Calculating Real-time Computer-Generated Holograms for Holographic 3D Displays through Deep Learning

Sheng-Chi Liu, Jin Li and Daping Chu*

Centre for Photonic Devices and Sensors,, Department of Engineering, University of Cambridge,
9 JJ Thomson Avenue, Cambridge CB3 0FA, UK
*Corresponding Author: dpc31@cam.ac.uk

Abstract: A deep learning method is proposed to calculate holograms in real-time. After training, it can generate holograms for all R/G/B channels within 10 msec. Simulation results confirm successfully reconstruct the target training and testing images.

1. Introduction

Digital holographic displays are widely recognised as a promising 3D display technology. Since this technique can reconstruct the wavefront of the target images, users can appreciate the displayed 3D images without unpleasant feelings. However, computer-generated holograms (CGHs), which the content to project holographic displays, require strong computing power to process massive amounts of data [1]. Therefore, CGH computations in real-time is a serious challenge for the technology. In order to reduce the computation time, this work proposes a real-time CGH generation method using deep learning. It is separated into two main concepts, data compression and data enlargement. With regard to compressing data, autoencoder is a useful artificial neural network for learning representations for dimensionality reduction [2-4]. It is an unsupervised learning method and Fig. 1a shows a deep neural network of a general autoencoder. It has two main parts, encoder and decoder. The former encodes original data to a small number of representation code which implies underlying information of the data. The latter on the other hand decodes from the representations to reconstruct the original inputs. In recent years, a new autoencoder was proposed, called variational autoencoder (VAE). It changes the latent variables from fixed vectors to probability distributions, adds constraints on the encoded representations and adopts another criterion of measuring the distance between the distribution and standard normal distribution. To speed the CGH computation process, this paper proposes a new VAE to encode CGHs to find the underlying information from the latent codes. Following that, we also introduce a new super-resolution technique for enlarging holograms to increase the resolution of the displays.

2. Deep Learning for CGH Computation

In the first part of the process, the proposed VAE contains 5 fully-connected layers for both the encoder and the decoder as shown in Fig 1a. A CGH with the resolution of 128×128 as input data, and the dimension of the input is reduced gradually through each layer and finally encoded in 16-variable code (z). Since a lower number of latent variables tends to cause poorer reconstructed performance, this work built the latent layer with 16 neurons. As for the decoder, the latent variables are decoded from 16 to the original dimension size.
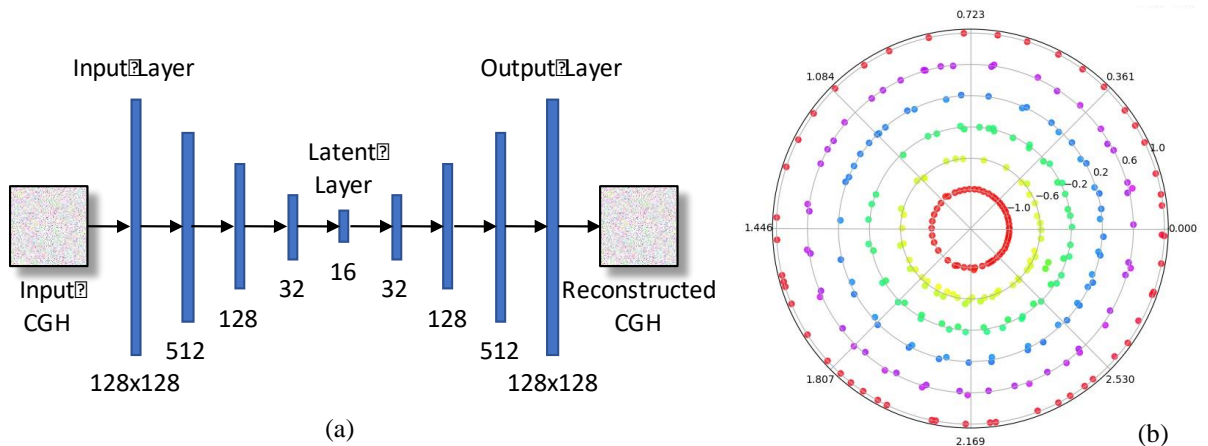

Fig. 1. (a) The architecture of autoencoder, and (b) the latent code (z_0 and z_8) in polar axis.

