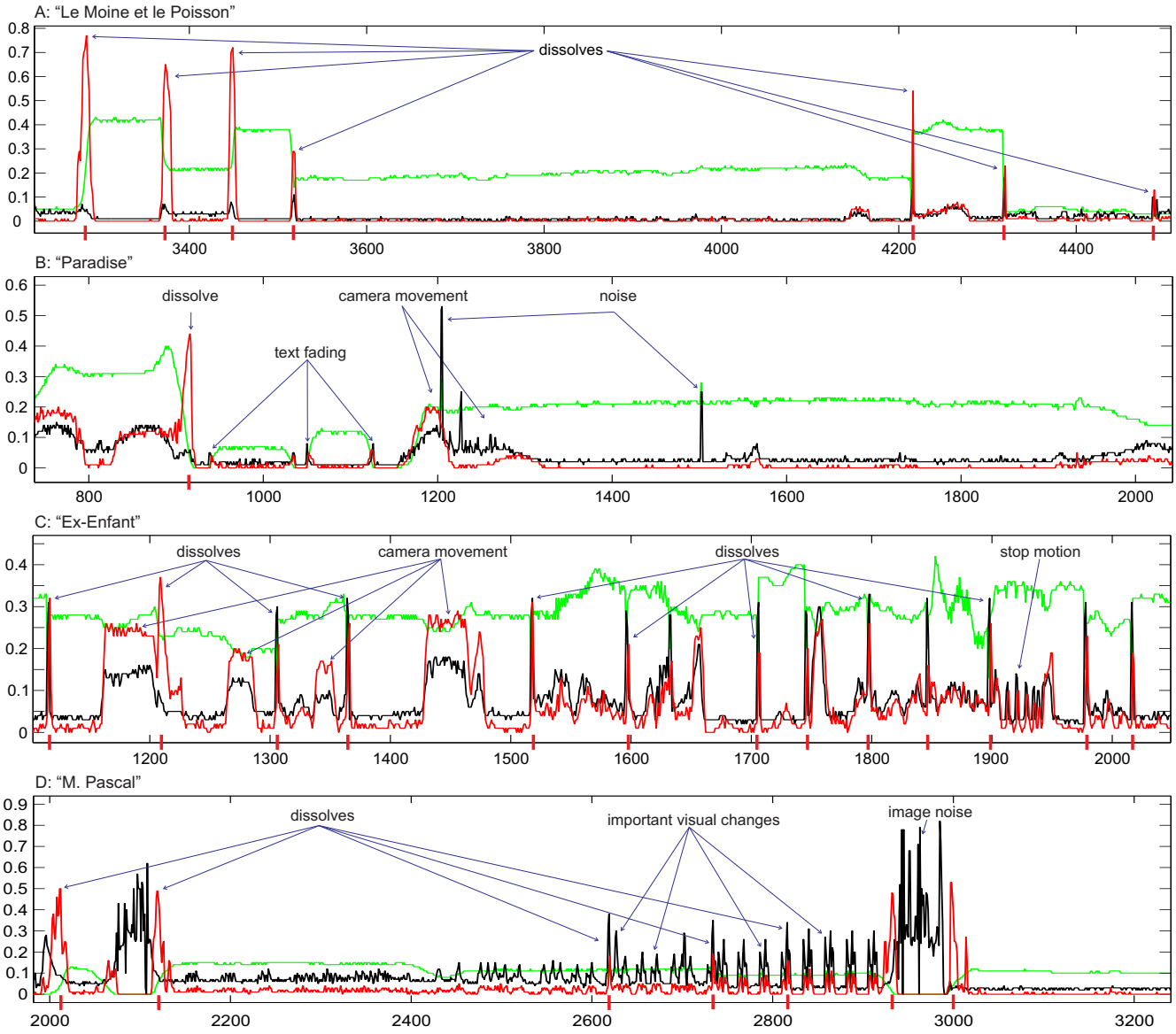


Figure 6: The proposed discontinuity function FP_k (in Red) against intensity variance (in Green, values are scaled to fit the other functions) and Edge Change Ratio (in Black). Dissolves which were successfully detected are marked on the temporal axis (oX) with vertical Red segments (graphs were deliberately slightly shifted on the oY axis with respect to 0 for visualization purpose; Video 1, QuickTime, MOV, 24MB, http://imag.pub.ro/∼bionescu/index_files/DemoDissolves.wmv).

inant enough to retrieve dissolves even when mixed-up with motion (see the Red line in Figure 6, e.g. the first detected dissolve in movie B or the second detected dissolve in movie C which are successfully separated from camera movement) and to avoid false detections (see movie C where camera movement and other visual effects are not taken as dissolves despite their high FP values).

