View Interpolation using Neural Network for Fullsphere 3D Telepresence

Kazuki Sakai
Toyama Prefectural University
Master's Programs Electrical and Computer Engineering
Toyama, Japan
t755008@st.pu-toyama.ac.jp

Takayuki Nakata
Toyama Prefectural University
Electrical and Computer Engineering, Toyama, Japan
nakata@pu-toyama.ac.jp

Hironari Mathuda
Toyama Prefectural University
Electrical and Computer Engineering, Toyama, Japan
hmatsuda@pu-toyama.ac.jp

***Abstract*— As the Internet and video equipment develop, service using VR technology will become popular. Telepresence use those technologies. This is expected to be applied in various fields, e.g. tourism. Multiple users can use the telepresence system at the same time by using the full sphere(Omnidirectional) camera. However, stereoscopic vision is impaired, because the camera is fixed at the time of recording and cannot respond to the position movement of the user's point of view. This paper aims to preserve stereoscopic vision in the fullsphere telepresence system by estimating the image of the position of the viewpoint of the user from the existing fullsphere image by the neural network. Images of the current line of sight direction is taken out from fullsphere cameras arranged in concentric circles and the image of the midpoint is estimated from the images of the two viewpoints by using the neural network. By using the image obtained at the midpoint, this processing is performed recursively to retrieve the image of the viewpoint of the user. In this paper, we experimentally verified whether the midpoint viewpoint can be estimated by neural network.**

Keywords—telepresence; fullsphere image; neural network; image processing;

I. INTRODUCTION

As the Internet and video equipment develop, service using VR technology will become popular. There is telepresence using those technologies. Telepresence can show users that the immersive scene of the remote place without actually going to there through Virtual reality head mounted display. Although it is applied in video distribution service etc. in VR, even if it is seen through VR HMD, there are few stereoscopic images. This is because the position of both eyes changes when the user rotates his / her head, but the viewpoint of the presented images does not move. It needs images of right and left eyes to present Stereoscopic movie to users, and when the user rotates his / her head, not maintaining the positional relationship between the left and right viewpoints impairs the user's realistic sensation. Recent 3D fullsphere images follow user's head rotation by setting many cameras on concentric sphere [1]. However, unnatural interpolation and overlap occur on border of camera.

In this work, we aim to eliminate unnatural interpolation of the camera image which loses realism when viewing by humans by using a neural network.

II. METHOD

A. System

Put the fullsphere cameras concentrically. From those fullsphere images captured by each, take out the image of the direction the user is currently viewing. For example, when four cameras are used, these viewpoints are assumed to be view point A, B, C and D. Based on these viewpoints, points closest to the eye position with respect to the line of sight direction is defined as the right eye V_R and the left eye V_L. These images of viewpoint are generated by interpolation model by neural network.

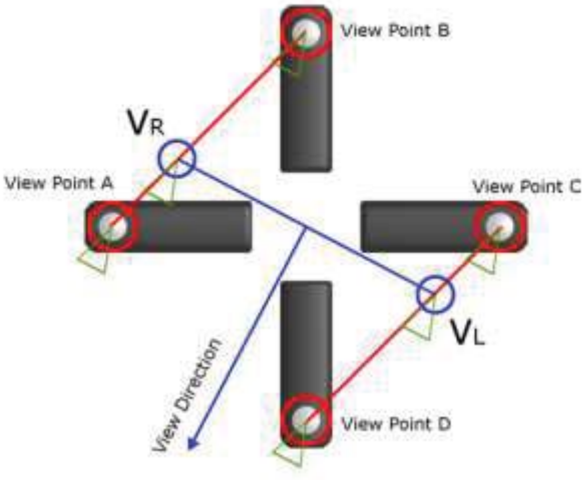

Figure 1: System appearance

B. Interpolation using neural network

Interpolation model by neural network in our paper is trained to generate intermediate image from two same direction images.

For example, when images in the same direction are captured by camera A and camera B and input to the neural network, an image of the direction viewed from the midpoint C of the line segment connecting camera A and camera B is output (Fig.2).

978-1-5386-2615-3/18/$31.00 ©2018 IEEE

Similarly, by inputting the image of the camera A and the outputted middle point C to the neural network, it is possible to output an image of the direction viewed from the midpoint C 'between the camera A and the midpoint C (Fig.3). By processing this recursively, it is possible to interpolate the image between camera A and camera B anywhere.

