Scene Change Detection using the Automated Threshold Estimation Algorithm

Kyong-Cheol Ko, Yang-Won Rhee
Dept. of Computer Information Science, Kunsan National University

Abstract

This paper presents a method for detecting scene changes in video sequences, in which the χ^2 -test is modified by imposing weights according to NTSC standard. To automatically determine threshold values for scene change detection, the proposed method utilizes the frame differences that are obtained by the weighted χ^2-test. In the first step, the mean and the standard deviation of the difference values are calculated, and then, we subtract the mean difference value from each difference value. In the next step, the same process is performed on the remained difference values, mean-subtracted frame differences, until the stopping criterion is satisfied. Finally, the threshold value for scene change detection is determined by the proposed automatic threshold estimation algorithm. The proposed method is tested on various video sources and, in the experimental results, it is shown that the proposed method is reliably estimates the thresholds and detects scene changes.

Keywords: scene change detection, weighted χ^2-test, mean, standard deviation, automated threshold estimation

1. Introduction

For the processing of a huge amount of video data, it is necessary to development of fast and efficient techniques in indexing, browsing and retrieval of videos [1] [2]. Video segmentation is the first step to establish video database systems and it is an essential basic work to segment video sequence into shots where each shot represents a sequence of frames having the same contents.

There are two kind of scene change, abrupt and gradual. The abrupt scene change is the change of scene with cut of camera and the gradual scene change is that of scene with camera action for fade-in, -out, and dissolves [3][4][5]. It is generally known that abrupt scene change detection is easier than gradual one. For detecting scene changes, thresholds have to be pre-assigned. This is the major problem of the scene change detection, and it is difficult to specify the correct threshold that determines the performance of scene change detection.

To segment a video sequence into scenes, a number of scene change detection algorithms have been reported in the literature [1, 3, 6-11]. In general, these algorithms can be categorized into four parts: Pixel-based algorithms [3], Histogram-based algorithms [6-7], Block-based algorithms [8-9], Clustering-based algorithms [10-11].

Pixel-based algorithms compare the pixels of two adjacent frames across the same location. Then, a scene change is declared if inter-frame difference (the sum of pixel by pixel differences across the same location) between consecutive frames exceeds the pre-assigned threshold.

In histogram-based algorithms, the difference of the intensity or color histogram is used as dissimilarity measure between two frames. Histogram-based methods provide a better tradeoff between accuracy and speed, and its performance is good for the case of abrupt scene changes such as cuts. The best performance is obtained by χ^2-test [7].

For block-based algorithms, local attributes are used to measure the difference of two frames so that the effect of noise or camera flash can be reduced. A scene change is detected if the number of blocks that the differences of corresponding blocks in two frames are greater than the pre-assigned threshold exceeds a given lower bound. The pre-assigned threshold is also the case with histogram-based algorithms and block-based algorithms.

Clustering techniques are used to categorize all the frames in a video sequence into several clusters (e.g. k-clusters in K-means clustering algorithm), in which a cluster represents a scene.

All scene change detection algorithms are based on pre-assigned threshold. If the threshold is too low, many key frames are extracted so that a video sequence is over-segmented. On the contrary, for a high threshold, many key frames may be missed, resulting in under-segmentation. Above all, a threshold that is appropriate for the others and it is not guaranteed that the threshold for one type of video data will not yield acceptable results for other types of inputs.

To solve the problem above mentioned, this paper proposes a method to determine thresholds that is adaptive for a variety of input video sequences. For robust scene change detection, we slightly modify the χ^2-test by imposing different weights on each channel of the RGB color space. We refer this χ^2-test as the weighted χ^2-test. To determine thresholds automatically according to an input video type, the proposed method utilizes the frame differences that are obtained by performing the weighted χ^2-test.

This research was supported by the Program for the Training of Graduate Students in Regional Innovation which was conducted by the Ministry of Commerce Industry and Energy of the Korean Government

[a]Department of Computer Science, Kunsan National University
roadkkc, ywrhee@kunsan.ac.kr

This paper proposes a scene-change detection method with three contributions: 1) extracting the difference values with weighted χ^2-test; 2) automatic threshold-estimation algorithm; and 3) higher detection rate (i.e., scene change should not be missed). The outline of this paper is as follows. In Section 2, we describe the weighted χ^2-test and the automatic threshold-estimation algorithm. A description of the proposed algorithm that was implemented is presented in this section. Experimental results or our evaluation are presented in section 3. Section 4 contains a summary and conclusions.

