Taylor & Francis
Taylor & Francis Group

Applied Artificial Intelligence
An International Journal

ISSN: 0883-9514 (Print) 1087-6545 (Online) Journal homepage: https://www.tandfonline.com/loi/uaai20

Novel Shot Boundary Detection in News Streams Based on Fuzzy Petri Nets

Shu-Hung Yang, Yi-Nan Lin, Gwo-Jen Chiou, Ming-Kuen Chen, Victor R. L. Shen & Hsin-Yi Tseng

To cite this article: Shu-Hung Yang, Yi-Nan Lin, Gwo-Jen Chiou, Ming-Kuen Chen, Victor R. L. Shen & Hsin-Yi Tseng (2019) Novel Shot Boundary Detection in News Streams Based on Fuzzy Petri Nets, Applied Artificial Intelligence, 33:12, 1035-1057, DOI: 10.1080/08839514.2019.1661118

To link to this article: https://doi.org/10.1080/08839514.2019.1661118

APPLIED ARTIFICIAL INTELLIGENCE
2019, VOL. 33, NO. 12, 1035–1057
https://doi.org/10.1080/08839514.2019.1661118

Novel Shot Boundary Detection in News Streams Based on Fuzzy Petri Nets

Shu-Hung Yang[a], Yi-Nan Lin[b], Gwo-Jen Chiou[c], Ming-Kuen Chen[a], Victor R. L. Shen[d], and Hsin-Yi Tseng[d]

[a]Department of Business Management, National Taipei University of Technology, Taipei City, Taiwan; [b]Department of Electronic Engineering, Ming Chi University of Technology, New Taipei City, Taiwan; [c]Department of Electrical Engineering, National Formosa University, Yunlin County, Taiwan; [d]Department of Computer Science and Information Engineering, National Taipei University, New Taipei City, and Department of Information Mnanagement, Chaoyang University of Technology, Taichung City, Taiwan

ABSTRACT

With the advent of a digital era, people have encountered some difficulty in using and absorbing overwhelming information generated by technological advances in multimedia. Thus, the development of video summarization enables people to catch a general idea about videos in a short time. In this paper, we focus on the shot change, a part of the video summarization, to conduct an experimental sample on news programs. Moreover, a high-level fuzzy Petri net model is presented to describe the frame combination which indicates a shot boundary used for a video frame sequence in order to detect both cut transitions and gradual transitions. This study has used feature functions to estimate the direct shot change in consideration of video shot boundary detection which adopts the HLFPN model to find a threshold value. The experimental results manifest that the proposed system saves a lot of time and reduces the occurrence of improper shot changes caused by the motions of objects and cameras when comparing the proposed approach with other existing ones.

Introduction

The fast development of multimedia technology and the ever-expanding computational resources have led to a large volume of video data. Video content analysis has become a very hot topic in the past research. It is important to organize video data properly for content-based video analysis and retrieval. As a result, the events of segmenting, describing, indexing, and managing video data of different genres are pivotal. The detection of shot boundary is the basic and difficult problem in content-based video retrieval, and it also determines the retrieval results directly. The video shot boundary detection technology has drawn more attention in the past, for the purpose of making the video summarization (Bian et al. 2015).

CONTACT Victor R. L. Shen rlshen@mail.ntpu.edu.tw; victor.rlshen@msa.hinet.net Department of Computer Science and Information Engineering, National Taipei University, New Taipei City 237, Taiwan

Color versions of one or more of the figures in the article can be found online at www.tandfonline.com/uaai.

Video summarization is an approach to create a static capsule representation of a long video in terms of a set of key frames that generate a shorter abstraction of long video. The shot boundary detection is actually the first step of video segmentation. A string of some video includes the segments of normal, close-up shot or commercial films. These video segments are all those frame sequences.

With the improvement of people's living standards, the news stream as a form of life is constantly enriched. The media makers make more and more news reports on the TV programs to enhance influence on the society. The same theme of a news stream broadcasts repeatedly, and the so-called follow-up news just reedits the previous reports. We still waste much time to search the news streams and watch the news report repeatedly. For the requirements of social work, we would read the news video more effectively. Therefore, we use the news stream to be a sample on the shot boundary detection. In order to increase the effectiveness, the work load must be reduced in a long term when people are watching the news report. Also, a lot of research works have been done by other samples such as sports video (Zhang, Wu, and Li 2018), movies (Chiu, Lin, and Chang 2008), and TV advertisement (Xia and Tang 2012; Zhu et al. 2010).

The high-level fuzzy Petri net (HLFPN) model has been applied to many fields, like the music segmentation and others (Shen and Cheng 2014) (Shen 2006) (Shen, Chang, and Juang 2010) (Shen and Cao 2011). The HLFPN model is based on Petri net theory and fuzzy reasoning. Petri net theory, proposed by Dr. Carl Adam Petri in 1962, is a graphical and mathematical modeling tool, which is concurrent, asynchronous, distributed, parallel, non-deterministic, and stochastic. It can be used to model and analyze various systems (Murata 1989). However, along with the advancement of the information system, the description of a Petri net is becoming more and more complex with the use of fuzzy production rules (Shen, Li, and Jeng 2011). In general, a fuzzy production rule describes the fuzzy relationship between the antecedent and the consequent. According to the above motivation and purposes, this study focuses on the development of the main techniques including the catching of shot changes, the frame segmentation, and the classified image sequence recognition.

The remainder of this paper is organized as follows. In Section II, the literature review of shot boundary detection and the HLFPN are formulated. The techniques for detecting video segmentation and our system design approach are introduced in Section III. The experimental results and discussions of the HLFPN model are presented in Section IV. Finally, the final remarks and future work are concluded in Section V.

Literature Review

A video shot consists of continuous frame sequences captured by a single camera action. Based on the previous research work in video segment, different shot transition detections are described and explained. Then, we present some basic definitions regarding the HLFPN model. Finally, we discuss some related work.

Shot Transition Classes

According to whether the transition between two shots is abrupt or gradual, the shot change can be categorized into two types, namely, abrupt shot change and gradual transition (Zhi and Cai 2005). The abrupt shot change means a direct connection of two different shots while the gradual transition has an editing effect between two different shots. The gradual transition can be further classified into flash light, abrupt changes cuts, fade in/out, dissolve, exposure, and wipe (Zhi and Cai 2005), according to the characteristics of the different editing effects. Shot boundary detection, also known as temporal video segmentation, is the process of identifying the transitions between the adjacent shots in Table 1.

Shot Boundary Detection

When changing scenes in the video, the color difference will form between two adjacent frames due to a variation of the video sets. Therefore, by calculating the color distribution of every frame in the video and analyzing the changes in the frames, we are able to configure the exact position and range of each shot. The focus of this study is on utilizing the difference between two adjacent frames with color, brightness, and continuity for analysis and positioning of the shot detection.