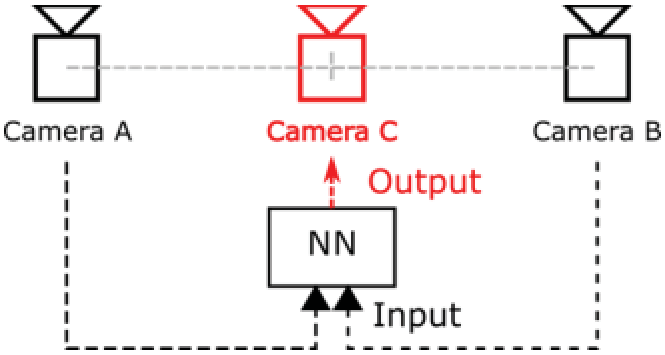

Figure 2: Image interpolation by neural network

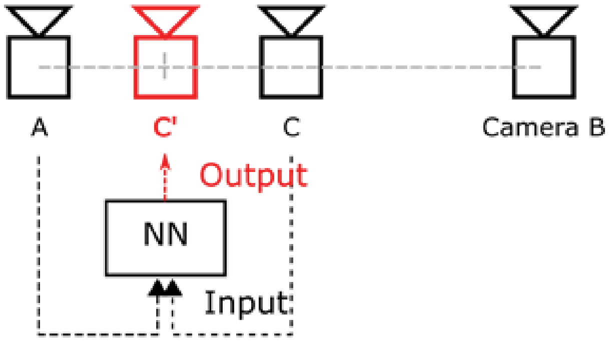

Figure 3: Interpolate arbitrary viewpoint with recursive processing

C. Structure of neural network

The proposed neural network reduces the computational time for generating the output image from the input image and reduces the model learned by the network in order to perform real-time streaming. Input to the network is input of an input image A satisfying the epipolar constraint and a pixel lines of horizontal row of pixels of the input image B connected (Fig.4). Since the photographed data set satisfies the epipolar constraint by camera calibration, comparison of images can be performed only in the horizontal direction. Therefore, the output image is pixel lines in a horizontal row of the intermediate viewpoint between the input image A and the input image B. These pixel lines are finally combined into one image. For example, an image of 640x480 is once divided into 640x1 image groups, each divided image is input to the network, and the output images are combined to form a single 640x480 image.

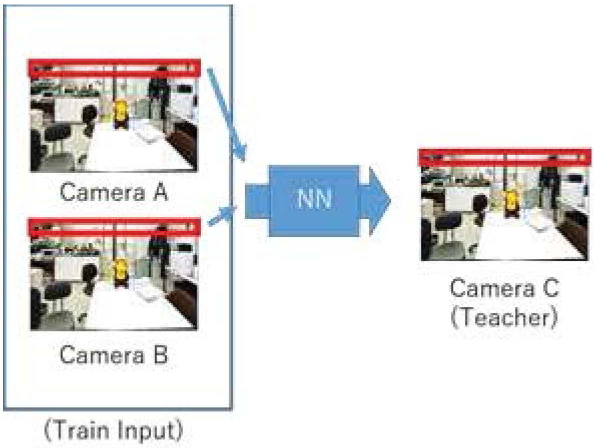

Figure 4: Structure of input image

To learn the network, use a dedicated camera arrangement. It prepares three cameras same as those used in the system and arranges them on a straight line. The central camera is placed at the midpoint of the two cameras. The cameras at both ends are set as input images and the central camera is taken as the teacher image for the output image of the network. Shoot the landscape with this camera arrangement and learn the network.

III. Experiment And Discussion

A. Experience

Before dealing with fullsphere images, we verified whether a general camera (DFK22BCU03) was used to correctly estimate the intermediate viewpoint. The input image for the training and the teacher image were taken using the university campus. The resolution of all cameras are 640x480. In order to simplify learning verification, an image in which the teacher image is shifted by 5 pixels to the left and right is set as an input image, and the RGB color image is converted into a gray scale image of 1 channel. We use a total of 40,000 training data sets and test data sets for the training data set.

Figure 5: Image of the university campus

B. Layered Design of Neural Network

The design of the neural network is as follows. In this experiment, we designed and compare three-layer neural network and single layer neural network. We use mean square error for the loss function.