2. The proposed Scene Change Detection method

2.1 The weighted χ^2-test

In this paper, we calculate the frame differences from the weighted χ^2-test which combined the color histogram [1] [2] with χ^2-test [6]. The weighted χ^2-test can subdivide the difference values of individual color channels by calculating the color intensities according to NTSC standard. The weighted χ^2-test formula ($d_{w\chi^2}$) is defined as

$$d_{w\chi^2}(f_i, f_j) = \frac{1}{3N}\sum_{K=0}^{N-1}\Big(\frac{(H_i^r(k)-H_j^r(k))^2}{\max(H_i^r(k), H_j^r(k))}\times\alpha + \frac{(H_i^g(k)-H_j^g(k))^2}{\max(H_i^g(k), H_j^g(k))}\times\beta + \frac{(H_i^b(k)-H_j^b(k))^2}{\max(H_i^b(k), H_j^b(k))}\times\gamma\Big) \quad \text{(Eq. 1)}$$

, where N is the number of bins, and $H_i^r(k)$, $H_i^g(k)$, and $H_i^b(k)$ are the bin values of the histogram of i'th frame in red, green, and blue color channels, respectively. α, β, and γ are constants and, according to NTSC standard, we set these constants to 0.299, 0.587, and 0.114 , respectively.

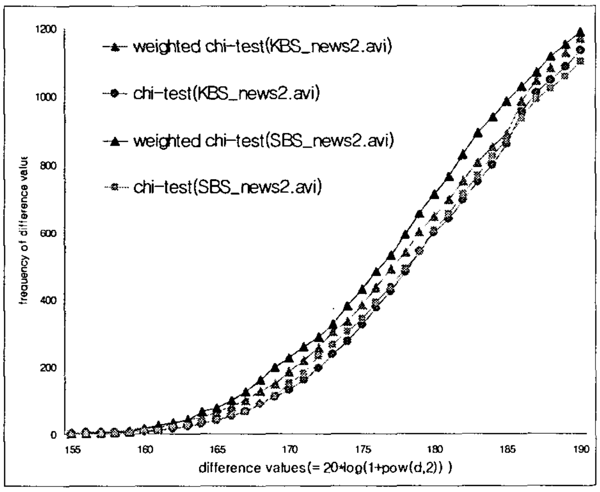


Figure 1. Comparison of the χ2-test and the weighted χ2-test on an input video sequence

Figure 1 shows the frame differences on a test video sequence, which are calculated by the χ^2-test and the weighted χ^2-test. Each difference values were scaled by the transform formula of image processing [12].

As shown in figure 1, the weighted χ^2-test has an advantage than the χ^2-test [7] in that the possibility of automatic threshold decision is high.

2.2 Automatic Threshold Estimation Algorithm using the means and the standard-deviation values

The difference values obtained from the weighted χ2-test is the basic data to extract the representative frames which was occurred during the scene change. Thus obtained data can be used for the feature extraction and it can be used for the detection of scene change. Pseudo code-1 shows the procedural calculation course of the mean and the standard-deviation from the extracted difference values to decide the most proper threshold from video sequence.

Pseudo code 1. Automatic threshold decision algorithm

Step 1: Calculate the total difference values and 1_th mean value from given video sequences

for(i=1; i<=n; i++){
// n = total number of frames

difference values $X = \{x_1, x_2, x_3, \ldots x_n\}$*;}*
/ X is calculated using the weighted* χ^2*-test from the consecutive frames* (f_i, f_j) **/*

1_th mean value $m = 1/n \times \sum_{i=1}^{n} x_i$ *;*
1_th standard deviation value $\sigma = \sqrt{1/n \times \sum_{i=1}^{n}(x_i - m)^2}$ *;*

Step 2: Calculate the t_th mean and standard deviation value

for(i=1; i<= n; i++) {
if($x_i > m^{t-1}$*)*
$X^t = \{x_i\}$*; //* $X^t = \{x_i, \ldots x_p\}$ and $(i \le p \le n)$
else x_i *is eliminated; }*

t_th mean value $m^t = 1/p \times \sum_p x_p$ *;*
t_th standard deviation value
$\sigma^t = \sqrt{1/k \times \sum_p (x_p - m^t)^2}$ *;}*
if($\sigma^t > \sigma^{t-1}$*) goto Step 2;*
else goto step 3;