During the broadcasting of a news stream on the same video, the camera is set by placing firmly, moving in parallel, or spinning frames for stability and comfort. In addition, for synchronizing with movement, the parallel and spinning shots are used. By doing so, the adjacent frames will have little changes in content. When the camera is moving slightly, the adjacent picture element will

Table 1. Shot transition class.

Flash light	A lens at the end of one scene within another scene changes with time and the lens extends, and see the slow changes from darkness to light.
Abrupt changes cuts	The most basic type of shot transition, the cut is the most common way to join two shots. In essence, it is the continuation of two different shots within the same time and space.
Fade in/out	A fade occurs when the picture gradually turns to be a single color, usually black, or when a picture gradually appears on screen.
Dissolve	A dissolve involves gradually changing the visibility of the picture. However, rather than transitioning from a shot to a color, a dissolve occurs when one shot is changed into another shot gradually.
Exposure	Simply stated, exposure is the amount of light it takes to record a scene correctly.
Wipe	One shot is progressively replaced by another shot in a geometric pattern. There are many types of wipe, from straight lines to complex shapes.

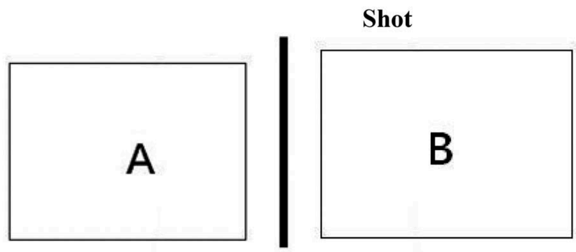

Figure 1. Connection of two adjacent frames called a shot or a cut.

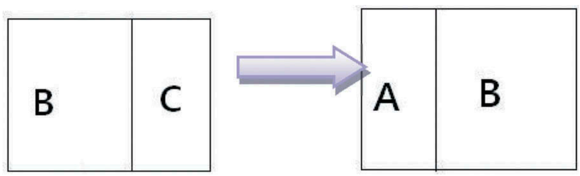

Figure 2. Changes between two adjacent frames.

move in the opposite direction of a camera. The previous frame can be split into two areas, namely, area *A* and area *B*. The connection of two adjacent frames is called a *shot* or a *cut* as shown in Figure 1. When the camera spins from the left to the right, area *A* will move out of the frame, area *B* will move left, and area *C* will come in on the right. On the other hand, when the shot suddenly changes, the adjacent frames are almost never the same as in Figure 2. These phenomena will cause every picture element to change positions.

Threshold Value Setting

This study addressed three feature functions in relation to the threshold value. The frame of the video in flash light will be changed within another scene due to the changes in brightness of the frame. The frame-based detection method was proposed using gray-scale histogram difference, which is used to make a comparison.

Several questions were raised in the video in connection with the flash light and abrupt changes cut. In this study, we adopt a feature function of zero mean difference between two adjacent frames, in order to determine the boundary of shot detection.

In order to avoid a faulty shot in the partial changes, in consequence, five consecutive sheets are set as a range of frames to calculate the zero mean difference. This study uses a feature function to estimate the direct shot change in consideration of video shot boundary detection. We adopt the HLFPN model to find a threshold value.

High-Level Fuzzy Petri Net

Scholars conduct their researches with evolutionary Petri net theories one after another, such as colored Petri nets (Robidoux et al. 2010), timed Petri nets (Moradkhani and Bigham 2017), fuzzy Petri nets (Balabanian, Da Silva, and Pedrini 2017), high-level fuzzy Petri nets (Shen and Cheng 2014) (Shen 2006) (Shen, Chang, and Juang 2010), and so on. This paper adopts the HLFPN model to make a decision. It provides the characters of Petri net theory and fuzzy logic theory, which can be used to express fuzzy production rules and conduct fuzzy reasoning with fuzzy production rules. The basic definitions and fuzzy reasoning approach are presented as follows:

1) Definitions,

- **Definition** 1: The HLFPN is defined as an 8-tuple.

$$HLFPN = (P, T, F, C, V, \alpha, \beta, \delta)$$

where

$P = \{p_1, \ldots, p_k\}$	A finite set of places.
$T = \{t_1, \ldots, t_l\}$	A finite set of transitions.
$P \cup T \neq \oslash$	
$F \subseteq (P\times T) \cup (T \times P)$	Called the flow relation and is also a finite set of arcs, each representing the fuzzy set (i.e. fuzzy term) for an antecedent or a consequent; where the positive arcs (i.e. THEN parts) are denoted by →.
$C = \{X, Y, Z\}$	A finite set of linguistic variables, e.g. X, Y, and Z, where $X = \{x_1, x_2, \ldots, x_n\}$, $Y = \{y_1, y_2, \ldots, y_m\}$, $Z = \{z_1, z_2, \ldots, z_q\}$.
$V = \{v_1, \ldots, v_l\}$	A finite set of fuzzy truth values known as the fuzzy relational matrix between the antecedent and the consequent of a rule.
$\alpha: P \rightarrow C$	An association function, mapping from places to linguistic variables. $\alpha(p_i) = \{c_i\}$, $i = 1, \ldots, l$, where $C = \{c_i\}$ is a set of linguistic variables in the knowledge base (KB) and is the number of linguistic variables in the KB.
$\beta: F \rightarrow [0, 1]$	An association function, mapping from the flow relations to the fuzzy truth values between zero and one.
$\delta: T \rightarrow V$	An association function, mapping from transitions to fuzzy relational matrices.

- **Definition** 2: Input and Output Functions

$I(t) = \{p \in P \vert (p, t) \in F\}$	A set of input places of transition t.
$I(p) = \{t \in T \mid (t, p) \in F\}$	A set of input transitions of place p.
$O(t) = \{p \in P \mid (t, p) \in F\}$	A set of output places of transition t.
$O(p) = \{t \in T \mid (p, t) \in F\}$	A set of output transitions of place p.

- **Definition** 3: Membership Function

The mapping function $Mem(p)$: $P \rightarrow [0,1]$ assigns each place a real value, where $Mem(p) = \mathrm{DOM}(\alpha(p))$, DOM represents the degree of membership in the associated proposition and data tokens are available in the set P of places.

- **Definition** 4: Max-Min Compositional Rule

In the HLFPN, $\forall$ transition t, $V(t)$ = min(fuzzy sets in I (t)); $\forall$ place p, $V(p)$ = max(fuzzy sets in I(p)). The Max-Min composition operator is denoted by $\circ$.

