ExtVision: Augmentation of Visual Experiences with Generation of Context Images for Peripheral Vision Using Deep Neural Network

Naoki Kimura
University of Tokyo
Tokyo, Japan
kimura-naoki@g.ecc.u-tokyo.ac.jp

Jun Rekimoto
University of Tokyo / Sony CSL
Tokyo, Japan
rekimoto@acm.org

Figure 1. ExtVision augments visual experiences. These images are examples of our application. The context-images are generated and projected onto the peripheral area around the TV by our system.

ABSTRACT

We propose a system, called ExtVision, to augment visual experiences by generating and projecting context-images onto the periphery of the television or computer screen. A peripheral projection of the context-image is one of the most effective techniques to enhance visual experiences. However, the projection is not commonly used at present, because of the difficulty in preparing the context-image. In this paper, we propose a deep neural network-based method to generate context-images for peripheral projection. A user study was performed to investigate the manner in which the proposed system augments traditional visual experiences. In addition, we present applications and future prospects of the developed system.

Author Keywords

Spatially augmented reality; immersion; augmented video; large field of view; human vision; video extrapolation

ACM Classification Keywords

H.5.1. Multimedia Information Systems: Artificial, augmented, and virtual realities

CHI 2018, April 21–26, 2018, Montreal, QC, Canada

ACM ISBN 978-1-4503-5620-6/18/04…$15.00
https://doi.org/10.1145/3173574.3174001

INTRODUCTION

Presentation of images to a viewer's peripheral vision makes the experience of watching a video more immersive and exciting. In this paper, we describe how images are displayed as context-images in peripheral areas to enhance visual experiences. Illumiroom [10] projects context-images or visual effects around the TV screen and enhances the traditional gaming experience. ScreenX [20] and BarcoEscape [21] provide sub-screens to display video on both sides of the main screen, providing a more immersive movie experience.

However, peripheral presentation systems are not very popular at present, because it is difficult to prepare the context- images. Researchers have attempted the boundary value shooting method and computer vision processing to address this, but a self-sufficient method still remains to be developed.

In recent years, deep neural network (DNN)-based methods have shown considerable success in image completion. The images generated by DNN-based methods are very plausible but somewhat inaccurate. Human peripheral vision is lower in resolution than the central visual field and is therefore weak in distinguishing visual accuracy; therefore, DNN-based methods are suitable for generating context-images that meet the requirements of peripheral vision.

In this study, we investigated context-image generation using DNN. The contributions of this study are as follows:

1. Implementation of two new DNN-based methods to generate context-images

2. Generation of natural-looking images as another heavy method and the discovery of the limitations of our methods

3. Ensuring a processing speed that is appropriate for real time generation (30 fps) and faster than conventional methods

4. Providing results and feedback from user tests, showing that the same effects as the Illumiroom system can be obtained for peripheral projection of existing context-images, as well as the type of videos are well-suited our system

RELATED WORK

Generating the Context-Image

If the context-image already exists, such as in games, PC work, and movies filmed with dedicated context-capturing equipment, we do not need to create it [3, 10]. However, in the case of existing legacy content or content shot without dedicated equipment, it is necessary to estimate and generate context-images only from video data. Philips's AmbilightTV [22] is the simplest implementation for creating context-images. The TV has a series of color LED strips on its edges, and the colors of the LEDs dynamically change to match television contents. In a follow-up project, Philips created AmibLuxTV [22] which projects an enlarged image of the television on the wall by a projector attached to the back of the television. Infinity-By-Nine [12] is a CAVE [4] -like system with three projectors and screens around a television. The context-image is generated by their method using optical flow, color analysis, and heuristics. Aides, et al. [1] applied the PatchMatch algorithm developed from an original work by Avraham and Schechner [2] to the extrapolation video to an image's peripheral area. Although the videos generated by this method seem very natural depending on the scene, it takes a few minutes of processing time per frame. Turban, et al. [16] developed a lighter algorithm than the PatchMatch algorithm, but the generated images are much more artificial than those generated by PatchMatch. As described above, methods have not yet been proposed for generating sufficient context-images with respect to processing speed and naturalness.

Image Completion by Deep Neural Network

In recent years, image completion technologies using Deep Neural Networks, especially convolutional neural networks (CNN), have experienced significant progress [7, 13]. Image completion means to fill a "hole" and to attempt to restore lost image data. The model proposed by Iizuka, et al. [7] succeeded in complementing a natural image such that it could not be recognized as "filled" at a glance. We assume that the peripheral imagery can be filled in approximately the same way by applying a mask to the peripheral portion and treating it as a "hole."

Peripheral Vision

The human fovea occupies 1% of the retina. The fovea is densely populated with photoreceptors in this narrow region, allowing high-resolution vision. On the other hand, the density of the photoreceptor decreases outside this narrow region, and the perception of visual signals becomes much coarser. Since DNN is efficient at constructing very plausible images, it is compatible with this property of human eyesight.

