International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 9, No.2 (Mar-2020)
http://dx.doi.org/10.12785/ijcds/090213

Automatic Effect Generation Method for 4D Films

Eunsu Goh[1], Daeyeol Kim[1], Suyeong Oh[1] and Chae-Bong Sohn[1]

[1] Department of Electronics and Communications Engineering, Kwangwoon University, 01897 Seoul, Korea

Received 27 Sep. 2019, Revised 24 Feb. 2020, Accepted 26 Feb. 2020, Published 01 Mar. 2020

Abstract: The 4D film is a technology that stimulates the viewer's senses by using motion chairs and special equipment to increase immersion. 4D movies have recently gained enormous popularity by satisfying the five senses of users by using water spray and wind scent of motion chairs. Recently, efforts have been made to apply 4D systems to personal equipment such as mobile devices. However, to create 4D content that can be used on 4D devices, a large number of skilled workers have to make manual effects for several decades. In this paper, we propose a method of generating 4d effects by classifying audio signals and motion of important objects in video using 4D movie's program stream.

Keywords: 4D Films, Effect Generation, Salient Object Detection, Optical Flow, Audio Classification

1. INTRODUCTION

Recently, as user-friendly content has become popular, interest in the 4D industry has been increasing. While 2D content is a traditional drama and movie 3D content is content that enjoys wearing stereoscopic glasses, 4D content adds a sense of immersion to the user by providing the five senses affect using motion chairs and special equipment.

As it is an industry of interest, research is being carried out to enjoy 4D contents at home, away from enjoying in large places such as movie theaters and science museums. Some companies make and sell home 4D devices. As 4D devices evolve, 4D content must be created as well, but it takes much time and labor to produce effects for one 4D movie as three professionals have to spend 16 days [1].

As many resources are needed to produce 4D effects, optical flow and RANSAC method [1], VGG19, YOLO, SoundNet method to produce the 4D effect [2], 4D using visual information Attempts to have been made to produce movie motion effects. Unlike the previous methods, we consider that the stored movie is a stream consisting of video and audio, and analyze the information of each stream, and use it to automatically produce 4d effects.

Through the analysis of object movement based on Salient Object and audio sound. Unlike traditional methods, it analyzes and uses video and audio streaming information. Divided video into frames and extract Salient Object and flow maps. Process new images using Salient objects and flow maps. We then propose a way to create 4D effects using new images and features created through CNN.

2. RELATED WORKS

A. MPEG Program Stream (PS)

According to [3], Video extensions like .mp4 consist of PS. PS is a packet made to store a movie-like program. It has both video and audio streams and their compression content. We obtain video and audio information from this stream information and use it in the proposed method. Fig. 1 presents program stream architecture.

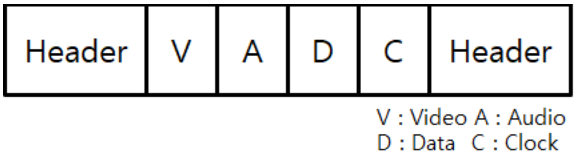

Figure 1. Program Stream architecture

MPEG-V part 3 defines a standard for transmitting 4D effects and defines a method of creating 4D effect information (SEM). The SEM includes the kind intensity of the effect and the presentation time stamp (pts). When the SEM is handed over to the media processing engine, it is delivered to the user through environmental devices such as monitors and motion chairs [4].

E-mail: dmstn96@kw.ac.kr, wagon0004@kw.ac.kr, pentalvega@kw.ac.kr, cbsohn@kw.ac.kr

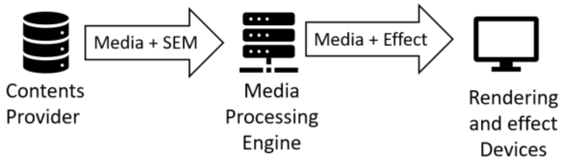

Figure 2. MPEG-V part3 system configuration

B. Salient Object Detection

To understand the visible image, the human first detects the main object and then filters the desired information in the surrounding area of the object. The Visual Saliency field, which studies how humans understand images, is largely divided into eye detection and salient object detection. This study focuses on the Salient Object Detection method because it proceeds to people who do not have many unspecified eye-tracking devices.