- **Definition** 5: Input Place, Hidden Place, and Output Place

In the HLFPN, $\forall$ place $p_i \in P$, if $\forall t_j \in T$, $p_i \notin O(t_j)$, then p_i is called an input place(IP) of t_j. If $\forall t_j \in T$, $p_i \notin$ I(t_j), then p_i is called an output place(OP) of t_j; otherwise, p_i is called a hidden place.

2) Fuzzy Reasoning

Let R be a set of fuzzy production rules, where $R = \{R_1, R_2, \ldots, R_n\}$. The general form of the i_{th} fuzzy production rule R_i is shown as follows:

R_i: IF d_j(X is A), THEN d_k (Y is B); ELSE, d_w (Z is C) … (V).

where "X is A", "Y is B" and "Z is C" are propositions; X is called the input linguistic variable; Y and Z are called the output linguistic variables, respectively; A is called the input fuzzy set; B and C are called the output fuzzy sets, respectively; the fuzzy truth values of the propositions "X is A", "Y is B" and "Z is C" are restricted to [0, 1]; "X is A" is the antecedent of the fuzzy production rule R_i; "Y is B" and "Z is C" are the consequents of the fuzzy production rule R_i; Let V represent the fuzzy relational matrix between the antecedent and the consequent of a fuzzy production rule.

- **Example 1**:

Let us consider the fuzzy production rule R_1 shown as follows:

R_1: IF it(X_1) is hot(A_1) AND the sky(X_2) is cloudy(A_2), THEN the humidity(Y) is high(B).

Based on the transformation procedure presented in Shen (2000), we can transform the above fuzzy production rule R_1 into the following first-order logic form:

R_1': IF $X_1(A_1)$ AND $X_2(A_2)$, THEN $Y(B)$.

Then, the HLFPN model is shown in Figure 3.

Assume that the fuzzy sets A_1, A_2, and B are shown as follows:

$$A_1 = 0.25/a_{11} + 0.60/a_{12} + 0.26/a_{13}$$

$$A_2 = 0.15/a_{21} + 0.80/a_{22} + 0.37/a_{23}$$

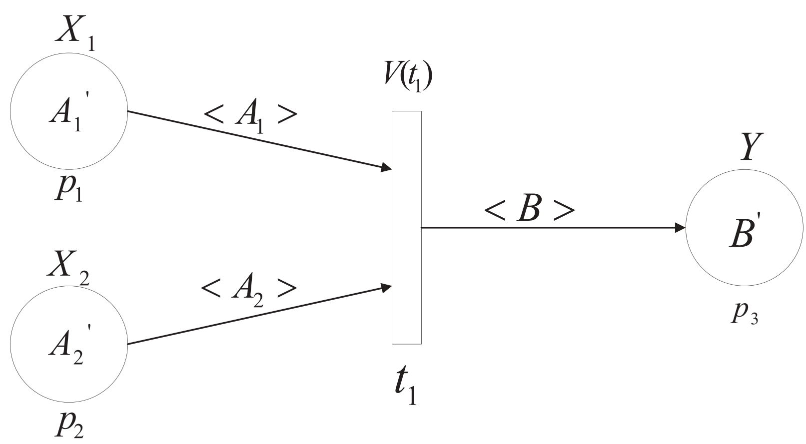

Figure 3. HLFPN for example 1.

$$B = 0.30/b_1 + 0.79/b_2 + 0.18/b_3$$

By the cylindrical extension operations (Shen 2006), we can obtain the antecedent fuzzy set A, shown as follows:

$$\begin{aligned} A = A_1 \times A_2 &= (0.25\ \ 0.60\ \ 0.26)^T \wedge (0.15\ \ 0.80\ \ 0.37) \\ &= \begin{vmatrix} 0.15 & 0.25 & 0.25 \\ 0.15 & 0.60 & 0.37 \\ 0.15 & 0.26 & 0.26 \end{vmatrix} \end{aligned}$$

Then, the fuzzy relational matrices $V_1(t_1)$, $V_2(t_2)$, and $V_3(t_3)$ between the antecedent and the consequent of fuzzy production rule R_1 can be obtained, shown as follows:

$$V_1(t_1) = \begin{vmatrix} 0.15 & 0.25 & 0.25 \\ 0.15 & 0.30 & 0.30 \\ 0.15 & 0.26 & 0.26 \end{vmatrix} \quad \in A \times B \times b_1$$

$$V_2(t_2) = \begin{vmatrix} 0.15 & 0.25 & 0.25 \\ 0.15 & 0.60 & 0.37 \\ 0.15 & 0.26 & 0.26 \end{vmatrix} \quad \in A \times B \times b_2$$

$$V_3(t_3) = \begin{vmatrix} 0.15 & 0.18 & 0.18 \\ 0.15 & 0.18 & 0.18 \\ 0.15 & 0.18 & 0.18 \end{vmatrix} \quad \in A \times B \times b_3$$

The most widely used fuzzy reasoning method is the max–min composition inference (Moradkhani and Bigham 2017). Assume that the input fuzzy sets A_1' and A_2' are shown as follows:

$$A_1{}' = 0.12/a_{11}' + 0.82/a_{12}' + 0.27/a_{13}'$$

$$A_2{}' = 0.27/a'_{21} + 0.94/a'_{22} + 0.38/a'_{23}$$

Then, we can get

$$A_1{}' \circ \mathrm{V}_1(\mathrm{t}_1) = (0.12 \;\; 0.82 \;\; 0.27) \circ \mathrm{V}_1(\mathrm{t}_1) = (0.15 \;\; 0.30 \;\; 0.30)$$

$$A_1{}' \circ V_2(t_1) = (0.12 \;\; 0.82 \;\; 0.27) \circ V_2(t_1) = (0.15 \;\; 0.60 \;\; 0.37)$$

$$A_1{}' \circ V_3(t_1) = (0.12 \;\; 0.82 \;\; 0.27) \circ V_3(t_1) = (0.15 \;\; 0.18 \;\; 0.18)$$

Finally, we can obtain

$$\begin{aligned} B' &= (0.27 \;\; 0.94 \;\; 0.38) \circ \begin{vmatrix} 0.15 & 0.15 & 0.15 \\ 0.30 & 0.60 & 0.18 \\ 0.30 & 0.37 & 0.18 \end{vmatrix} \\ &= (0.30 \;\; 0.60 \;\; 0.18) \\ &= 0.30/b'_1 + 0.60/b'_2 + 0.18/b'_3 \end{aligned}$$

The above description is the fuzzy reasoning process of an HLFPN model.

3) Fuzzy Reasoning Algorithm

In this sub-section, we briefly review a fuzzy reasoning algorithm (FRA) from Shen and Cheng (2014) to determine whether there exists a fuzzy relational matrix between the antecedent and the consequent of a fuzzy production rule.