Step 3: Decide the threshold value

*/*automatic threshold value is decided*/*
$th = m^t$ *; //threshold representative value*

$f_n = p$; //number of representative frames

/* confidence interval values are calculated to measure the sensitivity of estimated threshold */
/* α (%) confidence interval (α (95%) = 1.96) */

$val = \alpha \times \sigma^t / \sqrt{f_n}$;

/* calculation of max and min means based on confidence interval using the m^t */

// max_threshold confidence interval

$th_{MAX} = m^t + val$;

// min_threshold confidence interval

$th_{MIN} = m^t - val$;

// number of frames for the max and min mean values
for(i=1; i<=n; i++) {
  if($x_i > th_{MAX}$) $f_{\max}$++;
  if($x_i > th_{MIN}$) $f_{\min}$++; }

The mean and standard-deviation values extracted from the difference values are the important data to decide the automatic threshold. The mean is the essential data to decide the threshold from the detected difference values and the standard deviation is the reference data automatically decide the best threshold from the calculated means. The automatic decision of threshold value is based on the calculated means and standard deviation values over an entire difference values. The proposed automatic threshold-estimation algorithm can decide more adequate threshold according to the video sequence type.

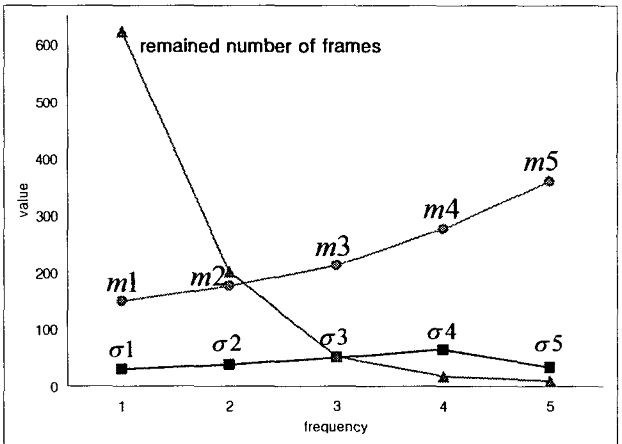


Figure 2. Distribution graph of mean, standard-deviation and remained number of frames

Figure 2 shows a distribution of means and standard-deviation values, and remained number of frames after the means exclusion according to the algorithm which is sequentially calculated by the proposed Pseudo code 1. All possible means and standard deviation vales are calculated and the remaining numbers of frames are checked.

Figure 3 shows a normal distribution graph using the calculated means and standard-deviation values which was calculated in Figure 2.

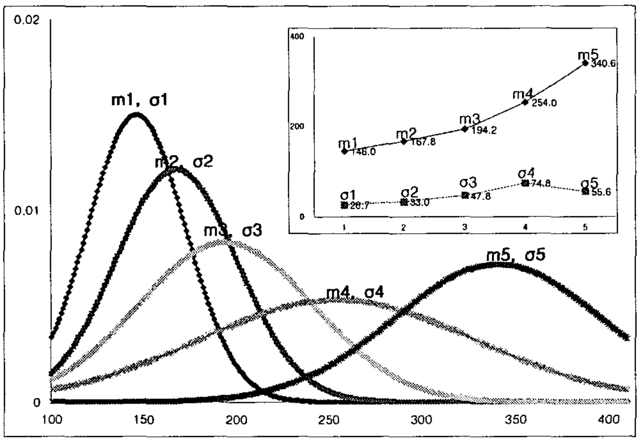


Figure 3. Normal distribution graph using the mean and standard deviation values

It shows a more widely normal distribution graph in 4-th means and standard-deviation values. This paper propose the automatic threshold decision algorithm and the threshold can be detected from the means and standard deviation values which is sequentially calculated by the proposed Pseudo code1. Thus the proper threshold could be selected from the mean value if the corresponding standard deviation has maximum value.

Figure 4 shows the distribution graph of remaining number of frames after means exclusion.