The Salient object detection model, unlike the Object segmentation task, is only used to extract Salient objects. Initially, Salient objects were extracted based on the area you specified. Since then, research has been carried out using methods such as Gaussian Mixture Model (GMM) [5], Principal Component Analysis (PCA) [6], Support Vector Machine (SVM) [7], and computational methods using pixel information such as SLIC [8]. However, these methods have considerable associations, which makes it difficult to apply them in real-time. Recently, the study of the Salient Object Detection Model, which has real-time properties due to the development of deep learning, is in progress.

We used PoolNet for salient object detection [9]. PoolNet's network structure is a U-Shape structure modified from the pyramid-like structure of the Convolutional Neural Network (CNN) [10].

Pyramid-like structure extracts feature extracted in a top-down manner and the low- and high-resolutions of the extracted results are grouped. Objects are detected by extracting features independently at each level, and multi-scale features can be used efficiently by reusing previously calculated features at higher levels. As CNN itself goes through the layers, it creates a pyramid structure, and as it goes forward, it becomes more semantic. Predictive processes are added to each layer to make the model more resistant to scale changes. This is a combination of pyramid structures generated from skip connection, top-down, and CNN forward. The semantic information extracted from the forward is upsampled in the top-down process to increase the resolution, and the local information lost from the forward is supplemented by the skip connection to be robust to scale change [11].

The pyramid-like structure of CNN has a limit that the acceptable field size is not proportional to the depth of the layer, and that shallow stages occupy more space. To overcome this, various classification networks were created, and the U-shape structure is one of them. PoolNet consists of two main modules, the global guidance module (GGM) and the feature aggregation module (FAM). Each layer of the network has a different feature.

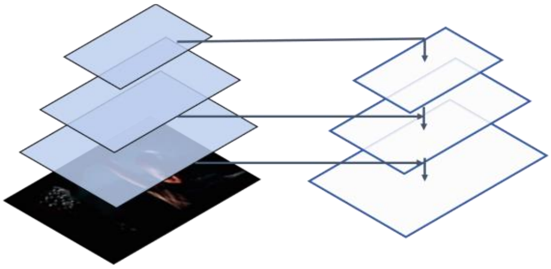

Figure 3. Feature Pyramid Network Structure

C. Motion Estimation

We use optical flow to find the direction pattern of the object in the image [12]. Optical flow refers to the pattern of motion of an object between successive video frames. We used optical flow to perform the motion estimation of the image object. FlowNet was created because it was not clear whether a standard CNN architecture [13] could perform optical flow estimation. FlowNet was built up to version 2.0 [14] but required too many parameters to perform optical flow, resulting in LiteFlowNet, a faster and lightweight version of FlowNet2.0. According to [15], LiteFlowNet uses a special architecture for data normalization and fidelity. It warps features per the pyramid level and creates a cascading flow. And it does feature-driven local convolution. Optical flow estimation visualizes the motion vector as an arrow-shaped line [16]. LiteFlowNet, on the other hand, creates a flow map that visualizes the motion vector using color. We can see in which direction the object moves by the color of this flow map.

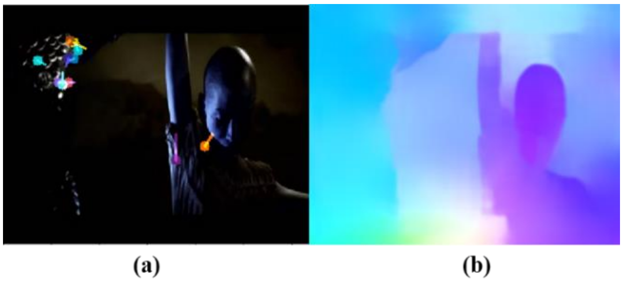

Figure 4. Optical flow representation

(a) original optical flow (b) flow map

D. Audio Classification

Audio classification technology using deep learning can be divided into two methods: using raw data directly and visualizing depending on how the data is generated.