INPUT: Mem(p_i) $\forall$ $p_i \in IP$, where IP denotes a set of input places. OUTPUT: Mem(p_i) $\forall$ $p_i \in OP$, where OP denotes a set of output places.

- *Step* 1:

Initially, assume that the Degree of Memberships (DOMs) in the propositions operating on input variables is available. Consequently, the initial marking function is shown as follows: $M(p_i) = 0$, if $p_i \notin IP$; $M(p_i)$ = the number of data tokens, if $p_i \in IP$.

- *Step* 2:

$\forall t_j \in T$, compute $V(t_j) = W_a \times W_c = (w_{a1}\ w_{a2}, \ldots, w_{am})^{\mathrm{T}} \wedge (w_{c1}, w_{c2}, \ldots, w_{cn})$, where T denotes a set of transitions; $V(t_j)$ is a fuzzy relational matrix between the antecedent and the consequent of rule t_j; $W_a = \{w_{a1}, w_{a2}, \ldots, w_{am}\}$ is a fuzzy set of weights in the antecedent; $W_c = \{w_{c1}, w_{c2}, \ldots, w_{cn}\}$ is a fuzzy set of weights in the consequent; and each element of a fuzzy set is denoted by a fuzzy weight interval.

- *Step* 3:

Input a data pattern W_a-input.

- *Step* 4:

Fire the enabled transitions. Let t_j be any enabled transition. Then, compute $t_j \in T/\forall p_k$, $M(p_k)$ = the number of data tokens, $W'_a = W_a$-input, $W'_c = W'_a \circ\ V(t_j)$ or $W'_a \circ\ V(t_j)$ if an ELSE part is available.

- *Step* 5:

For every output variable O, its associated membership distribution is $W'_c = \{w'_{ci}\} = \vee\ w'_{ci}$, $i = 1, 2, \ldots, I$, where I is the in degree of output variable O. Then, W'_c becomes an actual output.

- *Step* 6:

Go back to *Step* 4, while $\exists\ t_j \in T/M(p_i) = 1\ \forall\ p_i \in I(t_j)$.
(That is, while the enabled transitions still exist, go to *Step* 4.)

- *Step* 7:

The weighted average defuzzification method is applied, and the real operating value is computed.

- *Step* 8:

The end.

Related Work

In the video, multiple cameras are situated in different angles and positions, and are used to change a shot. In order to find the key frame in a video, we must first catch a frame of the shot for recognition and classification. It is a challenge to detect shot changes because there are lots of close-ups, slow-motion replays, and date displays in a video set. Many other studies directly deal with the colors of picture elements in a frame, using split-second changes to locate the shot (Tsai and Chen 2005) (Liu and Yang 2003) (Britsom, Bronselaer, and Tre 2015) (Pratama et al. 2015). But using this kind of feature functions for differentiation might not be able to yield the results we need. For example, under the circumstances such as fast-moving objects or flashes (Yang et al. 2005), Zhu and other people (Chiu, Lin, and Chang 2008) used color histograms to solve those problems concerning flash and also proposed the time-limited algorithms in order to fix the color differentiation caused by camera movement or film positioning.

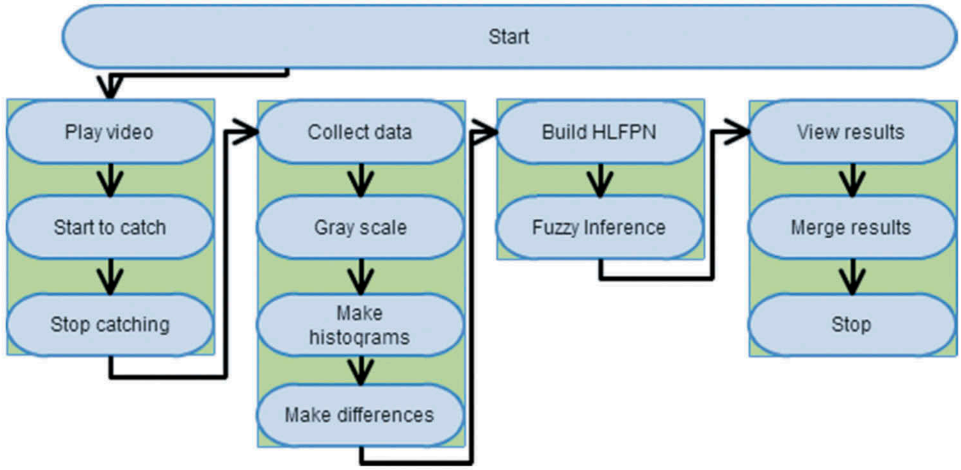

Figure 4. Block diagram of the proposed approach.

The Proposed Approach

In this section, how to apply the feature functions and the HLFPN model to video shot boundary detection are illustrated. As usual, we first need to decide the shot detection process. A threshold value is determined to check whether a cut is needed or not.

Shot Boundary Detection and Feature Extraction

This section explains the functions for the feature extraction in the shot detection. The study uses three feature functions to define the adaptive threshold value.

(1) *Feature Function 1*: Gray-Scale Histogram Difference

Assume that the video consists of n frames. The video segmentation of frame size is $r \times c$. Let i be the number of frames in the histogram. A number of gray scales can be denoted as $H_i(k)$, $0 \leqq k \leqq 255$. The gray scale histogram difference between two frames is defined as follows:

$$Pi = \sum_{k=0}^{255} (Hi(k) - Hi + 1(k)),^2 i = 1, \cdots, n - 1 \tag{1}$$

(2) *Feature Function 2*: Zero Mean Difference in Three Consecutive Frames

Assume that the zero mean of the gray scale for three consecutive frames is denoted as TP_i, then it is defined as follows:

$$TP_i = \frac{1}{3}\sum_{j=i}^{i+2} P_j \tag{2}$$

(3) *Feature Function 3*: Zero Mean Difference in Five Consecutive Frames

Assume that the zero mean difference of the gray scale for five consecutive frames is denoted as FP_i, then it is defined as follows:

$$FP_i = \frac{1}{5}\sum_{j=i}^{i+4} P_j \tag{3}$$

Definition of Membership Functions

In the decision method, we use three technical indices, namely, the gray scale histogram difference (P_i), the zero mean difference (TP_i), and the zero mean difference in five consecutives (FP_i). As a result, the membership functions of *Low, Middle*, and *High* for each index are defined. Then, three sets of technical indices include the S-Type, the Λ-Type and the Z-Type membership functions, as shown in Figure 5.

In addition, the decision is also divided into three conditions, namely, "*Cut*", "*Undecided*", and "*Uncut*". Therefore, two sets of cut and uncut belong to the Λ-Type of membership function, as shown in Figure 6.

All of them are defined in Table 2. The membership degrees of the decision for video segmentation are listed in Table 3.