Also, judging from the time evolution of the FP function, a global thresholding strategy, like the one proposed in [19] (see Section 2), is less efficient with most of the movies. This is due to the fact that most dissolves are either merged with other visual changes or have comparable amplitudes with other variations (e.g. see the Red line for movie C or D in Figure 6). In general, movies with a highly complex visual contents (denoted with $\uparrow$) tend to have similar behavior of the FP function such as for movie C (for reason for brevity, we limit to the presentation of only four examples in Figure 6).

In Figure 7 we present several examples of complex dissolves which were successfully detected regardless their atypical patterns: important global motion (movie E and F where the background is continuously shifting during dissolve), similar color intensities and structure for the start and end image of the dissolve (movie C), highly discontinuous content (movies A, C and D), very short dissolves (~ 3 frames, movies A and D). In what concerns the detection errors, most of the non-detections are due to very complex scene changes which makes impossible to provide separation for the dissolves. Less frequent are the false detections (see Table 1), which are occurring mostly due to visual changes with dissolve-like signatures. A typical pattern is an object/camera movement followed by a shot change (i.e. a cut) and again by motion.

The achieved results are very promising considering the difficulty of the test sequences or even compared to existing approaches, which applied to natural movies achieve average detection ratios around 80% (see Section 2).

Finally, the proposed approach provides also good computational performance. Table 2 presents the results obtained on a regular laptop computer, CPU Intel(R) Core(TM) i5 M460@2.53GHz, 4GB of RAM running on Microsoft Windows 7 - 64 bit[2]. The presented processing time includes also the delays caused by the visual interface, as images are displayed as being processed (application developed under Borland C++ Builder 6).

[2]for calculations we consider a frame rate of 25 frames per second.

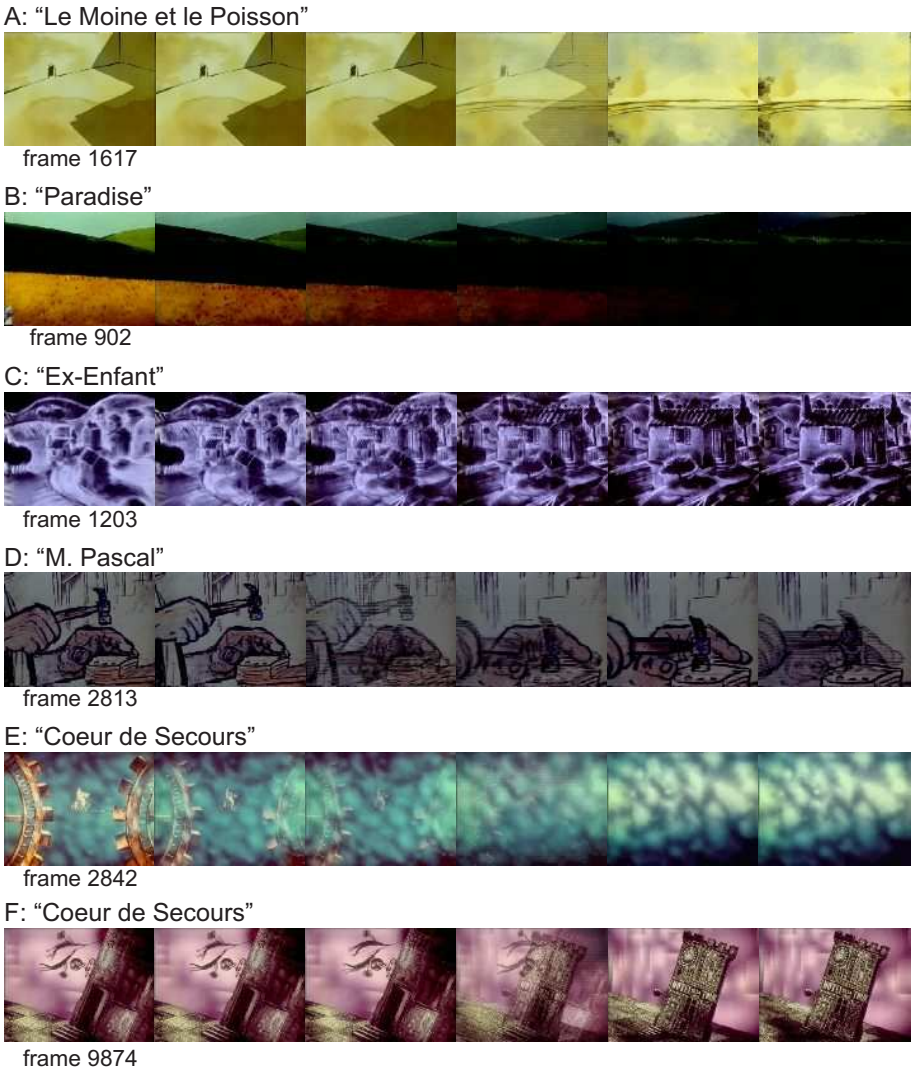


Figure 7: Example of complex or atypical dissolve transitions which were successfully detected with the proposed approach (movies from CITIA [1]). The oX axis is the temporal axis.