Directly using raw data forms raw data into 2-D wave images. It then uses windowing techniques to generate data at regular intervals. Then we enter the VGG base's network input and train the filter to classify the waveform. This achieved about 72.9% accuracy for ESC-50 data consisting of 50 different audio types [17].

There are two ways to visualize audio: using short-time Fourier-transform (STFT) and using log amplitude Mel-spectrogram with nonlinearity similar to the shape of receiving ear audio signal. Traditionally, this result is used to determine what sounds you hear through a set of rules. But deep learning classifies audio using the Convolutional Neural Network, which can extract image features [18-20].

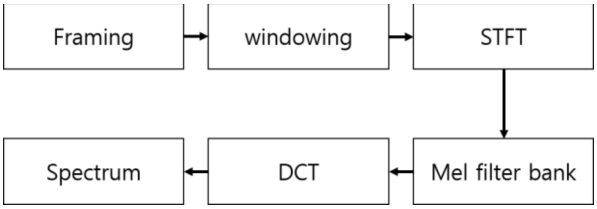

Figure 5. The audio feature extracting using MFCC (Mel Frequency Cepstral Coefficient)

MFCC (Mel Frequency Cepstral Coefficient was used for feature extraction of negative signals. To illustrate the overall process depicted in Fig. 5, the input audio is framing at regular intervals. After that, determine the window size for the corresponding section and proceed with signal processing. For the audio signal divided into short sections, STFT (Short-Time Fourier Transform) is applied to each section. Apply the Mell filter bank to the resulting spectrum and add up the energy of each filter. The output spectrum is obtained by applying DCT (Discrete Cosine Transform) to the logarithm of all filter bank energies.

3. PROPOSED METHOD

Our proposed method for 4d films effect using the deep learning method. In general, the effects of 4d film are produced by experts by analyzing the contents of the video frames and a significant change of audios. We will propose an automatic effect generator method by paying attention to the change of the movement of the Salient object. We divided the motion estimation module to produce motion effects and the audio classification module to produce the extra effect. Fig. 6 presents the overall architecture of the automatic effect generation method.

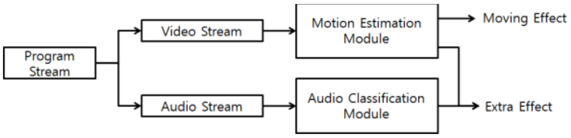

Figure 6. The Proposed method for Automatic effect generation method for 4D films

A. Motion Estimation Module

This module is a module for generating motion effects using a video stream. Fig. 7 presents Motion Estimation Module architecture.

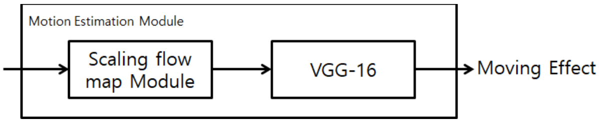

Figure 7. Motion Estimation Module

The Scaling flow map module uses the previous and current frames of the video to create a scaling image to create a Salient Object-based motion effect. Extract Salient Object using the current frame as an input of PoolNet. Then, using the previous frame and the current frame, a flow map for estimating the motion of the object is generated. The scaling factor A is then used to create a scaling image that represents the Salient Object's motion well. Fig.8 is a block diagram of the scaling flow map module.