CONTEXT-IMAGE-GENERATION METHODS

Pix2pix

In this research, we used pix2pix in our methods. Pix2pix [23] is an implementation of the paper "Image-to-Image Translation with Conditional Adversarial Networks" by Iola, et al. [9]. Pix2pix is open-source and available here: <https://github.com/phillipi/pix2pix>. Since we use this pix2pix almost by default, we include a summary of the Technical commentary in this paper. Pix2pix is a DCGAN [14]-based neural network that learns the conversion method between two images. The user prepares a dataset including image pairs and pix2pix learn how to convert between images. When you input the image, the user wants to convert after learning, it outputs the converted image according to the style pix2pix has learned. In their original paper, use cases in various applications are shown, such as "street scene from label," "black and white image to color image," "from aerial photograph to illustrated map," etc. To use published implementation, users only have to prepare a dataset of the pre-conversion/post-conversion images which the user forms into a resolution of 256×256. Pix2pix uses a Generative Adversarial Network (GAN) [5]. The generator produces data similar to training data to fool the discriminator. The discriminator then tries to judge whether the received data is training data or confederate data. It is theorized that the generator can produce data sufficiently similar to the training data to converge the discriminator's success rate to 50%. Pix2pix is also a type of conditional GAN [11], one that aims to improve generation-accuracy by adding additional conditions to the input, including meaningful vectors, images, and characters. In pix2pix, the generator receives inputted pre-conversion images as a condition and generates a post-conversion style image to fool the discriminator. In this paper, we adopted pix2pix for its ease of use and versatile performance.

Generation Method

We propose pix2pix-based methods to generate context-images, which we explain the basis of here. First of all, at the learning stage, we prepare a data set that collects cropped images and original images pair with resolutions of 256×256 (as shown in Figure 2). The "Cropped image" is an image cropped from the original images by 25% on the top, bottom, right, and left margins. Pix2pix learns how to convert from the cropped image to the original image. Using this method, pix2pix learns how to fill the black part with a natural image. In other words, pix2pix learns how to generate context-images. Based on this idea, this paper proposes two methods that differ in the way data sets are collected.

Figure 2. An example input image of pix2pix. The image was taken from the Place2 dataset [19].

Method 1: Category-limited method

The first method is the "category-limited method." This method relates to learning with image data sets that are divided into categories. For example, image data sets for machine learning are labeled "Mountain," "Field," etc., respectively. In other words, they are categorized by labels. By creating a data set for each category and undergoing learning, a model optimized for images belonging to that category is obtained. If the model has learnt with an image data set including "ocean" images, it outputs an image with a natural context-image when you input an image of an ocean. Conversely, if the categories of the data set pix2pix learned and that of the videos that you want to process do not match, you will not get good results. Therefore, to successfully generate images, it is necessary to select an appropriate learned model for the image to be processed. Due to the method's nature, it is difficult to use it for images in which various scenes like movies are frequently replaced.

In this study, we created six models, primarily using the place2 dataset [19] and videos collected from video sharing sites. Because the collected videos were often 16:9 or 4:3 aspect ratio, we change the aspect ratio to 1:1 for pix2pix. The correspondences between the model names and the datasets used for learning are shown in Table 1. In the "racing" model, training images are cropped up, down, left, and right 15%, because if they are 25% cropped, the feature often appears in the images as almost only 'ground,' and the scenery containing the sky and the course was not generated correctly.

Model	Dataset used for learning
Ocean	Images extracted from movie data totaling 5 hours and 2 minutes collected on video sharing website, one frame at a time (6046 images)
Mountain	"Mountain" in Place2 dataset (5000 images)
Field	"Field-cultivated" in Place2 dataset (5000 images)
Racing	Images extracted one frame at a time from capture video of racing game (3394 images)
Desert road	"Desert road" in Place2 dataset (5000 images)

Table 1. This Table shows what dataset was used for each model learning.

Method 2: Recursive Method

The second method is "Recursive method." In this method, the image itself that you want to process is used as training data. First, we extracted frames from the video and resized them to 256 x 256 for input to pix2pix. We then create a training dataset consisting of the original images and cropped images from the resized images. With a dataset made by this process, it is possible to generate context-images in almost the same way for any video. In other words, there is no need to undergo the procedure of choosing the appropriate model that was necessary for the first method. Compared to the first method, this method tends to output good result for videos where the cuts frequently change. In this paper, all training including Method 1 was done over 200 epochs, which is the default setting of pix2pix.