Before training the network, we have to build a training dataset to generate CGHs, and therefore two 3D Torus-Knot objects had built, and captured 50 images from various viewing angles by rotating the 3D objects. Including R, G and B channels, there are a total of 300 CGHs were computed based on the layer-based CGH computation method. By training the network to generate CGHs of the objects, Fig. 1b plots all the 300 data compress into the two latent variables (z_0 and z_8). Even though there are 16 variables in the latent space of the proposed network, the new VAE is able to successfully encode all the information in the two variables, z_0 and z_8, remain others with the value of zero. Figure 1b also shows all the 300 samples stay separately at around 6 different radial distances. This is because Samples of R, G, and B of the 3D objects have a roughly z_0 value of 1.0, 0.6, 0.2, -0.2 -0.6, -1.0 respectively, and these samples are distributed at various z_8 values. This indicates that we can sample a CGH of an orientation image of a specific channel of one of the two objects by given two numbers, z_0 and z_8. At the next stage, the sampled holograms can be enlarged through our CGH super-resolution neural net, and Fig 2 displays the reconstructed images of the super-resolved holograms.
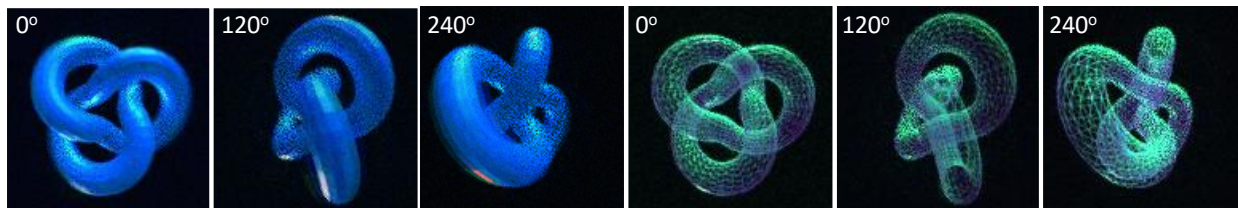

Fig. 2. Simulation results: reconstructed images which are Fourier transform results of the CGHs computed by our proposed deep neural network.

3. Holographic 3D Display System

A holographic 3D display is represented in Fig 3. The system uses a digital micromirror device (DMD) as a spatial light modulator (SLM) to control phase and amplitude of the projected light. The light beams from R, G and B lasers expand to form plane waves before reach the display device. After the lights incident the DMD, reflect and pass through the first 4f lens set. The scanner changes the light direction to form angular-tiled holographic 3D displays. The reconstructed images are enlarged by the second 4f lens set.
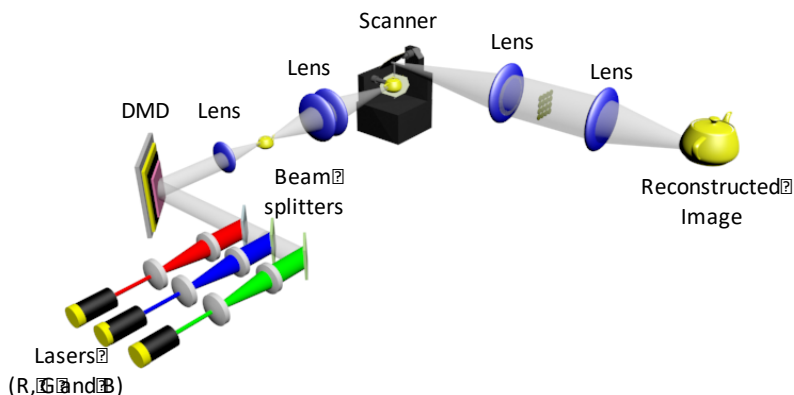

Fig. 3. Angular-tiled Holographic 3D Displays.

4. Conclusion

Two deep neural network techniques are proposed to compute CGHs in real-time. The approach first successfully reduces the dimensionality of the CGHs into only two variables (the other variables of the 16 remain a value of zero). Hence, we can sample a CGH of any orientation angle of the training 3D objects via these two variables. This decreases the data size very much. We then can enlarge the sampled CGHs from a size 128×128×RGB to 1024×1024×RGB in only 0.0090 seconds using the above super-resolution neural network for CGHs.

5. References

[1] T. Sugie, et al., "High-performance Parallel Computing for Next-generation Holographic Imaging", Nature Electronics, 254-259 (2018).

[2] D. P. Kingma, & M. Welling,. "Auto-Encoding Variational Bayes". CoRR, abs/1312.6114. (2013).

[3] C. P. Burgess, et al., "Understanding Disentangling in Beta-VAE", NIPS Workshop of Learning Disentangled Features, (2017).

[4] W. Shi, and etc. "Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-pixel Convolutional Neural Network", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1874–1883, (2016).