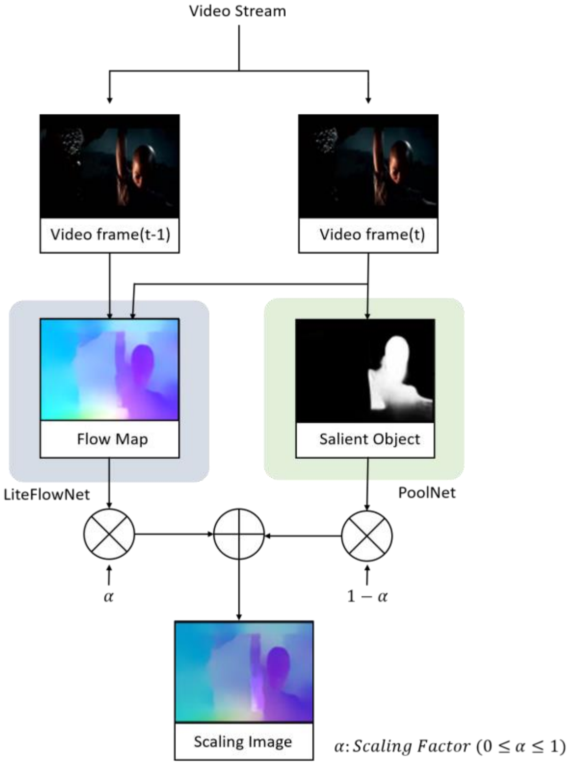

Figure 8. Scaling flow map Module

B. Audio Classification Module

This module is a module for generating Extra effects using an Audio stream. Fig. 9 presents the Audio Classification Module architecture.

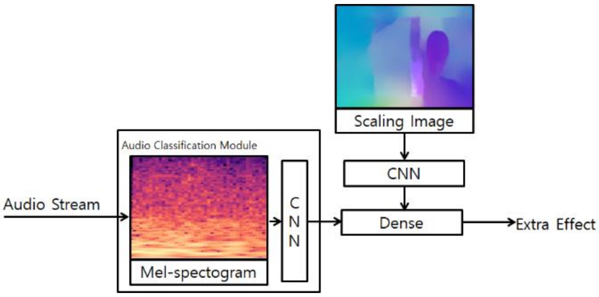

Figure 9. Audio Classification Module

The most representative music information retrieval (MIR) of audio classification is based on imaging the audio signal [21]. Among them, Mel-spectrogram is the most useful, and we want to use it to classify audio signals.

4. EXPERIMENT

A. *Machine Environment*

We trained the proposed model in Table Ⅰ environment and conducted a simulation viewing and testing in the Unity cinema environment. Fig. 10 shows our Unity cinema environment.

TABLE I. MACHINE ENVIRONMENT

CPU	AMD Ryzen 7 2700 octa-core Processor 3.40GHz
RAM	16GB
VGA	NVIDIA GeForce GTX 1080 Ti

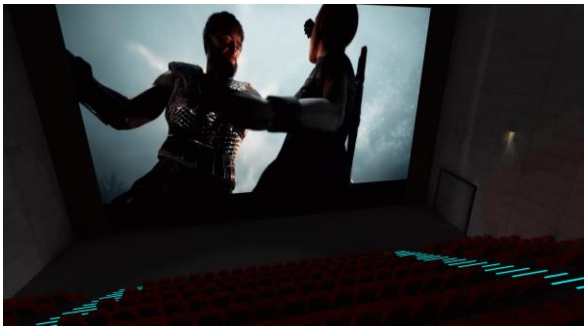

Figure 10. Unity cinema environment

B. *Custom Dataset*

We chose two cinematic videos of Assassin's Creed [22] for the experiment. The events corresponding to each front of the video are divided into Moving Effect and Extra Effect and input into one-hot encoding. The Moving Effect consists of five parts: moving back(mb) and forth(f), left(l), right(r), up and down(ud), and no effect(n). Extra Effect consists of four parts: vibration(v), water jet(wa), wind jet(wi), and no effect(n).

Figure 11. Assassin's Creed cinematic video
(a) 2018 E3 World Premier (b) Origin Cinematic

We created separate files for each of the motion and other effects. Each file also contains frame number one-hot encoding information like Table Ⅱ

TABLE II. EXAMPLE OF LABEL FILE

Motion Effect Label	Extra Effect Label
frame, mb, f, l, r, n	frame, v, wa, wi, n
1, 0, 0, 0, 0, 1	1, 0, 0, 0, 1
2, 0, 0, 0, 0, 1	2, 0, 0, 0, 1
3, 0, 0, 0, 1, 0	3, 0, 0, 0, 1
4, 0, 0, 0, 1, 0	4, 0, 0, 0, 1
5, 0, 0, 0, 0, 1	5, 0, 0, 0, 1
..	..

Creating a data set requires a lot of labor and time. For this reason, the following data augmentation techniques are used.