Patch Size

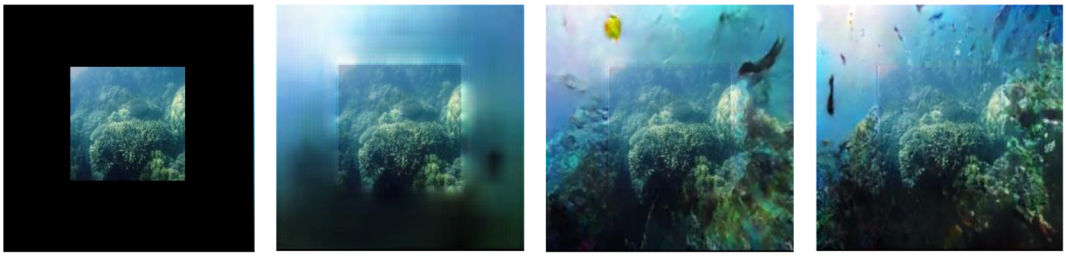
Pix2pix adopts patchGAN. Instead of putting the entire image into the discriminator, by cropping the image into patches, PatchGAN reduces the number of learning parameters while leaving global judgment intact, enabling more efficient learning. Changing the patch size produces different results. The smaller the patch size, the more local judgment is required. The larger the patch size, the more global judgment is performed. As a general parameter for various tasks, a setting of 70 × 70 patch size is used for pix2pix. However, to find the patch size that is most suitable for the context-image generation, we investigated four different conditions: from 1 × 1, 16 × 16, 70 × 70, and 256 × 256. The results are shown in Figure 3. Because the 256 ×256 patch size results in the most noise-free and natural video generation possible, we used it in all the experiments reported in this paper. When pix2pix is used with the default 70 × 70 setting, the discriminator's structure is C64- C128 - C256 - C512. When this is changed to a patch size of 256 × 256, the discriminator's structure is C64 - C128 - C256 - C512 - C512 - C512.

EXPERIMENTAL RESULTS

Context-Image Generation results

Figure 4 shows the results of the context-image generation using Method 1. We can see that detailed textures such as clouds, surfaces of rock and coral are expressed in the context-image areas.

Figure 5 shows the results of generating peripheral vision images for four images using both Methods 1 and 2. Method 2 has parts where noise is present, but it reproduces colors and textures atypical to images compared to Method 1.



input patch size: 1 patch size: 70 patch size: 256

Figure 3. An example input image of pix2pix. The image was taken from Place2 dataset [19]. 1x1 patch makes the image blurred. 70x 70 patch makes a natural image but it is a bit noisy and the boundary stands out.

The results compared to existing methods are shown in Figure 6. As the place, size, and aspect ratios generated in each paper [1, 16] are different, we adjust the size per the extant goal of doubling the original video. Therefore, we set the aspect ratio to 1:1. Results not listed in these papers are not shown in Figure 6 and are left blank. Moreover, because the original videos are too short, learning cannot take place in Method 2. The results of Aides, et al. [1] and Turban, et al. [16] are posted as "Multiscale," and "Extrafoveal" respectively, after the titles of their papers. In the image of semitrailer, Multiscale [1] can generate predictions for tracks outside the original image range, but these cannot be generated with Extrafoveal [16] and our Method 1. This is because Extrafoveal [16] and our Method 1 use only the current frame for prediction and generation, whereas in the Multiscale method the frame that is the basis of prediction generation should be a number of frames before and after. However, there is a possibility that this can be resolved by inserting vector information as an input, not to only the frame that needs to be predicted or generated, but also to several tens of frames before and after.

Computational Time

The time needed for processing 1200 images was around 30 – 35 seconds on models created by our methods. In this study, we evaluated our method on a PC using an Intel Core i7-5930K CPU with NVIDIA GeForce TITAN X GPU. All the models we generated in this paper could process over 30 fps, which is common for TV contents or movies. As a characteristic of the neural network, the processing time basically does not depend on either the number of learning images or model.

Our methods are faster than the Multiscale method [1]. To process one frame, our methods take about 0.025 seconds on GPU and 0.262 seconds on CPU, Intel i7-5930K @ 3.5GHz with 64GB RAM on Ubuntu ver.16.04. In contrast, the Multiscale study reported several minutes per frame, with an Intel Core 2 Duo E8400 @ 3 GHz with 4 GB RAM. The authors more precisely reported 3 to 15 minutes for the three different sequences in their paper. Since the Multiscale's extrapolation (342 × 234 to 1280 × 1024) is about 4 times the size used in our case, and considering that the processing time correlates linearly with the size of the image, to first approximation, this would suggest the Multiscale method should take approximately 45 - 225 seconds to process one frame. We can also compare the speeds in the task by extrapolating the image by a factor of 16. In our method, this can be realized by processing twice. The processing time is then 0.262 × 2, which is 0.524. Compared to the 180-900 seconds required for multiscale, our method is approximately 360 times faster.

Although we can't make exact comparisons because experiments were not done in the same environment and the Multiscale method's speed varies according to the settings such as the number of frames to be searched and the resolution of the input image, our methods are much faster

Ocean Desert road Mountain Field

Figure 4. These are images with their context-images processed by Method1. The central part in a red square of each image is the original input image. The model used is shown below each image. The image was taken from Place2 dataset [19].