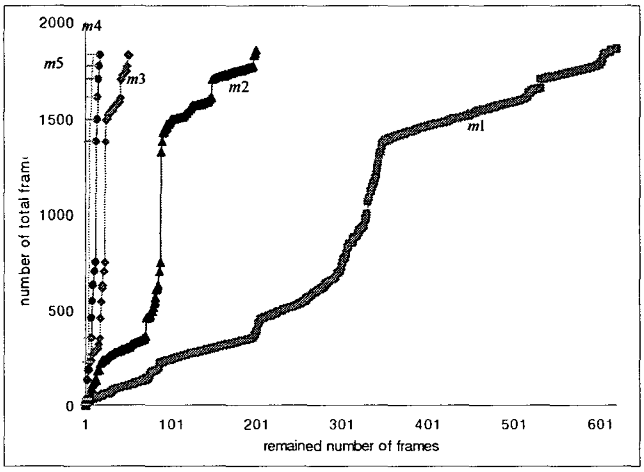


Figure 4. Distribution of remained number of frames after the means exclusion

As shown in Figure 4, we can estimate the mean values (m1~m4) as the threshold according to step 3 course of the proposed algorithm, and 4-th means value is the proper threshold because the standard deviation value is bigger than the previous values. So the estimated threshold value (m4) can detect 18 numbers of representative frames and it shows all possible abrupt scene change (12) are detected and the rest frames (6) is a gradual scene changes. If the mean is too low, many frames are extracted so that a representative frames is over segmented. On the contrary, for high means, many representative frames may be missed. So the threshold selection is a difficult problem.

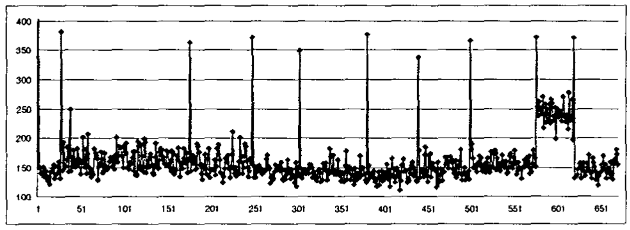
3. Experimental Results

We evaluate the performance of our proposed method with DirectX 8.1 SDK, MS-Visual C++ 6.0 on Windows XP. The proposed method has been tested on several video sequences such as news videos that a lot of scene changes occurs, as shown in table 1. Each video sequences has the various types digitized in 320x240 resolution at 20 frames/sec.

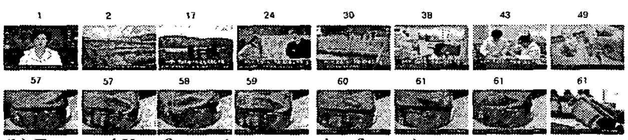
Table 1. Video sequences types used in the experiment

video	# of frames	# of predefined scene changes (A)		
		Abrupt	Gradual	Total
M_newsl	397	5	0	5
K_news2	1857	17	0	16
K2_news3	898	10	1	11
Y_news4	670	10	0	10
S_news5	1871	12	2	14
Adv_1	697	23	3	26
Adv_2	720	4	1	5
Adv_3	652	6	2	8
Adv_4	682	7	2	9
Adv_5	592	9	4	13

Figure 5, Figure 6 shows an example of scene change detection, in which frame differences are calculated by the weighted χ2-test and the extracted key frames are the representative frames of scene change in video sequence. Distribution of mean and standard deviation value is calculated with the proposed automatic threshold decision algorithm



(a) Frame differences by the weighted χ2-test

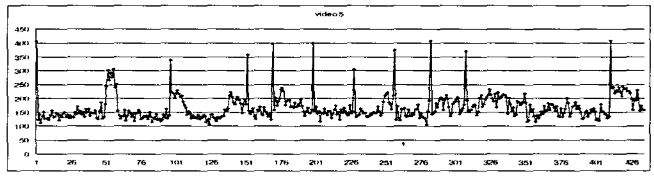


(b) Extracted Key frames(representative frames)

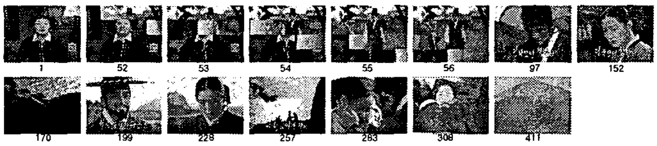
Adv_5	-	1			2			3			4		
		th_{-min}	th	th_{-max}	th_{-min}	th	th_{-max}	th_{-min}	th	th_{-max}	th_{-min}	th	th_{-max}
m-values	-	157.2	160	162.9	191.6	198.4	205.2	247.3	261.1	274.9	305.7	331.6	357.6
σ-values	-	37.98			49.49			53.50			52.91		
frames	670	241	203	184	64	58	54	27	16	11	10	10	8