Table 2: Dissolve detection processing time.

image size (pixels)	frames/s	time processing 10 min.
60×45	134	112s
120×90	128	117s
240×180	94	160s
480×360	72	208s
740×480 (original)	43	349s

For instance, at a frame resolution of 120×90 pixels it achieves more than 128 frames per second (5 times faster than real time). Compared to using original frame resolution, it is three times faster (for the later we achieve only 43 frames per second). Nevertheless, even in this case, the detection performs almost twice faster than real time. In terms of detection errors, we obtain results very

similar at all scales, therefore, by reducing the image size we increase the performance efficiency.

5 Conclusion

We proposed a dissolve detection approach that addresses the specificity of the animated videos (e.g. many animated movies are created frame by frame thus affecting the continuity of the visual and motion flow, each animated movie has a specific color palette and therefore there is a strong color similarity between shots, and so on). The proposed method exploits pixel intensities in terms of amount of fading-out and fading-in pixels. The main novelty of our method is in the way we carry out the localization of the dissolves within the discontinuity function. We use a twin-thresholding mechanism and the shape analysis of the signal.

Experimental tests conducted on more than 452 dissolve transitions show the potential of this approach in cases where traditional methods (adapted to natural movies) tend to fail. It allows to reduce false detections caused by steep intensity fluctuations (due to noise, movement, visual effects, etc.), as well as to retrieve dissolves caught up in other visual effects or scene movements (very frequent in animated movies) leading to precision and recall ratios up to 100% (producing a very low number of false alarms). In terms of computational complexity, the proposed method performs five times faster than real time on a regular computer.

The method seems to be less efficient when dealing with some very complex scene changes and fade transitions that involve camera movement and special effects. Another limitation is the impossibility of determining the exact dissolve boundaries. Future work will mainly consists on addressing this limitation by investigating the behavior of various features in the neighborhood of the starting and ending frames of a dissolve.

Acknowledgements

This work has been supported under the Sectoral Operational Programme Human Resources Development 2007-2013 of the Romanian Ministry of Labour, Family and Social Protection through the Financial Agreement POSDRU/89/1.5/S/62557. The authors would like to thank CITIA - The City of Moving Images and Folimage Animation Company for providing them with access to their animated movie database and for their support.

References

[1] Citia - city of moving images, http://www.citia.info/.

[2] O. Bao, M. Lian, and L. Guan. Enhancement of dissolved shot boundary detection with twin-windows amplification method. *SPIE Optical Engineering*, 46(12), 2007.

[3] G. Boccignone, M. De Santo, and G. Percannella. Automated threshold selection for the detection of dissolves in mpeg video. In *IEEE International Conference on Multimedia and Expo*, pages 1535–1538, 2000.

[4] J.S. Boreczky and L.D. Wilcox. A hidden markov model framework for video segmentation using audion and image features. In *IEEE International Conference on Acoustics, Speech, and Signal Processing; Seattle, USA*, 1998.

[5] J. Canny. A computational approach to edge detection. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

[6] S. Das, S. Sural, and A. K. Majumdar. Detection of hard cuts and gradual transitions from video using fuzzy logic. *International Journal of Artificial Intelligence and Soft Computing*, 1(1):77–98, 2008.

[7] B. Han, X. Gao, and H. Ji. A unified framework for shot boundary detection. *Springer LNCS Pattern Recognition*, 3801:997–1002, 2005.

[8] B. Ionescu, V. Buzuloiu, P. Lambert, and D. Coquin. Improved cut detection for the segmentation of animation movies. In *IEEE International Conference on Acoustic, Speech and Signal Processing; Toulouse, France*, 2006.

[9] B. Ionescu, V. Buzuloiu, P. Lambert, and D. Coquin. Dissolve detection in abstract video contents. In *IEEE International Conference on Acoustic, Speech and Signal Processing; Prague, Czech Republic*, 2011.