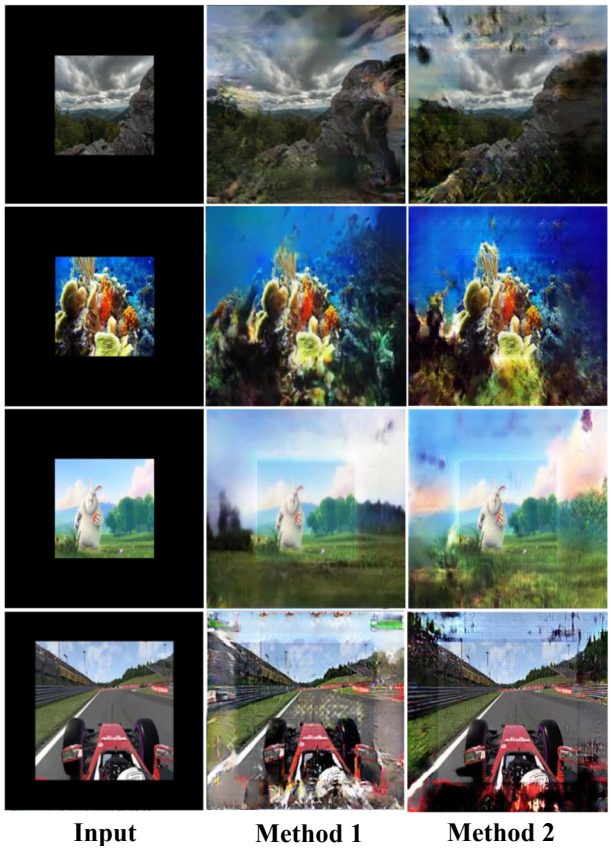


Figure 5. Results of Method 1 and Method 2. (Mountain : ©2015, Garrett Martin / https://vimeo.com/139927695 Ocean : ©2017, Eric Falconi / https://vimeo.com/199482708 Field : ©2008, Blender Foundation / www.bigbuckbunny.org Racing : F1(TM)2016)

than Multiscale even taking into consideration the CPU performance difference.

Flicker Problem

Our pix2pix-based method sometimes caused a flashing flicker effect when the generated frame was returned to a moving image in a sequence. This problem occurs because the image generation process is performed for each frame and the generation results differ slightly even between adjacent frames. To solve this problem, we use a time smoothing filter that averages five frames, including the target frame and two frames before and two after. This is repeated for each frame in the sequence. You can see the filter's effects here <https://youtu.be/wdE8a8woSx4>. This technique almost wholly solves the flashing flicker problem caused by differences in brightness. However, the flicker caused by differences in the positions of objects remains after applying our filter.

USER STUDY

We conducted a user study to investigate the quality of experience (QoE) of peripherally projected context-images generated by our method. Because there is currently no defined protocol to evaluate QoE for peripheral projection,

Figure 6. Comparison with other algorithms. Test data (Mountain, Underwater, Dustroad, Semitrailer) is opened by Aides.et.al [1].

we defined a specific protocol based on two other studies about a peripheral projection: Illumiroom and Extrafoveal.

Setup

For the user study, we implemented a television with a peripheral presentation of the context-image by a projector. The television played the 720p "inside" contents, and the projector played the "outside" context-image on the peripheral area of the TV. A user sat on a chair that was 1.9 m from a 55-in TV installed on a wall; their field-of-vision became 36°. A projector was mounted on a shelf opposite the TV, 6 m from the TV, and 2 m above the ground. The setup of the experiment is shown in Figure 7. We recruited 12 individuals of ages 20-40, including two female participants. Each round was kept under 30 min to avoid boredom and fatigue, which could adversely affect results.

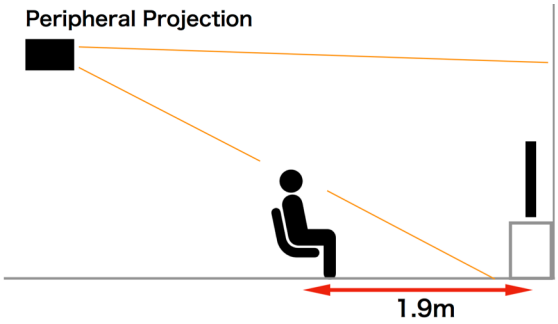


Figure 7. The peripheral projection setup used in user study.

Content

To investigate QoE of our system and what kind of videos were most effective, we selected six scenes from different types of videos made up from 2 movies, one music concert video, one first-person point of view video captured by an action camera, one drama, and one video capturing a natural scene. Detailed information on each movie is explained below:

Movie 1: A movie depicting the conflict between humans and indigenous peoples living on a certain planet. We chose a battle scene with explosions and intense camera movements.

Bike: A video captured by an action camera attached to the head of a motocross bike rider running around in a small city. It is in first-person viewpoint movie with the quickly-changing scenery.

Music Concert: This is a video of a world-famous band in concert. The hall is vast, and the video displays characteristic camera work that alternately shows performers and spectators.

Drama: "Drama" is a scene where two lawyers quarrel in a narrow room of an office from a drama video depicting American lawyers. The two people are alternately photographed, although there are many cuts, the camera movement is minimal.

Movie 2: "Movie 2" is a Hollywood movie with a pirate theme. The scene used depicts pirates and soldiers are fighting and an explosion. Slow motion is often used, and the movement of the camera is small.

Scuba Diving: "Scuba Diving" is a first-person video capturing during underwater scuba diving. The movement of the camera is slow and there are few cuts.