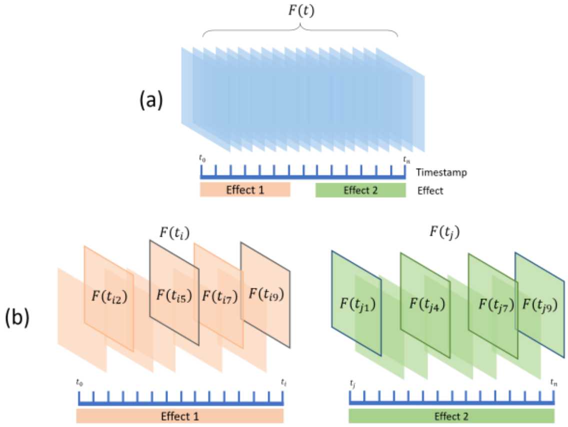

Figure 12. Data Augmentation Method
(a) Video with effect timestamps (b) Video sequence about the effect

Fig. 12 shows how the data augmentation method. The effect has more than one frame. One effect also has a constant pattern of optical patterns. Therefore, we propose a method of learning by making ordered pairs as shown in Table Ⅲ.

TABLE III. PSEUDOCODE FOR DATA AUGMENTATION METHOD

```
DATA Augmentation
for (i = 1 ; i <= number of effects ; i++)
{
  for (j  = start of effect(i) ; j < end of effect(i); j++)
   {
      for (k = j+1;  k <= end of effect(i) ; k++)
          add data pair(frame(j,k))
    }
 }
```

The proposed data augmentation method helps to generate motion and additional effects in high motion movies, even when learning low motion movies.

C. Training Deep Learning Models

We use PoolNet for salient object detection. We used a dataset called DUTS on it [23], which is the large benchmark for salient object detection. DUTS contains 10,553 training images and 5,019 test images.
LiteFlowNet was used for motion estimation, and three optical flow datasets were used for LiteFlowNet. First, the “flying chair” dataset consists of 22,872 images [24]. Second, Middlebury (Middlebury Stereo Dataset) has 71 images [25], and finally, MPI sintel Dataset contains 5.3GB of images [26].

We use a pre-trained PoolNet to get a Salient Object. The flow map was obtained using the pre-trained Lite Flow Net. create a new Scaling Image by modifying the Scaling Factor for each obtained image. This scaling image is used for both motion effect generation and other effect generation, and the scaling factor greatly affects the accuracy of the effect generation. We expressed the salient object in white and the other background in black. Therefore, when generating the scaling image using the flow map and the salient object using the scaling factor, the motion information in the salient object can be more emphasized.

To create a motion effect, we trained the scaling image and the motion effect label using the VGG model [27]. VGG is a network created with the idea of extracting the same feature: two convolutions with a 3x3 filter and one convolution with a 5x5 filter. VGG is used a lot because it is easy to apply, and extracts feature well. However, due to many parameters, proper customization is required. All CNN structures of this paper are designed based on VGG, and the appropriate parameter size was determined by changing the size of the convolution filter. We performed the experiment using only the scaling image without the shape of the eye, nose, mouth, etc., so the number of filter filters of the VGG was reduced to 1/4.

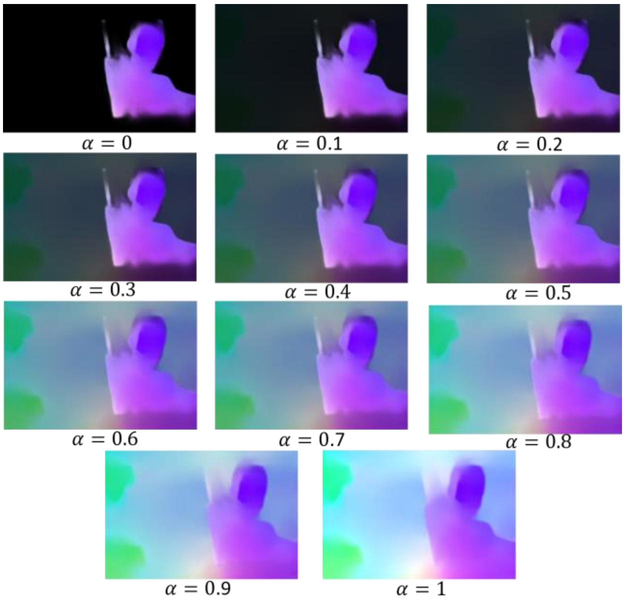

Figure 13. Image Change by Scaling Factor (α)