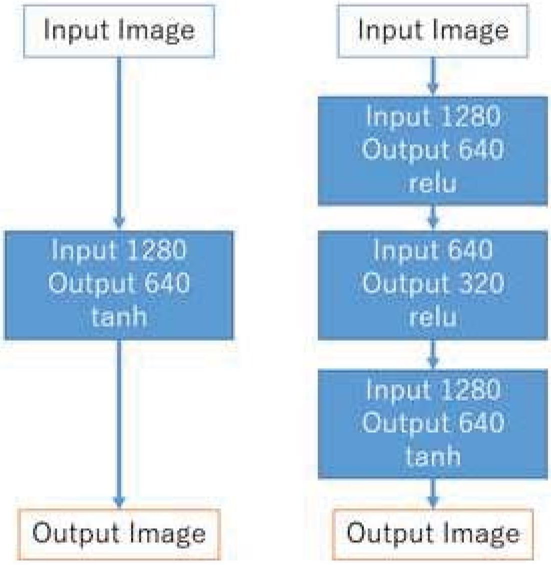

Figure 6: Layered Design

(Left: single layer, Right: three-layer)

C. Result

The learning results by those neural network are shown below. Fig.7 and Fig.9 show the mean square error of luminance in the 640 × 1 image in the learning process. However, the luminance value of the image is normalized to 0 to 1. Fig.8 and Fig.10 are comparison between the teacher images and the output images of the network. The blue line shows the output image and the red line shows the teacher image.

In the three-layer neural network, the final error is 12.85389, which indicates that there is a difference of 36.1 points per pixel in the luminance value of 256 tones. In the case of single layer, it is 12.67908, which indicates that there is a difference of 35.9 points per pixel at the luminance value of 256 tones. It turned out that there is only a difference of around 1 per pixel between 3-layer and single layer network. In addition, as shown in Fig.8 and Fig. 10, the amplitude of the value of each pixel is smaller in the three-layer than in the case of single layer, change in luminance is looked more smooth than teacher image.