The videos were processed using Method 2. In this study, all of the frames in about 2 minute scenes were used for learning. Thus, about 5000 to 6000 frames were used for every model. Furthermore, to obtain more angle of view, we applied some light-weighting effects of widening on the "outside" videos projected in the peripheral area. We cropped the "inside" area, applied lens distortion removal in reverse and Gaussian blur on edge. The image processing is shown in Figure 8. The reason why we use the light-weight effects instead of twice processing using our method is to widen the angle of view while keeping fast processing time and naturalness of generated images. If the too wide area was cropped to predict a larger context-image, input information decreases and the quality of context-image drops. On the other hand, if we process with our method twice, it was impossible to keep processing speed above 30 fps on our PC.

Procedure

The participants were divided into four groups to experience the videos in a different order. To exclude the effect on results due to fatigue and boredom, all the video had been reduced to approximately 90-second scenes. Experiments

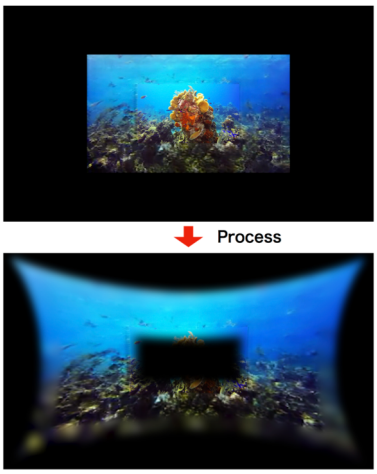


Figure 8. The image process applied on "outside" video.

took place in about 30 to 40 minutes (ITU recommendations advise sessions of 30 minutes [15]). Firstly, the subjects were explained to watch six sets of videos and to answer the questionnaires about their experiences about videos they watched. A set consisted of "Effect 1" and "Effect 2". "Effect 1" was a video without its context-image, i.e., a simulated traditional visual experience. To prevent creating a bias when switching from peripheral light to no peripheral light, a gray background image was projected around the television. "Effect 2" was a video with a peripheral projection of its context-image. Every time viewing of each set was completed, the subject answered ten questionnaires on a 7 Likert-type scale regarding each effect. When user finished answering all of the questionnaires, the next set started. After experiencing six sets of experience, participants were required to answer questions about the overall experience.

Following the studies of Illumiroom [10] and Extrafoveal [1], we chose evaluation items consisting of "Consistency," "Immersion," "Comfort," "Emotion," "Enjoyment," and "Sickness." In other studies, evaluation items and questions corresponded one-on-one, but to prevent the result from changing due to fluctuation in expression during translation into Japanese, we prepared two questions with different expressions on the items, "Consistency," "Immersion," "Comfort," and "Emotion."

[Enjoyment]

- Q1: Did you enjoy watching?

[Sickness]

- Q2: Did you feel sick during viewing?

[Emotion]

- Q3: Were you impressed by watching the video?
- O4: Were you excited by watching the video?

[Immersion]

- Q5: Did you feel like you entered the image world?
- Q6: Did you feel present?

[Comfort]

- Q7: Was the experience comfortable?
- Q8: Did the video on the outside of TV disturb viewing?

[Consistency]

- Q9: Did you feel that the inside and outside of the TV were connected?
- Q10: Did you feel any incompatibility between the picture on the TV and outside of the TV?

The questionnaires asked after the six sets of watching for overall evaluation were as follows:

- Please rank them in the order in which you felt that Effect 2 was most effective.
- Please tell us the reasons for this order and the impressions about the system.

Results & Feedback

For a summary of results see Figure 9. When the participants selected two options, a numerical value with a large absolute value was adopted. The ratings for all videos are summed in each question in Figure 9, because no significant trend difference was seen for items other than drama. The result in Effect 2 in Q9 "Did you feel that the inside and outside of the TV were connected?" shows a higher rate of agree than Effect 1 (Effect 1: 6%, Effect 2: 68%). In response to Q10 "Did you feel any incompatibility in the picture on the outside of the TV?", although the ratio of agree increased with Effect 2 (Effect 1: 24%, Effect 2: 39%), the rate of disagree was higher than agree and the median was nearly neutral. From these, it seems that context-images generated by our method appear to be found consistent with "inside videos." Regarding the consistency of "outside video," feedback included:

・"*When the color of "outside" video matched that of "inside" video, I thought they were connected. I thought this system was especially useful for the video with vivid colors and dark backgrounds."*

From the results of the questions about Immersion (Q5, Q6), it seems that the participants found the experience immersive. Regarding Immersion, the feedback below was received.

・"*When the "inside" image was with the color or brightness changes, the outside image seemed more realistic. The image with a strong aperture (such as when the person is focused, and the background is blurred) was more realistic. The flame increases the sense of presence."*

・"*The larger the area of the background occupying the screen ("inside" video), the more I felt the immersive feeling increased."*