TABLE IV. MOTION EFFECT GENERATING RESULT

Scaling Factor	Train acc (%)	Test acc (%)
0.1	76.2	76.0
0.2	76.4	75.5
0.3	77.1	71.6
0.4	79.9	77.2
0.5	84.2	81.1
0.6	**95.7**	**83.9**
0.7	89.9	81.4
0.8	87.4	83.9
0.9	84.6	80.7

Since the Motion Effect is affected by the direction and strength of the Salient Object, we experimented with adjusting the Scaling Factor. The experimental results showed the highest accuracy at 0.6 Scaling Factor. In addition, the accuracy is not very different when compared with the conventional network, which is reduced to 1/4 the number of filters in VGG.

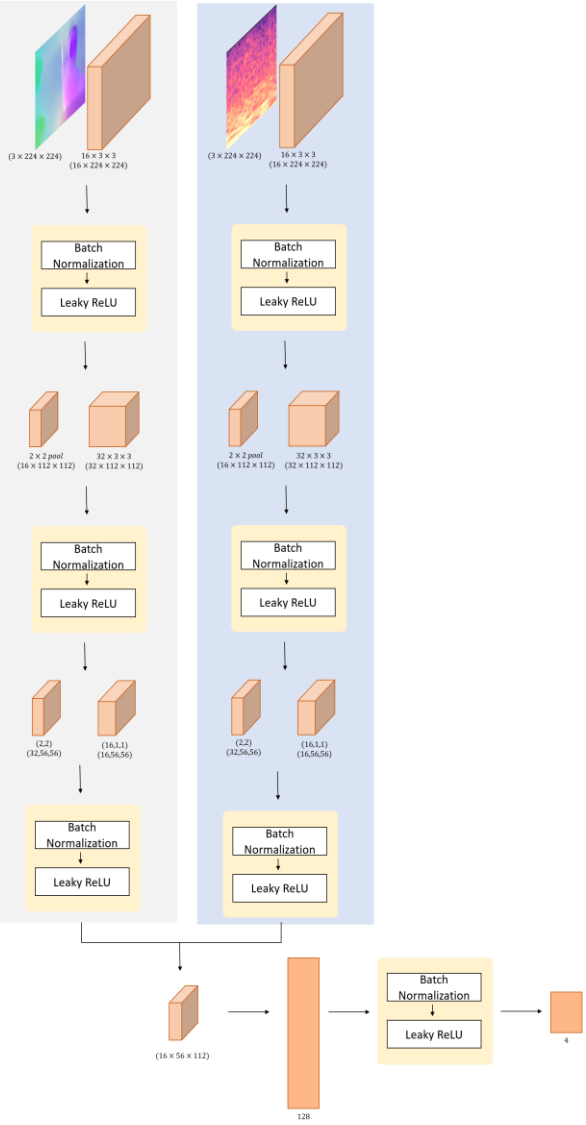

Figure 14. Extra Effect Classification Network Architecture

In the dual-input CNN module for creating an extra effect, the extra effect information was used as a label. The architecture of our dual-input CNN is shown in Fig. 14. We defined a convolution block using 32 filters, 3x3 kernel CNN, 2x2 2D Max pooling, and batch normalization.

The dual-input CNN share their form with each other, according to Table V. We experimented with changing the architecture of the Fully Connect Layer. We experimented with varying the size of the Convolution Layer (C) filter, the use of Dropout (Drop), the use of BatchNorm (BN), and the number of neurons in the Fully Connect Layer (Dense)[28-29].