(c) Distribution of mean and standard-deviation values from frame differences

Figure 5. Example of the scene change detection using the proposed algorithm in news video



(a) Frame differences by the weighted χ2-test



(b) Extracted Key frames(representative frames)

Adv_5	-	1			2			3			4		
		th_{-min}	th	th_{-max}	th_{-min}	th	th_{-max}	th_{-min}	th	th_{-max}	th_{-min}	th	th_{-max}
m-values	-	163.5	167.8	172.2	203.2	211	218.9	243.2	259.7	276.2	321.6	347.6	373.7
σ-values	-	46.3			50.9			61.8			51.5		
frames	435	171	160	142	66	54	40	18	15	14	9	8	6

(c) Distribution of mean and standard-deviation values from frame differences

Figure 6. Example of the scene change detection using the proposed algorithm in entertainment video

The performances of automatic scene change detection algorithms are usually expressed in terms of recall and precision. The recall parameter defined the percentage of true detection with respect to the overall events (scene changes) in the video sequences. Similarly, the precision is the percentage of correct detection with respect to the overall declared event. The recall and precision are defined as

$$\text{Recall} = \frac{N_c}{N_c + N_m} \times 100\% \text{ , and}$$

$$\text{Precision} = \frac{N_c}{N_c + N_f} \times 100\% \qquad \text{Eq. 2)}$$

, where N_c: number of correct detection; N_m: number of miss; N_f: number of false detection; $N_c + N_m$: number of the existing events; $N_c + N_f$: number of overall declaration.

Table 2 show the automatic decided thresholds and detected number of frames according to the automatic threshold decision algorithm. The means of confidence interval (95%) has the maximum value th_{max} (= 95%(+)) and minimum value th_{min} (=95%(-)).

Table 2. Automatic decided threshold and detected number of frames

video	Automatic decided threshold			Detected number of frames(B)		
	th_{max}	th	th_{min}	th_{max}	th	th_{min}
M_news1	203.73	228.19	252.64	14	12	7
K_news2	259.88	277.41	294.94	20	18	18
K2_news3	271.54	302.04	332.54	11	10	10
Y_news4	247.31	261.07	274.84	27	16	11
S_news5	233.07	254.03	274.99	19	18	14
Adv_1	281.35	290.82	300.29	45	38	34
Adv_2	180.65	196.19	211.73	19	10	10
Adv_3	216.16	232.14	248.12	27	16	11
Adv_4	245.34	260.33	275.33	15	13	10
Adv_5	250.51	259.42	268.34	38	33	29

Means of confidence interval are calculated to measure the sensitivity of decided thresholds. The difference of detected number of frames between mean of decided threshold and mean of confidence interval value (th_{max}, th_{min}) will be adequate when it has small value.

Table 3 shows the correctness and detection rate of detected frames with the scope of threshold. We show the statistics of our experimental results in Table 3.

Table3. Distribution of the detected rate, recall and precision

video	Detected rate ((B/A) * 100)			Recall(%)			Precision(%)		
	th_{max}	th	th_{min}	th_{max}	th	th_{min}	th_{max}	th	th_{min}
M_news1	280	240	140	100	100	100	61	63	78
K_news2	118	106	106	100	100	100	87	95	95
K2_news3	100	90	90	92	91	91	92	100	100
Y_news4	270	160	110	100	100	100	61	73	92
S_news5	136	129	100	100	100	93	79	86	100
Adv_1	173	146	131	100	100	100	70	76	81
Adv_2	380	200	200	100	100	100	58	67	67
Adv_3	338	200	138	100	100	100	59	67	79
Adv_4	167	144	111	100	100	91	71	76	91
Adv_5	292	254	223	100	100	100	66	62	64

For the proposed scene-change detection algorithm, the reported recall is about 99.1% and the detected rate is about 166.9%. Most of the scene changes are detected. The missed scene changes are very ambiguous due to the slow camera movement.