In the analysis, the values of input parameter are usually set between 0 and 1. However, due to the membership function defined above, the scope of variable domain is set between 0 and 1. Thus, the values of input parameter must be

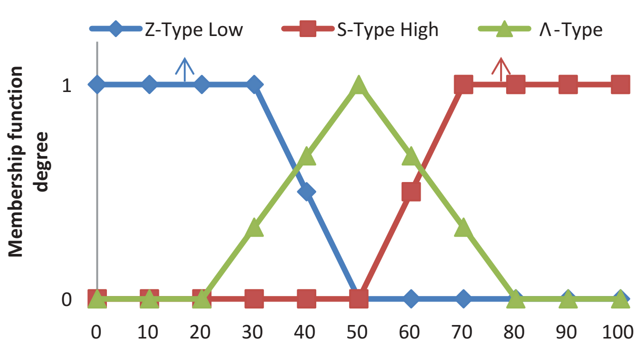

Figure 5. The type of membership functions for technical indices.

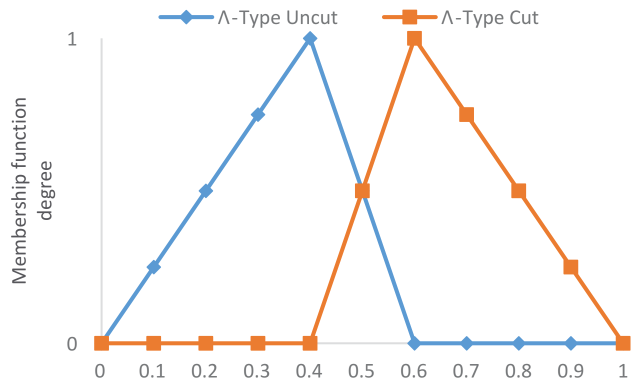

Figure 6. The type of membership functions for technical decision.

Table 2. Membership degrees of input parameter.

Input Parameter	*Low*			*Middle*		*High*	
Pi	10	30	20	50	80	70	90
TPi	10	30	20	50	80	70	90
FPi	10	30	20	50	80	70	90

Table 3. Membership degrees of technical decision.

Uncut		Undecide		Cut	
0	0.4	0.4	0.6	0.6	1

converted to the values between 0 and 1 before they are substituted into the rule base for calculation.

$$\mu_H(x) = \begin{Bmatrix} 1, & x \geq 90 \\ \frac{1}{20}(x-70), & 70 < x < 90 \\ 0, & x \leq 70 \end{Bmatrix} \tag{4}$$

$$\mu_M(x) = \begin{Bmatrix} 0, & x \geq 80 \\ \frac{-1}{30}(x-80), & 50 \leq x < 80 \\ \frac{1}{30}(x-20), & 20 < x < 50 \\ 0, & x \leq 20 \end{Bmatrix} \tag{5}$$

$$\mu_L(x) = \begin{Bmatrix} 0, & x \geq 30 \\ \frac{-1}{20}(x-30), & 10 < x < 30 \\ 1, & x \leq 10 \end{Bmatrix} \tag{6}$$

$$\mu_C(x) = \begin{Bmatrix} 1, & x \geq 1 \\ \frac{1}{0.4}(x-1), & 0.6 < x < 1 \\ 0, & x \leq 0.6 \end{Bmatrix} \tag{7}$$

$$\mu_{UD}(x) = \left\{ \begin{array}{cc} 0, & x \geq 0.6 \\ \frac{1}{0.2}(x - 0.6), & 0.4 < x < 0.6 \\ 0, & x \leq 0.4 \end{array} \right\} \tag{8}$$

$$\mu_{UC}(x) = \left\{ \begin{array}{cc} 0, & x \geq 0.4 \\ \frac{-1}{0.4}(x - 0.4), & 0 < x < 0.4 \\ 1, & x \leq 0 \end{array} \right\} \tag{9}$$

Fuzzy Reasoning and Building HLFPN

According to the fuzzy sets and their corresponding membership functions defined previously, each technical index is input to a fuzzifier and the technical index to get the membership degree of each set is calculated. Based on the size of a technical index, it is transformed to an 「*If* … *then* … 」 statement in order to establish a fuzzy production rule.

Assume that P_i, TP_i and FP_i are input linguistic variables with fuzzy terms: high (H), middle (M), and low (L); and assume that the decision (D) is an output linguistic variable with fuzzy terms: strong (S), indeterminate (I), and weak (WK). The fuzzy production rules are defined as follows:

R_1: If P_i is H, then D is S.
R_2: If P_i is M, then D is I.
R_3: If P_i is L, then D is WK.
R_4: If TP_i is H, then D is S.
R_5: If TP_i is M, then D is I.
R_6: If TP_i is L, then D is WK.
R_7: If FP_i is H, then D is S.
R_8: If FP_i is M, then D is I.
R_9: If FP_i is L, then D is WK.

Then, the fuzzy production rules are transformed into the HLFPN model as shown in Figure 7, and the parameters are described in Table 4.

Main Results

In order to demonstrate that the proposed HLFPN-based decision system is feasible when using fuzzy reasoning algorithm, a fuzzy reasoning algorithm (FRA) (Shen and Cheng 2014) is used to determine whether a fuzzy relational matrix between the antecedent and the consequent of a fuzzy production rule exists or not. An example is illustrated in this section to demonstrate the feasibility of fuzzy hypothesis.