TABLE V. CONVNET + FC CONFIGURATION

ConvNet + FC Configuration					
A	B	C	D	**E**	F
Input Scaling Image (224 x224 RGB)					
Conv3-16 Drop-0.25	Conv3-16 BN	Conv3-32 BN	Conv3-16 Drop-0.25	**Conv3-16 BN**	Conv3-32 BN

LeakyReLu(0.1)
Maxpool(2x2)					

Conv3-32 Drop-0.25	Conv3-32 BN	Conv3-64 BN	Conv3-32 Drop-0.25	**Conv3-32 BN**	Conv3-64 BN

LeakyReLu(0.1)
Maxpool(2x2)					

Conv3-64 Drop-0.4	Conv1-16 BN	Conv1-32 BN	Conv3-64 Drop -0.4	**Conv1-16 BN**	Conv1-32 BN

LeakyReLu(0.1)
Maxpool(2x2)					

Dense-256	Dense-256	Dense-256	Dense-128	**Dense-128**	Dense-128

TABLE VI. ACCURACY OF TABLE V

A	B	C	D	**E**	F
73%	79%	80%	75%	**83%**	80%

The E model is the most accurate. The 1x1 filter increases the nonlinearity without affecting the receptive field, thus reducing the parameters and speeding up the learning accuracy [30]. Also, because the batch normalization affects the regularization of the data set, the accuracy of learning can be improved.

TABLE VII. EXTRA EFFECT GENERATING THE RESULT

Scaling Factor	Train acc (%)	Test acc (%)
0.1	55.7	55.7
0.2	61.2	60.9
0.3	70.0	70.2
0.4	71.5	70.7
0.5	77.4	73.6
0.6	**83.0**	**77.9**
0.7	81.4	78.2
0.8	80.1	79.3
0.9	78.9	76.8

An experiment was conducted to determine the effect of the scaling factors on the production of the extra effect. As a result, the extra effect is more sensitive to motion information.

5. RESULTS AND DISCUSSION

We used two videos in the experiment and divided one video into two sections and used them for training and testing. 80% of one video was used for training and the remaining 20% was tested. Then, the similarity was measured after the experiment with the remaining video. The similarity measurement formula for this is the F1 score.

$$\mathrm{F1-score} = 2 * \frac{precision * recall}{precision + recall} \tag{1}$$

The similarity was measured using the highest test accuracy of 0.6 scaling factor. The sum of the Precision and Recall for each of the video audio was measured. As a result, an F1-score of 0.736 was obtained.

6. CONCLUSION

The effects of 4D content are divided into motion effects and other environmental effects. In this paper, the data set is generated by using a one-hot encoding method for each situation. We learned using the data set created by ourselves. As a result, it was possible to create an effect that was appropriate for each situation. However, because of the one-hot encoding method, the strength information of each effect could not be deduced.

Also, the proposed method requires about 10fps of slow speed and about 20GB of VGA memory when used in real-time.

This study showed that 4D contents can be produced using deep learning models. In the future, we will study lightweight models with real-time capabilities so that 4D content can be produced and viewed directly at home or in the cinema. Also, we want to output multiple effects in a scene and study them to infer the intensity of each effect.

ACKNOWLEDGMENT

This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2019-2016-0-00288) supervised by the IITP (Institute for Information & communications Technology Planning & Evaluation).

REFERENCES

[1] Lee, J., Han, B., & Choi, S. (2015). Motion effects synthesis for 4D films. IEEE transactions on visualization and computer graphics, 22(10), 2300-2314.

[2] Zhou, Y., Tapaswi, M., & Fidler, S. (2018). Now You Shake Me: Towards Automatic 4D Cinema. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 7425-7434).

[3] Li, W. (2001). Overview of fine granularity scalability in MPEG-4 video standard. IEEE Transactions on circuits and systems for video technology, 11(3), 301-317.

[4] Sensory information ISO/IEC CD 23005-3 3rd Edition Sensory Information

[5] Zivkovic, Z. (2004, August). Improved adaptive Gaussian mixture model for background subtraction. In Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004. (Vol. 2, pp. 28-31). IEEE.

[6] Jolliffe, I. (2011). Principal component analysis (pp. 1094-1096). Springer Berlin Heidelberg.