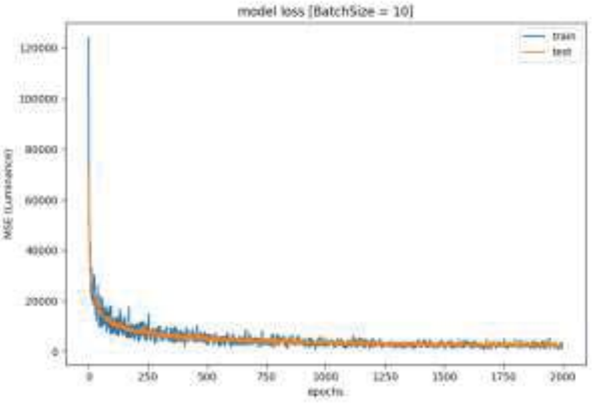

Figure 7: [Single Layer] Changes in error due to learning

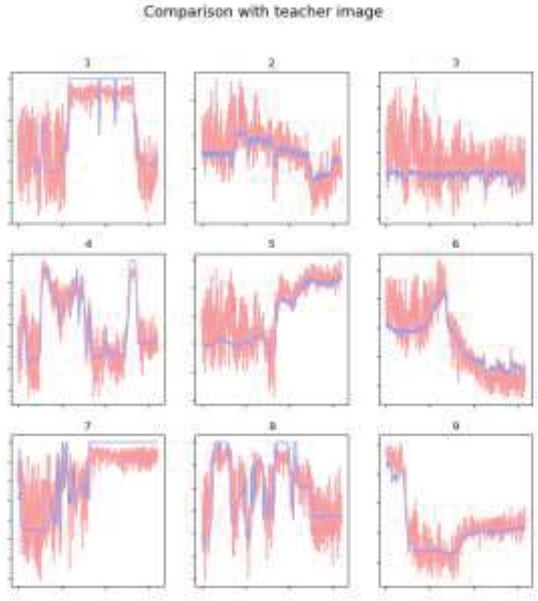

Figure 8: [Single Layer] Comparison of output and teacher image

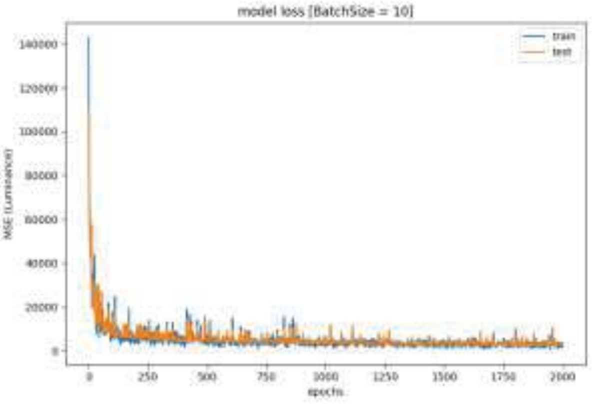

Figure 9: [Three Layer] Changes in error due to learning

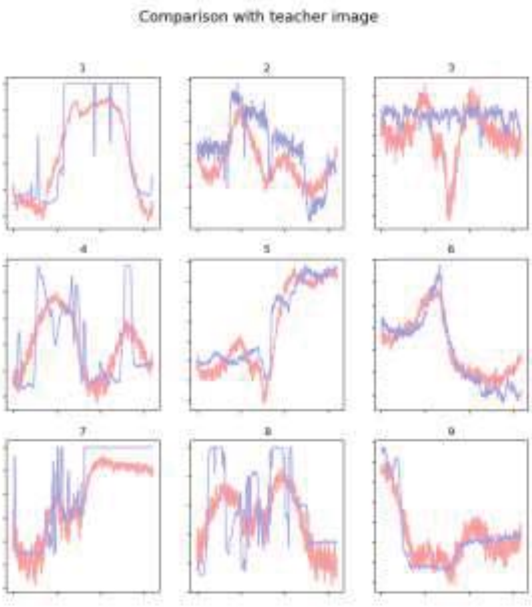

Figure 10: [Three Layer] Comparison of output and teacher image

The reason why a large error occurs is that tanh is used for the output layer and it is impossible to output a value close to the maximum value. Therefore, regardless of the number of parameters of the network, it is highly likely that this value is the highest precision when this activation function is used.

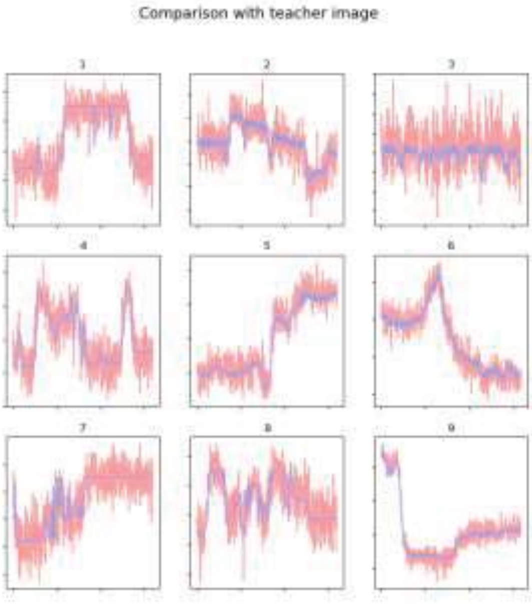

Figure 11: [Single Layer] Comparison of output and teacher image [maximum 0.5]

If the data set is deformed (0.5 times the data set) to the value in the range expressible by the activation function tanh, the learning result is improved (Fig.11).

However, even then, it does not converge to a perfectly correct value. Since it is confirmed that higher accuracy is present by manual adjustment, there is a high possibility that it is falling into a localized solution.

I. Conclusion

In the current neural network, in the problem of outputting an intermediate image from a simple pixel shift, even if the number of parameters is increased, the effect is small since the original problem is simple. In this problem probably one affine layer suffices. However, with the current optimization method, it is difficult to make the error almost 0, and it is known to fall into a local solution.

There are two ways to solve this problem, one is using a method other than the Estimated gradient descent method for the optimization method. Secondly, since it is an image on which high-frequency noise is superimposed on the currently output image, it is a method to reduce the error by smoothing by smoothing filter.

In the future, instead of using pixels simply shifted that used in this experiment, images with different amounts of pixels displacement depending on the depth of the object are used. So, there is a need to increase the number of parameters to improve the expressiveness of the network. Also, it is desirable to use a convolution neural network. This is because it is suitable for image processing. In that case, you should use an image with height instead of one pixel in the horizontal direction.

References

[1] Robert Anderson, David Gallup, Jonathan T. Barron, Janne Kontkanen, Noah Snavely, Carlos Hernandez Esteban, Sameer Agarwal, Steven M. Seitz, “Jump: Virtual Reality Video,” ACM Transactions on Graphics(Proc. of SIGGRAPH Asia 2016), 2016