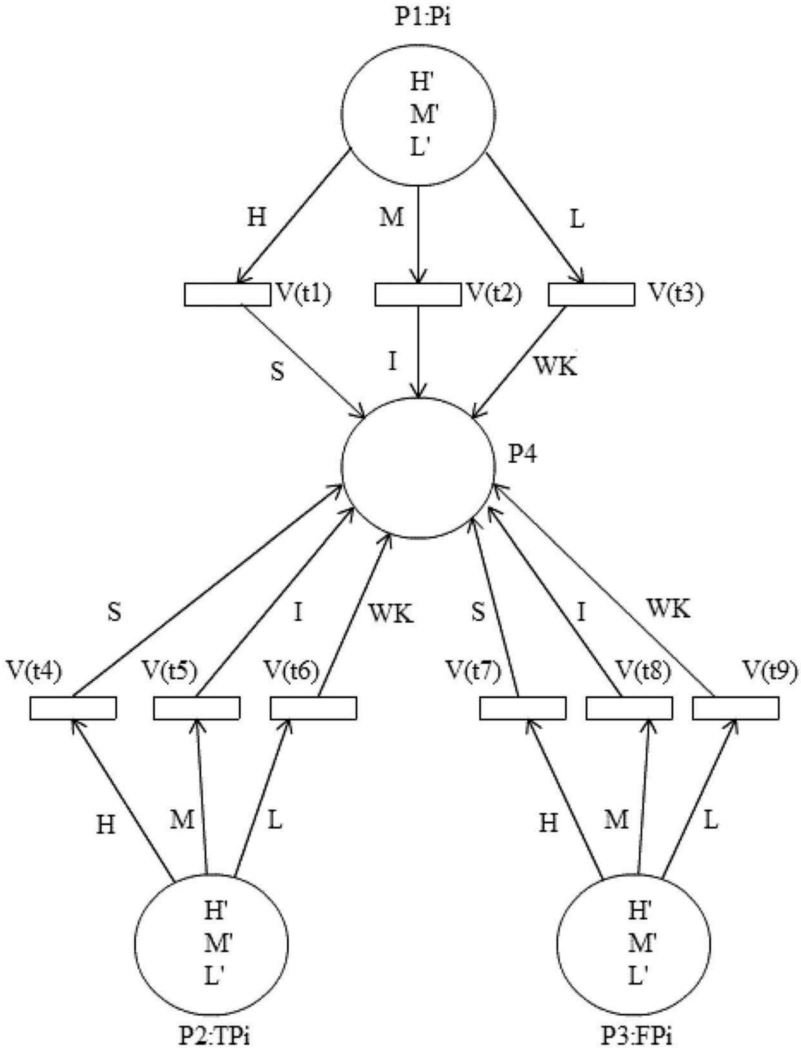

Figure 7. The HLFPN model with six fuzzy production rules.

Table 4. Description of parameters.

Name of Parameter	*Description of Parameter*
P_i, TP_i, FP_i	Denote P_i, TP_i and FP_i value, an input place, P_1.
D	Denotes decision, an output place, P_4.
H, M, L	Denote high, middle, and low fuzzy sets, respectively.
S, I, WK	Denote strong, indeterminate, and weak fuzzy sets, respectively.
$V(t_i)$, $i = 1, 2, \ldots, 9$	Denotes the fuzzy relational matrix of P_i, TP_i, FP_i and the decision.
H', M', L'	Denote high, middle and low fuzzy sets of input value.

Example of HLFPN in Fuzzy Reasoning

- [*Step* 1]

Initially, assume that only the DOMs in the propositions operating on input variables are available. Assume that six fuzzy sets are shown as follows:

$$H = \frac{0.00}{h_L} + \frac{0.00}{h_M} + \frac{0.80}{h_H}$$

$$M = \frac{0.00}{m_L} + \frac{0.50}{m_M} + \frac{0.00}{m_H}$$

$$L = \frac{0.20}{l_L} + \frac{0.00}{l_M} + \frac{0.00}{l_H}$$

$$S = \frac{0.00}{s_L} + \frac{0.00}{s_M} + \frac{0.80}{s_H}$$

$$I = \frac{0.00}{I_L} + \frac{0.50}{I_M} + \frac{0.00}{I_H}$$

$$WK = \frac{0.20}{b_L} + \frac{0.00}{b_M} + \frac{0.00}{b_H}$$

- [*Step* 2]

By cylindrical extension, we obtain the antecedent fuzzy relational matrices shown as follows:

$$\begin{aligned} V(t1) &= H * S \\ &= (0.00\ 0.00\ 0.80)T \wedge (0.00\ 0.00\ 0.80) \\ &= \begin{bmatrix} 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.80 \end{bmatrix} \end{aligned}$$

$$\begin{aligned} V(t4) = &\ V(t1) \\ = &\begin{bmatrix} 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.80 \end{bmatrix} \end{aligned}$$

$$\begin{aligned} V(t7) = &\ V(t1) \\ = &\begin{bmatrix} 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.80 \end{bmatrix} \end{aligned}$$

$$V(t2) = M * I = (0.00\ 0.50\ 0.00)T \wedge (0.00\ 0.50\ 0.00) = \begin{bmatrix} 0.00 & 0.00 & 0.00 \\ 0.00 & 0.50 & 0.00 \\ 0.00 & 0.00 & 0.00 \end{bmatrix}$$

$$V(t5) = V(t2) = \begin{bmatrix} 0.00 & 0.00 & 0.00 \\ 0.00 & 0.50 & 0.00 \\ 0.00 & 0.00 & 0.00 \end{bmatrix}$$

$$V(t8) = V(t2) = \begin{bmatrix} 0.00 & 0.00 & 0.00 \\ 0.00 & 0.50 & 0.00 \\ 0.00 & 0.00 & 0.00 \end{bmatrix}$$

$$V(t3) = L * WK = (0.20\ 0.00\ 0.00)T \wedge (0.20\ 0.00\ 0.00) = \begin{bmatrix} 0.20 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \end{bmatrix}$$

$$V(t6) = V(t3) = \begin{bmatrix} 0.20 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \end{bmatrix}$$

$$V(t9) = V(t3) = \begin{bmatrix} 0.20 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \end{bmatrix}$$

- [*Step* 3]

Input the data pattern, shown as follows:

$$H'Pi = \frac{0.00}{h'_L} + \frac{0.00}{h'_M} + \frac{1.00}{h'_H}$$

$$M'Pi = \frac{0.00}{m'_L} + \frac{0.00}{m'_M} + \frac{0.00}{m'_H}$$

$$L'Pi = \frac{0.00}{l'_L} + \frac{0.00}{l'_M} + \frac{0.00}{l'_H}$$

$$H'TPi = \frac{0.00}{h'_L} + \frac{0.00}{h'_M} + \frac{0.00}{h'_H}$$

$$M'TPi = \frac{0.00}{m'_L} + \frac{0.53}{m'_M} + \frac{0.00}{m'_H}$$

$$L'TPi = \frac{0.00}{l'_L} + \frac{0.00}{l'_M} + \frac{0.00}{l'_H}$$

$$H'FPi = \frac{0.00}{h'_L} + \frac{0.00}{h'_M} + \frac{0.00}{h'_H}$$

$$M'FPi = \frac{0.00}{m'_L} + \frac{0.07}{m'_M} + \frac{0.00}{m'_H}$$

$$L'FPi = \frac{0.40}{l'_L} + \frac{0.00}{l'_M} + \frac{0.00}{l'_H}$$

- [*Step* 4]