[7] Chang, C. C., & Lin, C. J. (2011). LIBSVM: A library for support vector machines. ACM transactions on intelligent systems and technology (TIST), 2(3), 27.

[8] Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., & Süsstrunk, S. (2012). SLIC superpixels compared to state-of-the-art superpixel methods. IEEE transactions on pattern analysis and machine intelligence, 34(11), 2274-2282.

[9] Liu, J. J., Hou, Q., Cheng, M. M., Feng, J., & Jiang, J. (2019). A Simple Pooling-Based Design for Real-Time Salient Object Detection. arXiv preprint arXiv:1904.09569.

[10] Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, Serge Belongie. Feature Pyramid Networks for Object Detection. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2117-2125

[11] Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2117-2125).

[12] Kajo, I., Malik, A. S., & Kamel, N. (2015, December). Motion estimation of crowd flow using optical flow techniques: A review. In 2015 9th International Conference on Signal Processing and Communication Systems (ICSPCS) (pp. 1-9). IEEE.

[13] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).

[14] Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., & Brox, T. (2017). Flownet 2.0: Evolution of optical flow estimation with deep networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2462-2470).

[15] Hui, T. W., Tang, X., & Change Loy, C. (2018). Liteflownet: A lightweight convolutional neural network for optical flow estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 8981-8989).

[16] Horn, B. K., & Schunck, B. G. (1981). Determining optical flow. Artificial intelligence, 17(1-3), 185-203.

[17] Piczak, K. J. (2015, October). ESC: Dataset for environmental sound classification. In Proceedings of the 23rd ACM international conference on Multimedia (pp. 1015-1018). ACM.

[18] Allen, J. (1977). Short term spectral analysis, synthesis, and modification by discrete Fourier transform. IEEE Transactions on Acoustics, Speech, and Signal Processing, 25(3), 235-238.

[19] Sahidullah, M., & Saha, G. (2012). Design, analysis and experimental evaluation of block based transformation in MFCC computation for speaker recognition. Speech Communication, 54(4), 543-565.

[20] Salamon, J., & Bello, J. P. (2017). Deep convolutional neural networks and data augmentation for environmental sound classification. IEEE Signal Processing Letters, 24(3), 279-283.

[21] Downie, J. S. (2003). Music information retrieval. Annual review of information science and technology, 37(1), 295-340.

[22] Quebec. U. (2018). Assassin's Creed Odyssey. Quebec: Ubisoft Quebec.

[23] DUTS Image Dataset http://saliencydetection.net/duts/#org0602ffb

[24] "Flying Chairs" Dataset https://lmb.informatik.uni-freiburg.de/resources/datasets/FlyingChairs.en.html

[25] Middlebury Stereo Dataset http://vision.middlebury.edu/stereo/data/

[26] MPL Sintel Dataset http://sintel.is.tue.mpg.de/

[27] Department of Engineering Science, University of Oxford. Visual Geometry Group

[28] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, 15(1), 1929-1958.

[29] Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167.

[30] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1-9).

Eunsu Goh received the B.S degree in Electronic Communication and Engineering from Kwangwoon University, Seoul, Korea in 2019. She is currently Master Student in department of Electronics and Communications Engineering, Kwangwoon University, Seoul, Korea. Her research interests include image and video processing, Deep Learning, and A.I. systems

Daeyeol Kim received the B.S degree in Electronic Communication and Engineering from Kwangwoon University, Seoul, Korea in 2016. He is currently Ph.D candidate in department of Electronics and Communications Engineering, Kwangwoon University, Seoul, Korea. His research interests include image and video processing, MIR systems, and A.I. systems

Suyeong Oh undergraduate in Electronic Communication and Engineering from Kwangwoon University, Seoul, Korea. His research interests include NLP system, A.I. system especially Deep Neural Networks.

Chae-Bong Sohn received the B.S., M.S., and Ph.D. degree in Electronic Engineering from Kwangwoon University, Seoul, Korea in 1993, 1995, and 2006, respectively. He is currently an associate professor in department of Electronics and Communications Engineering, Kwangwoon University, Seoul, Korea. His research interests include image and video processing, NLP systems, and A.I. systems