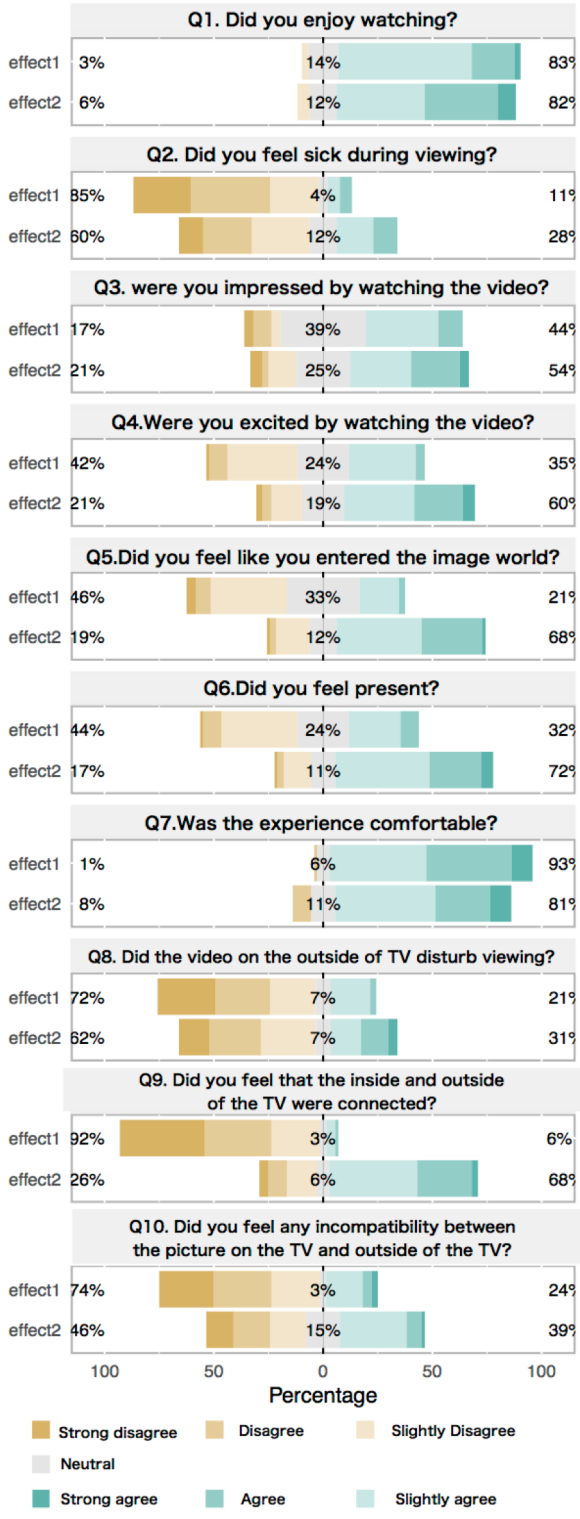


Figure 9. Percentage of each answer to each question. The number in the middle of bar indicates the percentage of answers "Neutral". The percentage on the left side of the bars indicates sum of the percentage of answers "Strong disagree," "Disagree," "Slightly Disagree". The one on the right side indicates sum of answers agreeing.

• "*Because when I see things that were spreading like the sea or fire, I felt immersion.*"

In two questions about Emotion (Q3, Q4), the ratio of agree increased in both with Effect 2.

In Q2, which asked about Sickness, Effect 2 had a higher agree ratio than Effect 1. Although there is a possibility that it may cause more sickness compared to the case without peripheral projection, the ratio of disagree is still much higher than that of agree. Therefore, it is not thought to cause severe sickness. In Illumiroom study [10], "F+C Full" which is closest to our system, augmented visual experience ("I was fun", "I was moving," "I was in the game," "The game and the room were connected") without enhancing sickness. Our user study showed that our system augments the experience almost as good as Illumiroom.

From the contents of the feedback showed below, it turns out that many users felt flicker as an essential factor. They made grounds for negative evaluation that there were many flickers, and based on the positive one that flicker was small.

• "*I felt it was a system that fits live images because black flickers are anxious.*"

• "*Because the drama was a calm scene, it did not match most.*"

• "*The concert was dark, and the flicker of the peripheral area was not disturbing. So, I could feel immersion.*"

• "*Since "movie1" and "diving" were slow images, the degree of feeling discomfort was high compared with others.*"

The results of the ranking by participants are shown in Figure 10. Participants found the peripheral projection some of the most useful for "Music Concert." For instance, these were some comments for their reasons:

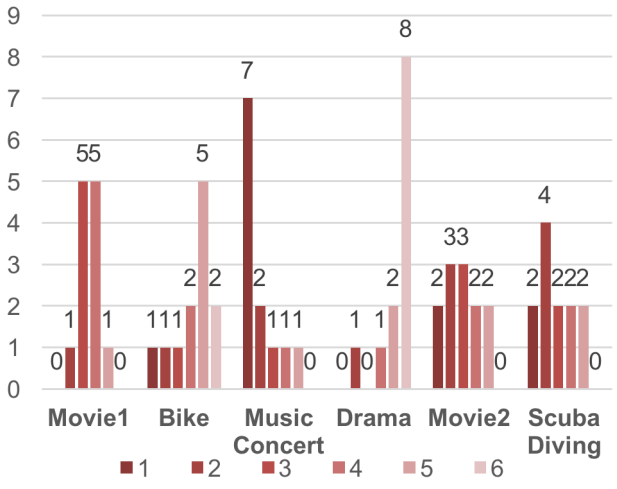


Figure 10. Total ordering of each video. One participant answered only 1st to 3rd, to adjust the total of points, we assigned "5" to the rest of the video.