Fire the enabled transitions:

$$S'Pi = H'Pi \circ V(t_1) = [0.00 \quad 0.00 \quad 1.00] \circ \begin{bmatrix} 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.80 \end{bmatrix} = [0.00 \quad 0.00 \quad 0.80]$$

$$I'Pi = M'Pi \circ V(t_2) = [0.00 \quad 0.00 \quad 0.00] \circ \begin{bmatrix} 0.00 & 0.00 & 0.00 \\ 0.00 & 0.50 & 0.00 \\ 0.00 & 0.00 & 0.00 \end{bmatrix} = [0.00 \quad 0.00 \quad 0.00]$$

$$WK'Pi = L'Pi \circ V(t3) = [0.00 \quad 0.00 \quad 0.00] \circ \begin{bmatrix} 0.20 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \end{bmatrix} = [0.00 \quad 0.00 \quad 0.00]$$

$$S'TPi = H'TPi \circ V(t4) = [0.00 \quad 0.00 \quad 0.00] \circ \begin{bmatrix} 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.80 \end{bmatrix} = [0.00 \quad 0.00 \quad 0.00]$$

$$I'TPi = M'TPi \circ V(t5) = [0.00 \quad 0.53 \quad 0.00] \circ \begin{bmatrix} 0.00 & 0.00 & 0.00 \\ 0.00 & 0.50 & 0.00 \\ 0.00 & 0.00 & 0.00 \end{bmatrix}$$
$$= [0.00 \quad 0.50 \quad 0.00]$$

$$WK'TPi = L'TPi \circ V(t6) = [0.00 \quad 0.00 \quad 0.00] \circ \begin{bmatrix} 0.20 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \end{bmatrix}$$
$$= [0.00 \quad 0.00 \quad 0.00]$$

$$S'FPi = H'FPi \circ V(t7) = [0.00 \quad 0.00 \quad 0.00] \circ \begin{bmatrix} 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.80 \end{bmatrix}$$

$$= [0.00 \quad 0.00 \quad 0.00]$$

$$I'FPi = M'FPi \circ V(t8) = [0.00 \quad 0.07 \quad 0.00] \circ \begin{bmatrix} 0.00 & 0.00 & 0.00 \\ 0.00 & 0.50 & 0.00 \\ 0.00 & 0.00 & 0.00 \end{bmatrix}$$
$$= [0.00 \quad 0.07 \quad 0.00]$$

$$WK'FPi = L'FPi \circ V(t9) = [0.40 \quad 0.00 \quad 0.00] \circ \begin{bmatrix} 0.20 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \end{bmatrix}$$
$$= [0.20 \quad 0.00 \quad 0.00]$$

- [*Steps* 5–6]

Finally, the fuzzy reasoning result is:

$$D = S'_{Pi} \cup I'_{Pi} \cup WK'_{Pi} \cup S'_{TPi} \cup I'_{TPi} \cup WK'_{TPi} \cup S'_{FPi} \cup I'_{FPi} \cup WK'_{FPi}$$
$$= \frac{0.20}{d_L} + \frac{0.50}{d_M} + \frac{0.80}{d_H}$$

- [*Steps* 7–8]

If the weighted average defuzzification method is applied, then the real operation value of "Decision" can be computed, shown as follows:

$$\begin{aligned} Decision &= \frac{(0.20 * d_L + 0.50 * d_M + 0.80 * d_H)}{0.20 + 0.50 + 0.80} \\ &= \frac{(0.20 * 0.2 + 0.50 * 0.5 + 0.80 * 0.8)}{1.5} \\ &= 0.62 \end{aligned}$$

Due to 0.6 ≤ Decision ≤ 1.0, the decision is made to be "Cut".

Experimental Results

To compare retrieval effectiveness, both the traditional histogram-based and the HLFPN-based image retrieval methods have been implemented. The retrieval effectiveness is measured using the precision and the recall (Karasulu 2018). The calculation is shown as follows:

$$Precision = \frac{correct\ shots}{correct\ shots + incorrect\ shots} \tag{10}$$

$$Recall = \frac{correct\ shots}{correct\ shots + missing\ shots} \tag{11}$$

Precision is the ratio of the number of correctly detected shot changes to total number of correctly and incorrectly detected shot changes. Similarly, recall is the ratio of the number of correctly detected shot changes to total number of actual shot changes including correct shots and missing shots. The precision measures the retrieval accuracy, while the recall measures the sensitivity of retrieving relevant images. The higher the above two values are, the better the shot boundary detection will become.

In our experiment, the performance of the shot boundary detection is evaluated on a database of nine video sets. The database contains a total of 10624 frames, 223 abrupt cuts, and 30 gradual transitions, the details of which are shown in Table 5.

In the experiment, we have used the Java programming language to develop the user interface for users conducting their experiments. The experimental steps are shown in Figure 8.

The user interface is shown in Figure 9.

The experimental results are shown in Table 6, from which we can see our HLFPN-based shot detection has acceptable performance.

Table 5. Experimental results.

Video	Frames	Abrupt Cuts	Gradual Transitions
Test 1	1322	25	2
Test 2	1266	11	10
Test 3	1259	15	3
Test 4	926	17	0
Test 5	1450	34	2
Test 6	995	28	5
Test 7	1275	33	2
Test 8	1094	32	4
Test 9	1037	28	2

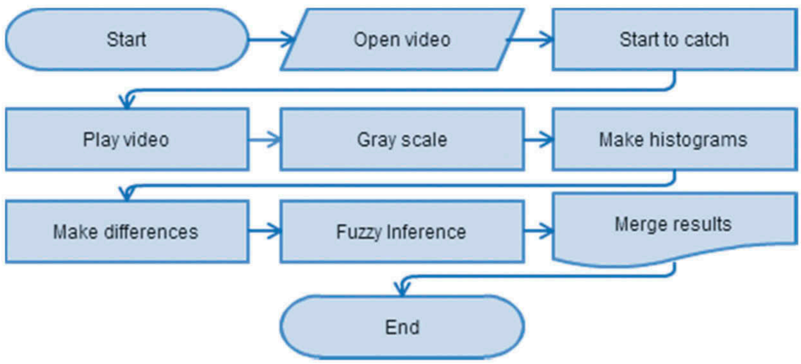

Figure 8. Experimental steps.

Figure 9. User interface.

To compare with the histograms, we have designed another histogram base to ensure that our HLFPN model performs better. The experimental results are shown in Table 7.

Table 6. The HLFPN shot detection results.

Video	Precision	Recall
Test 1	0.89	0.92
Test 2	0.78	0.78
Test 3	1	0.88
Test 4	0.89	0.80
Test 5	0.94	0.89
Test 6	0.93	0.87
Test 7	0.94	0.89
Test 8	0.94	0.88
Test 9	0.93	0.87

Table 7. Comparison between HLFPN and histogram.

	HLFPN		Histogram	
Video	Precision	Recall	Precision	Recall
Test 1	0.89	0.92	0.60	0.80
Test 2	0.78	0.78	0.55	0.78
Test 3	1	0.88	0.78	0.88
Test 4	0.89	0.80	0.68	0.80
Test 5	0.94	0.89	0.77	0.80
Test 6	0.93	0.87	0.82	0.87
Test 7	0.94	0.89	0.70	0.80
Test 8	0.94	0.88	0.66	0.80
Test 9	0.93	0.87	0.66	0.80
Average	0.92	0.86	0.69	0.81

Conclusion

In this study, we have designed a Java program to implement the video shot boundary detection and used the HLFPN model to act as a feature extraction system. In our experimental results, it saves a lot of time and reduces the occurrence of improper shot changes caused by the motions of objects and cameras when comparing the proposed approach with a manual method.