The purpose of proposed algorithm is not to detect the correct scene change frames but is to detect the all possible scene change frames from video sequence. Experimental result shows the proposed algorithm is effective and can be adapted to video indexing system.

4. Conclusion

This paper has been proposed a method for scene change detection using the weighted χ^2-test and the automatic threshold decision algorithm. In experimental results, we showed that the proposed algorithm effectively calculated the frame differences and extracted scene changes with adaptive thresholds to each input video. The proposed χ^2- test showed the better performance than the χ^2-test. In addition, we showed that the frame differences by the proposed method are applicable for determining a threshold value that is appropriate to a given input video.

This paper focused on abrupt scene changes, so, if video sequences have gradual changes, flashlights, or lightening effects, the automatically decided thresholds will not be appropriate to detect scene changes correctly. Thus, we are considering the condensation of the frame differences for the further research.

References

[1] Huang C.L. and Liao B. Y., "A Robust Scene Change Detection Method for Video Segmentation", IEEE Trans on CSVT, Vol. 11, No. 12, December(2001) 1281-1288.

[2] Gargi U., Kasturi R., and Strayer S.H.,"Performance Characterization of Video Shot Change Detection Methods, IEEE Trans on CSVT, Vol. 10, No. 1, February(2000)0001-0013.

[3] Zhang H., Kankamhalli A. and Smoliar S., "Automatic partitioning of full-motion video, ACM Multimedia Systems, New York: ACM Press, Vol. 1, (1993) 10-28.

[4] Dailianas A., Allen R. B., England p., "Comparison of Automatic Video Segmentation Algorithms, Large Commercial Media Delivery Systems, Proc. SPIE 2615, Oct. (1995) 2-16.

[5] Rainer Lienhart, "Comparison of Automatic Shot Boundary Detection Algorithms", Storage and Retrieval for Still Image and Video Databases VII, Proc. SPIE 3656~29, (1999).

[6] Nagasaka A. and Tanaka Y., "Automatic video indexing and full-video search for object appearances", Visual Database System II, (1992) 113-127.

[7] Sethi I. K. And Patel N., "A statistical approach to scene change detection, SPIE, vol. 2420, (1995) 329-338.

[8] Ekin A., Tekalp A.M. and Mehrotra R., "Automatic soccer video analysis and summarization", IEEE Transaction on Image Processing, vol. 12, no. 7, July (2003) 696-807.

[9] Pengwei Han and Ying Chen, "Co-Histogram and Its Application in Video Analysis", ICME., Vol. 3, June (2004) 1543-1546.

[10] Joshi A., Auephanwiriyakul S. And Krishnapuram R., "On Fuzzy clustering and

Content Based Access to Networked Video Database", IEEE conference, English International workshop on Continuous-Media Databases and Applications, (1998) 42-49.

[11] Hanjalic A., Zhang J., "An Integrated Scheme for Automated Video Abstraction Based on Unsupervised Clusterdity Analysis", IEEE Transactions on Circuits and Systems for Video Technology 9, (1999) 1280-1289.

[12] P. Browne, A. F. Smeaton, N. Murphy, N. O'Conner, "Evaluation and Combining digital video shot boundary detection algorithms", in Proc. of the 4th Irish Machine Vision and Information Processing Conference, Queens University Belfast, 2000.

[13] Z, Cerbejivam C. Nikou, I. Pitas, "Shot Detection in Video Sequences using Entrophy-Based Metrics", ICIP 2002, Vol. 3, pp. 421-424, 2002.

[14] Alan Hanjalic, "Shot-boundary detection: unraveled and resolved", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 12, No. 2, pp. 90-105, February. 2002.

[15] Dong Zhang, Wei Qi, Hong Jiang, "A New Shot Boundary Detection Algorithm", IEEE Pacific Rim Conference on Multimedia, pp. 63-70, 2001.

Kyong-Cheol Ko is an post-graduate student in the department of Computer Information Science at Kunsan National University. Ko's research focus is Intelligent agent and computer vision, video processing technology. He is a member of the Association for KIPS and the KISS.

Yang-Won Rhee is a professor in the department of Computer Information Science at Kunsan National University. Dr. Rhee's research focus is artificial intelligence and computer vision, virtual reality, video processing technology. He is an active member of the Association for KIPS and the KISS