[10] B. Ionescu, L. Ott, P. Lambert, D. Coquin, A. Pacureanu, and V. Buzuloiu. Tackling action - based video abstraction of animated movies for video browsing. *SPIE - Journal of Electronic Imaging*, 19(3), 2010.

[11] B. Ionescu, C. Vertan, P. Lambert, and A. Benoit. A color-action perceptual approach to the classification of animated movies. In *International Conference on Multimedia Retrieval; Trento, Italy*, 2011.

[12] Q.-G. Ji, J.-W. Feng, J. Zhao, and Z.-M. Lu. Effective dissolve detection based on accumulating histogram difference and the support point. In *International Conference on Pervasive Computing Signal Processing and Applications*, pages 273–276, 2010.

[13] R. Lienhart. Comparison of automatic shot boundary detection algorithms. In *SPIE Storage and Retrieval for Still Image and Video Databases VII*, pages 290–301, 1999.

[14] R. Lienhart. Reliable transition detection in videos: A survey and practitioner's guide. *International Journal of Image and Graphics*, 1(3):469–486, 2001.

[15] S.J. Park, K.-D. Seo, J.-G. Kim, and S.M.-H. Song. Automatic dissolve detection scheme based on visual rhythm spectrum. *LNCS Advances in Mulitmedia Information Processing*, pages 787–798, 2005.

[16] C. Patrick and V. Gary. In John D. Eds. Owens, I.-Jong Lin, Yu-Jin Zhang, and Giordano B. Beretta, editors, *Parallel Processing for Imaging Applications*, volume 7872, pages 78720D–78720D–9. SPIE, 2011.

[17] A. F. Smeaton, P. Over, and W. Kraaij. High-level feature detection from video in trecvid: a 5-year retrospective of achievements. In *Multimedia Content Analysis, Theory and Applications*, pages 151–174. Springer Verlag, ISBN 978-0-387-76567-9, Berlin, 2009.

[18] Cees G.M. Snoek and M. Worring. Multimodal video indexing: A review of the state-of-the-art. *Multimedia Tools and Applications*, 25(1):5–35, 2005.

[19] C.W. Su, H.-Y.M. Liao, H.R. Tyan, K.C. Fan, and L.H. Chen. A motion-tolerant dissolve detection algorithm. *IEEE Transactions on Multimedia*, 7(6):1106–1113, 2005.

[20] B.T. Truong, C. Dorai, and S. Venkatesh. New enhancements to cut, fade, and dissolve detection processes in video segmentation. In *ACM Multimedia; Los Angeles, CA, USA*, pages 219–227, 2000.

[21] Y. Wang, Y. Yang, T. Ren, and G. Wu. A motion-insensitive dissolve detection method with surf. In *International Conference on Image and Graphics; Xi'an, Shanxi, China*, pages 290–301, 2009.

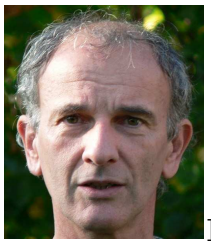
[22] R. Zabih, J. Miller, and K. Mai. A feature-based algorithm for detecting and classification production effects. *Multimedia Systems*, 7:119–128, 1999.

[23] H. Zhang, A. Kankanhalli, and S.W. Smoliar. Automatic partitioning of full-motion video. *Multimedia Systems*, 1(1):10–28, 1993.

[24] W. Zhang, C. Liu, Q. Huang, S. Jiang, and W. Gao. Coarse-to-fine dissolve detection based on image quality assessment. *Springer book The Era of Interactive Media, ISBN: 978-1-4614-3500-6*, pages 277–287, 2013.

Biographies

Bogdan Ionescu is currently a Lecturer with University "Politehnica" of Bucharest-Romania. He holds a B.S. degree in applied electronics (2002) and an M.S. degree in computing systems (2003), both from University Politehnica of Bucharest. He also holds a Ph.D. degree in image processing (2007) from, both, the University of Savoie and University "Politehnica" of Bucharest. He is a collaborator of the University of Savoie and University of Trento. His scientific interests cover video processing, video retrieval, computer vision, software engineering, and computer science. He is a Member of IEEE, SPIE, ACM, EURASIP and GDR-ISIS.



Patrick Lambert received the enginer degree in electrical engineering in 1978, and the PhD degree in signal processing in 1983, both from the National Polytechnic Institute of Grenoble, France. He is currently a Full Professor at the School of Engineering of University of Savoie, Annecy, France and a member of the Informatics, Systems, Information and Knowledge Processing Laboratory (LISTIC), Annecy, France. His research interests are in the field of image and video analysis, and actually dedicated to non linear color filtering and video semantic indexing.