The contributions of this paper include: 1) Through the reasoning algorithm of the HLFPN model, we can figure out the threshold value for the shot boundary detection. 2) The work load can be reduced when people are watching a long news report. 3) For meeting the requirements of social work, we can read the news report faster. 4) The system can automatically segment the video frames.

In the future work, we will improve this system for sports video or other kinds of video. Hopefully, our proposed system can become faster and be used in real-time applications. On the other hand, the HLFPN model can be used for the analysis and development of many other systems. The results of this study may be used in the relevant electronic equipment including smart devices.

Acknowledgments

The authors are very grateful to the anonymous reviewers for their constructive comments which have improved the quality of this paper. This work was supported by the Ministry of Science and Technology, Taiwan, under grant MOST 107- 2221- E-845- 002-MY3.

Funding

This work was supported by the Ministry of Science and Technology, Taiwan [MOST 107-2221-E-845-002-MY3].

References

Balabanian, F., E. S. Da Silva, and H. Pedrini. 2017. Image thresholding improved by global optimization methods. *Applied Artificial Intelligence* 31:197–208. Online: April 10.

Bian, J., Y. Yang, H. Zhang, and T.-S. Chua. 2015. Multimedia summarization for social events in microblog stream. *IEEE Transaction on Multimedia* 17 (2):216–28. February. doi: 10.1109/TMM.2014.2384912.

Britsom, D. V., A. Bronselaer, and G. D. Tre. 2015. Using data merging techniques for generating multidocument summarizations. *IEEE Transactions on Fuzzy Systems* 23 (3):576–92. June. doi: 10.1109/TFUZZ.2014.2317516.

Chiu, S. T., G. S. Lin, and M. K. Chang. 2008. An effective shot boundary detection algorithm for movies and sports. Proc. of the 3rd International Conference on Innovative Computing Information and Control, 173, June 18–20, Dalian, China.

Karasulu, B. 2018. An optimized image segmentation approach based on Boltzmann machine. *Applied Artificial Intelligence* 31:792–802. Online: March 1.

Liu, Q., and L. Yang. 2003. Two-difference algorithm for video's abrupt shot change detection. Proceedings of the 2003 International Conference on Neural Networks and Signal Processing, 1177–80, December, Nanjing, China.

Moradkhani, F., and B. S. Bigham. 2017. A new image mining approach for detecting micro-calcification in digital mammograms. *Applied Artificial Intelligence*:411–24. Online: October 25. doi:10.1080/08839514.2017.1378082.

Murata, T. 1989. Petri nets: properties, analysis and applications. *Proceeding of the IEEE* 77 (4):541–80. April. doi: 10.1109/5.24143.

Pratama, M., S. G. Anavatti, M. Joo, and E. D. Lughofer. 2015. pClass: an effective classifier for streaming examples. *IEEE Transactions on Fuzzy Systems* 23 (2):369–86. April. doi: 10.1109/TFUZZ.2014.2312983.

Robidoux, R., H. P. Xu, L. D. Xing, and M. C. Zhou. 2010. Automated modeling of dynamic reliability block diagrams using colored Petri nets. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans* 40 (2):337–51. March. doi: 10.1109/TSMCA.2009.2034837.

Shen, S., and J. Cao. 2011. Abrupt shot boundary detection algorithm based on fuzzy clustering neural network. Proc. of the 3rd International Conference on Computer Research and Development, 246–48, March 11–13, Shanghai, China.

Shen, V. R. L. 2000. Correctness in hierarchical knowledge-based requirements. *IEEE Transactions on Systems, Man, and Cybernetics. Part B: Cybernetics* 30 (4):625–31. August. doi: 10.1109/3477.865185.

Shen, V. R. L. 2006. Knowledge representation using high-level fuzzy Petri nets. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans* 36 (6):1220–27. November. doi: 10.1109/TSMCA.2006.878968.

Shen, V. R. L., Y. S. Chang, and T. T. Y. Juang. 2010. Supervised and unsupervised learning by using Petri nets. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans* 40 (2):363–75. March. doi: 10.1109/TSMCA.2009.2038068.

Shen, V. R. L., and H.-C. Cheng. 2014. An automatic calibration system for Chinese karaoke lyrics based on high-level fuzzy Petri nets. Proc. of the IEEE International Conference on 2014 Machine Learning and Cybernetics and 2014 International Conference on Wavelet Analysis and Pattern Recognition, Crowne Plaza Lanzhou, Lanzhou, Gansu, China, 544–49, 13–16 July.

Shen, V. R. L., W. C. Li, and J. T. Jeng. 2011. Application of support vector regression and fuzzy theory to exchange rate from new Taiwan dollars to U.S. dollars. Proc. of the IEEE International Conference on Intelligent Computing and Intelligent Systems(ICIS), Guilin, China, vol. 3, 327–31.

Tsai, T. H., and Y. C. Chen. 2005. A robust shot change detection method for content-base retrieval. Proc. of International Symposium on Circuits and Systems, 4590–93, May 23–26, Kobe, JN.

Xia, Q., and G. Tang. 2012. Research on TV advertisement detection based on video shot. Proc. of the 3rd International Conference on System Science, Engineering Design and Manufacturing Informatization, 245–48, October 20–21, Chengdu, China.

Yang, L. J., H. Lu, B. Wang, X. Y. Xue, and Y. P. Tan. 2005. Shot boundary classification and refinement using inter-frame similarity patterns. Proc. of the IEEE International Conference on Information, 673–67, December, Las Vegas, USA.

Zedeh, L. A. 1989. Knowledge representation in fuzzy logic. *IEEE Transactions on Knowledge and Data Engineering* 1 (1):89–100. March. doi: 10.1109/69.43406.

Zhang, L., K.-T. Wu, and P. Li. 2018. An improved adaptive level set method for image segmenttation. *International Journal of Pattern Recognition and Artificial Intelligence* 32 (5):1–19. May. doi: 10.1142/S0218001418540137.

Zhi, M., and A. N. Cai. 2005. Shot change detection with adaptive thresholds. Proc. of IEEE International Workshop on VLSI Design and Video Technology, 144–49, May, Suzhou, China.

Zhu, L., J. Qu, M. A. Rahman, and W. Hong. 2010. An integrated method for video shot boundary detection. Proc. of the IEEE Southeast Conference, 151–54, March 18–21, Charlotte-Concord, North Carolina, USA.