• "*I was concerned about black flickers, so I felt it was a system that fits the concert video.*"

• "*The concert was dark around; there was realistic, I did not care about flicker. I felt that the video that changed color and brightness* was more realistic."

In contrast, "Drama" was often ranked in 6th place. Some of the responses to questions about "Drama" are shown in figure 11. From the rating of the Consistency (Q9), it seems that the context-images were considered unnatural by participants. Furthermore, from the result of Q8, it seems that the projection of the context-image disturbed viewing. The experience of "Drama" with Effect 2 tends to be rated negatively in other items. The following comments are feedback about "Drama."

• "*I felt it was rather disturbing for dramas.*"

• "*Because the drama was a calm scene, it did not match most.*"

DISCUSSION

Limitations and Future Work

In our study, it was found that our method could generate context-images that were considered consistent with the inside videos while keeping processing speeds that were fast enough for the real-time generation. However, several problems remain. First, in order to realize strict real-time processing, it is necessary to increase the generality of generation. Method 1 requires users to select suitable models for the video. On the other hand, Method 2 needs long learning times for each video. If no one has trained the model and there is no pre-trained model in the world, processing time should include the very long learning time. Therefore, we need to create a generic model that can properly process any image. We also found that the flicker will hinder the video experience according to our user study. It is also an issue that it is not possible to take into account temporal changes, and predictive generation of objects that frame in / frame out cannot be generated in our model.

Several proposals for these issues are conceivable. Although pix2pix is a wonderful model with a versatile performance for various tasks, it is not fine-tuned for our task. Therefore, a performance improvement is predicted by optimizing the network. For example, by inputting a plurality of frames, it is conceivable that they can be generated considering of temporal changes. In addition, since increasing the number of input frames means increasing the condition, it is expected that the generation result will be stable and that flicker can be prevented.

As a method for enhancing versatility and making a model capable of performing appropriate context-image generation for any image, as in the research of Iizuka et al. [8], combining the global features network will be useful.

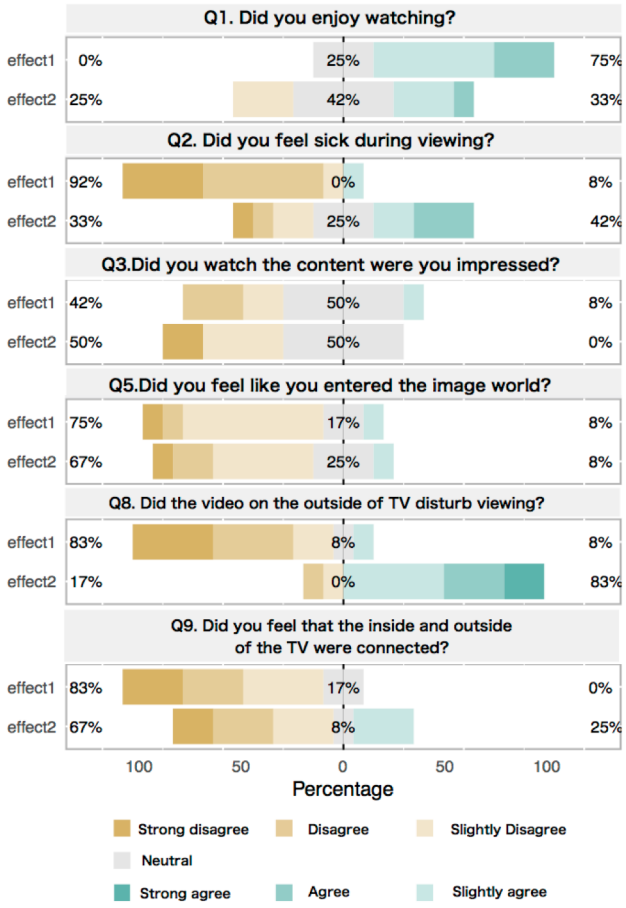


Figure 11. Percentage of each answer to some questions of "Drama".

DNN for HCI

We believe that DNN can make many contributions to HCI. DNN often provides high performance in recognition and generation tasks. For example, Improvement of accuracy in Gesture Recognition will advance some interactive systems [17]. In addition, models using DNN, especially Convolutional Neural Network, demonstrate high performance for tasks using images. Studies of generating haptic feedback from images [18] will greatly contribute to Virtual Reality Entertainment techniques. In this research, we adopted an Illumiroom-style application, but our idea could also contribute to the VR field. For example, 360-degree videos have disadvantages such as large size, choosing a playback environment, and difficulty in streaming. Therefore, by transmitting only a section of about 180 degrees and restoring the rest on the receiving side by our method, it may be possible that 360-degree video is more widely used.

CONCLUSION

We have described two methods using DNN to generate peripheral context-images for videos and have presented our results. With our methods, a patch size of 256 × 256 pixels allows the most natural reconstruction of images (i.e., the discriminator should see the images globally in the context-image generating task). Our methods can generate textured images as fine as the Multiscale method [1] keeping processing speed faster than 30 fps, which is sufficient for real-time generation of videos at 30 fps. Our user study showed the peripherally projected images generated by our methods are sufficient for a positive enhancement of visual experiences. However, sequenced images generated by our methods have a flicker and noise problems caused by differences between frames. Problems on the versatility of model and consideration of temporal changes also remain. To solve these problems, further optimization of the network for our task will be useful. We expect not only our research but also that DNN will make a further contribution to HCI.

REFERENCES

1. Amit Aides, Tamar Avraham, and Yoav Y. Schechner. 2011. Multiscale ultrawide foveated video extrapolation. In *2011 IEEE International Conference on Computational Photography (ICCP)*, 1–8. https://doi.org/10.1109/ICCPHOT.2011.5753126
2. Tamar Avraham and Yoav Y. Schechner. 2011. Ultrawide Foveated Video Extrapolation. *IEEE Journal of Selected Topics in Signal Processing* 5, 2: 321–334. https://doi.org/10.1109/JSTSP.2010.2065213
3. Patrick Baudisch, Nathaniel Good, and Paul Stewart. 2001. Focus plus context screens. In *Proceedings of the 14th annual ACM symposium on User interface software and technology - UIST '01*, 31. https://doi.org/10.1145/502348.502354
4. Carolina Cruz-Neira, Daniel J. Sandin, and Thomas A. DeFanti. 1993. Surround-screen projection-based virtual reality. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques - SIGGRAPH '93*, 135–142. https://doi.org/10.1145/166117.166134
5. Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative Adversarial Nets. *Advances in Neural Information Processing Systems 27*: 2672–2680. https://doi.org/10.1017/CBO9781139058452
6. Yixiang Huang, Xuan F. Zha, Jay Lee, and Chengliang Liu. 2013. Discriminant diffusion maps analysis: A robust manifold learner for dimensionality reduction and its applications in machine condition monitoring and fault diagnosis. *Mechanical Systems and Signal Processing* 34, 1–2: 277–297. https://doi.org/10.1016/j.ymssp.2012.04.021
7. Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. 2017. Globally and Locally Consistent Image Completion. *ACM Transactions on Graphics ACM Trans. Graph. Article* 36, 13. https://doi.org/10.1145/3072959.3073659

8. Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. 2016. Let there be Color!: Joint End-to-end Learning of Global and Local Image Priors for Automatic Image Colorization with Simultaneous Classification. *ACM Transactions on Graphics (Proc. of SIGGRAPH 2016)* 35, 4.
9. Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. 2017. Image-to-Image Translation with Conditional Adversarial Networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5967–5976. https://doi.org/10.1109/CVPR.2017.632
10. Brett R. Jones, Hrvoje Benko, Eyal Ofek, and Andrew D. Wilson. 2013. IllumiRoom: Peripheral Projected Illusions for Interactive Experiences. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13*: 869. https://doi.org/10.1145/2470654.2466112
11. Mehdi Mirza and Simon Osindero. 2014. Conditional Generative Adversarial Nets. Retrieved January 9, 2018 from http://arxiv.org/abs/1411.1784
12. Daniel E Novy. 2013. COMPUTATIONAL IMMERSIVE DISPLAYS. Retrieved January 9, 2018 from http://excedrin.media.mit.edu/wp-content/uploads/sites/10/2013/07/novyms.pdf
13. Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. 2016. Context Encoders: Feature Learning by Inpainting. https://doi.org/10.1109/CVPR.2016.278
14. Alec Radford, Luke Metz, and Soumith Chintala. 2015. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. https://doi.org/10.1051/0004-6361/201527329
15. Recommendation Itu-.Methodology for the subjective assessment of video quality in multimedia applications. Retrieved January 9, 2018 from https://www.itu.int/dms_pubrec/itu-r/rec/bt/R-REC-BT.1788-0-200701-I!!PDF-E.pdf
16. Laura Turban, Fabrice Urban, and Philippe Guillotel. 2017. Extrafoveal Video Extension for an Immersive Viewing Experience. *IEEE Transactions on Visualization and Computer Graphics* 23, 5: 1520–1533. https://doi.org/10.1109/TVCG.2016.2527649
17. Saiwen Wang, Jie Song, Jaime Lien, Ivan Poupyrev, and Otmar Hilliges. 2016. Interacting with Soli: Exploring Fine-Grained Dynamic Gesture Recognition in the Radio-Frequency Spectrum. https://doi.org/10.1145/2984511.2984565
18. Kentaro Yoshida, Seki Inoue, Yasutoshi Makino, and Hiroyuki Shinoda. 2017. VibVid: VIBration Estimation from VIDeo by using Neural Network. 37–44. https://doi.org/10.2312/egve.20171336
19. Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. 2017. Places: A 10 million Image Database for Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*: 1–1. https://doi.org/10.1109/TPAMI.2017.2723009
20. SCREENX. Retrieved January 9, 2018 from https://screenx.co.kr/
21. Ready 2 Escape – The ultimate immersive cinema experience. Retrieved January 9, 2018 from https://ready2escape.com/
22. Philips TV. Experience Ambilight | Philips. Retrieved January 9, 2018 from https://www.philips.co.uk/c-m-so/televisions/p/ambilight
23. Phillipi. 2017. phillipi/pix2pix. (December 2017). Retrieved January 8, 2018 from https://github.com/phillipi/pix2pix